Kişisel Video Bilgi Yönetim Sistemleri için bir Çatı Geliştirilmesi

Proje No: 107E234

Doç. Dr. Nihan Kesim ÇİÇEKLİ

Mart 2011

ANKARA

ÖNSÖZ

Bu projede kişisel video bilgi yönetim sistemleri için gerekli altyapı çalışmaları yapılmış, böyle bir sistemin geliştirilebilmesi için bir çatı tasarlanıp, uygulanmıştır. Proje kapsamında değişik araştırma alanlarında çalışılmıştır. Öncelikle videoların ve anlambilimsel etiketlerinin saklanabilmesi için ontoloji-tabanlı bir veri modeli geliştirilmiştir. Bu veritabanındaki videoları sorgulamak için değişik sorgulama yöntemleri araştırılmış ve orijinal arayüzler geliştirilmiştir. Videoların içeriğini sorgulamak için gerekli etiketlerin (yarı-) otomatik olarak çıkarılması için video analiz ve bilgi çıkarım algoritmaları çalışılmış, yeni yaklaşımlar geliştirilmiştir. Projenin diğer önemli bir ayağı da kişiselleştirmedir. Kullanıcı tercihlerine göre yeni içerik önerme kapasitesini sisteme eklemek için değişik yaklaşımlar denenmiş, ontoloji tabanlı kullanıcı profillerine göre yeni video öneren bir sistem geliştirilmiştir. Çalışmalarımızın sonuçları üç dergi makalesi ve onbir konferans makalesi olarak yayınlanmıştır. Proje kapsamında toplam onüç yüksek lisans tezi tamamlanmıştır. Bu rapor, TÜBİTAK (Türkiye Bilimsel ve Teknolojik Araştırma Kurumu) tarafından desteklenen 107E234 numaralı, “Kişisel Video Bilgi Yönetim Sistemleri için bir Çatı Geliştirilmesi” başlıklı bu projenin üç yıllık proje süresinde yapılan çalışmaları özetleyen proje sonuç raporudur.

Doç. Dr. Nihan Kesim Çiçekli

Özet

Çokluortam içeriği e-ticaret, güvenlik, eğitim ve eğlence gibi pek çok alanda kullanılmaktadır. Özellikle, cep telefonları, kameralar ve kayıt cihazları gibi içerik üreten kaynakların hızla çoğalmasıyla kullanıcıların ürettikleri çokluortam verileri çarpıcı bir şekilde artmaktadır. Kaynağı mobil cihazlar olsun, ya da televizyon yayınları, Internet ya da ev cihazları olsun, kişilerin bu çok miktardaki çokluortam verisini saklamak, aramak, erişmek ve tüketmek için akıllı yöntemlere ihtiyaç vardır. Bu projede video içeriğini tarif etmek, organize etmek ve kişiselleştirmek için,bu büyük miktardaki kişisel çokluortam verileri saklamanın, etiketlemenin ve sorgulamanın gelişmiş yollarını mümkün kılacakbir video bilgi yönetim sistemi çatısıönerilmektedir. Önerilen sistemin ayırt edici özellikleri, ontoloji tabanlı bir veri modeli sayesinde videoları anlambilimsel içeriklerine göre saklayabilmesi, videoların içeriğinin ontolojik olarak sorgulanması, (yarı-) otomatik yöntemlerle videoların içeriklerinin etiketlenmesi ve kullanıcı tercihlerine göre yeni videoların önerebilmesidir.

Bu proje kapsamında sunulan sistem, MPEG-7 ontolojisi tabanlıdır ve bu alt yapı sisteme, diğer MPEG-7 ontolojisi uyumlu sistemlerle ortak bir dil üzerinden haberleşebilme ve birlikte çalışabilirlik yeteneği kazandırmıştır. MPEG-7 ontolojisi sistemde temel ontoloji olarak kullanılmakta, kullanıcıya ontoloji tabanlı video içeriği etiketleme ve sorgulama kabiliyetlerini sağlayabilmek için alana özel ontolojiler temel ontolojiye entegre edilebilmektedir. Sunulan sistem, video veritabanlarında, alana özel ontolojilerin kavramları kullanılarak içeriğe yönelik etiketleme ve uzay-zamansal veri modelleme işlevlerini desteklemektedir. Bunların yanı sıra, geliştirilen sistem alana özel ontolojik kavram ve nesnelere yönelik ontoloji tabanlı sorguların oluşturulması ve işlenmesine olanak sağlar. Sorgular form-tabanlı bir arayüz aracılığı ile oluşturulabileceği gibi, MPEG-7 tabanlı alan ontolojilerini anlambilimsel ve uzay-zamansal sorgulamayı sağlamak için gelişitirilen doğal dil sorgu arayüzü ile İngilizce cümleler olarak da sisteme girilebilir. Ayrıca,ontoloji tabanlı bilgi çıkarma ve sorgulama yöntemleri de çalışılmış ve bunların futbol alanına uygulanması gerçekleştirilmiştir.

Bu projede videonun anlamsal içeriğini tarif eden yardımcı verilerin otomatik çıkarılması için video analiz yöntemleri ve video ile birlikte gelen metinden bilgi çıkarım yöntemleri çalışılmıştır. Futbol, film ve haber bültenleri gibi yardımcı veriyle etiketlenmemiş ama açıklayıcı metinle birlikte gelen videoları otomatik etiketlemek için metinden bilgi çıkarım yöntemleri gerçekleştirilmiştir. Uygulama alanı olarak futbol maçları çalışılmış, maç özetlerinden maçın önemli olayları ve zamanları çıkarılarak videolar MPEG-7 standardına göre etiketlenmiştir. Kişiye ait kameralardan ya da cep telefonlarından edinilen ve metin bilgisi olmayan videoların ise video analiz yöntemleri ile yarı-otomatik etiketlenmesi için algoritmalar geliştirilmiştir. Kullanıcı video içinde önemli gördüğü insan yüzlerini birkaç kez etiketlemekte, etiketledikçe bu yüzler öğrenilmekte ve bir süre sonra videonun kalan kısmında o yüzler otomatik olarak etiketlenebilmektedir. Bu bağlamda insanlarınyaş ve cinsiyetlerinin kategorilendirmesi de yapılmıştır. Ayrıca belgesellerin, altyazıları kullanılarak otomatik sınıflandırılması ve özetlenmesi de gerçekleştirilmiştir.

Kişiselleştirme bu projenin diğer önemli bir araştırma yönüdür. Bu projede,kullanıcının beğenebileceği yeni video içeriği önermek için melez bir video öneri sisteminin tasarım, geliştirme ve değerlendirme çalışmaları yapılmıştır. Kullanıcı tercihleri ontoloji tabanlı profiller ile tanımlanmaktadır.Geliştirilen öneri sistemi hem benzer kullanıcı tercihlerini hem de benzer video içeriklerini gözönüne almakta olup,farklı ontolojiler ile farklı içeriklerin önerilmesi için kullanılabilmektedir.

Anahtar Kelimeler:

Çokluortam veri yönetimi, metinden bilgi çıkarımı, anlambilimsel sorgulama, öneri sistemleri, video analizi, doğal dille sorgulama

Abstract

Multimedia content is being used in a wide number of domains ranging from e-commerce, security, education and entertainment. Specifically, the user generated volume of multimedia data has been increasing dramatically with the advent of widely available content generating sources such as mobile phones, camcorders and recorders. Whether from mobile sources, or from TV broadcasts, or from the Internet or from home devices, there is a real need for intelligent ways of storing, searching, retrieving and consuming this large amount of data of individuals. In this project we proposea video information management framework for the description, organization, and personalization of video content that will facilitate advanced ways of storing, annotating and querying enormous amount of personal multimedia data. The distinctive features of the proposed system are: its ability the store semantic contents of videos with respect to its ontology-based data model; ontological querying of the semantic content of videos; (semi-) automatic annotation of the video contents; and recommending new contentaccording to user preferences.

The proposed system is based on MPEG-7 ontology which provides interoperability and common communication platform with other MPEG-7 ontology compatible systems. The MPEG-7 ontology is used as the core ontology and domain specific ontologies are integrated to the core ontology in order to provide ontology-based video content annotation and querying capabilities to the user. The proposed system supports content-based annotation and spatio-temporal data modeling in video databases by using the domain ontology concepts. Moreover, the system enables ontology-driven query formulation and processing according to the domain ontology instances and concepts. The queries can be not only formed by form-based query interfaces but also entered as English sentences through a natural language query interface which is developed for semantic and spatio-temporal querying of MPEG-7 based domain ontologies. In addition, an ontology-based information extraction and retrieval approaches have also been studied and its application to soccer domain is implemented.

In this project, we have studied different algorithms for video analysis and methods for extracting information from the text accompanying the videos in order to automatically extract metadata which describes the semantic content of the videos. New information extraction methods have been developed to automatically annotate videos like soccer games, movies and news which do not have any metadata but come with an explanatory text. Soccer videos are studied as the implementation domain, by extracting the important events in a match and their time periods from match reports, the videos are annotated according to the MPEG-7 standard. In addition, new methods have been developed for semi-automatically annotating videoswhich are obtained from mobile phones or personal camerarecorders and do not have any accompanying text. The user annotates the faces of people who are of special interest manually for a few times, and as s/he annotates the system learns these faces and after some time the system annotates those faces automatically in the remaining part of the video. In this context, the categorization of age and gender has also been studied. In addition we have also implemented automatic categorization and summarization of documentaries by using their subtitles.

Personalisation is another important aspect of the project. This project, proposes the design, development and evaluation of a hybrid video recommendation system to recommend new content that the user would prefer. User preferences are represented as ontology-based profiles. The proposed recommendation system considers both similar user preferences and similar contents of videos and therefore can be used to recommend different content with different ontologies.

Keywords:

Multimedia data management, information extraction from text, semantic querying ,recommender systems, video analysis, natural language querying

İçindekiler

Şekiller

1. Giriş

Video, görüntü, ses gibi çokluortam içeriğin kullanımı her geçen gün yaygınlaşmakta, e-ticaret, güvenlik, eğitim, eğlence gibi pek çok alanda çokluortam verilerinin saklanmasına, sorgulanmasına ihtiyaç duyulmaktadır. Özellikle mobil telefonlar, dijital kameralar, video kayıt cihazları gibi içerik üreten kaynakların geniş çapta mevcut olması nedeniyle, kişisel video, görüntü ve ses kayıtlarının miktarı da çarpıcı bir şekilde artmaktadır.Kaynağı ne olursa olsun bu çokluortam verilerin verimli bir şekilde işlenip, saklanması, taranması, içeriğin anlambilimsel olarak sorgulanması kaçınılmaz hale gelmiştir.

Büyük miktarda çokluortam verilerinin yönetilmesi bilişim teknolojileri alanında aktif bir şekilde çalışılan bir araştırma ve uygulama alanıdır. Çokluortam verilerinin yönetilmesi,temel olarak,video, görüntüve sesle birlikte içeriği tarif eden yardımcı verilerin de saklanmasını, anlambilimsel olarak sorgulanmasını ve değişik ortamlardan erişebilme yöntemlerini içerir. Bu konuda şimdiye kadar yapılmış pek çok uygulama ve akademik çalışma bulunmaktadır. İçeriği tanımlamak için kullanılan yardımcı verileri saklamak için birçok standart geliştirilmiştir. Bunlar arasında MPEG-7 en yaygın kabul gören standarttır. MPEG-7 kullanarak videonun hem fiziksel içeriği hem de anlambilimsel içeriğini tanımlamak ve saklamak mümkündür. Ancak MPEG-7 standardının çok güçlü bir ifade kabiliyetine sahip olması, aynı bilginin farklı şekillerde gösterilebilmesi problemine yol açmaktadır. Başka bir deyişle, aynı bilgi, aynı standardı kullanan farklı sistemler tarafından değişik depolanmakta ve sistemler arasında kopukluk olmaktadır. Uygulamalar arasında birlikte çalışılabilirliği sağlamak amacıyla MPEG-7 standardı yerine MPEG7 ontolojisi kullanılması öngörülmüştür.Ontoloji kullanımıyla çokluortam verilerin içerikleri uygulama alanına uygun olarak çok detaylı bir şekilde yardımcı verilerle etiketlenebilmekte ve böylece içerik anlambilimsel olarak sorgulanabilmektedir.

Çokluortam veri arşivlerinde bir diğer amaç da kullanıcıya-özel içeriğin saklanmasıdır.Arşiv içeriği kullanıcı tarafından güncellenebileceği gibi otomatik arama ajanı tarafından bulunan yeni içerik ile de genişletilebilir. Yeni içeriğin çok çeşitli kaynaklardan aranıp bulunması ve kişiselleştirilmesi gerekir. Kişiselleştirmede amaç kullanıcıların kendi tercihlerine en uygun içeriği bulmalarını sağlamaktır. Öneri sistemleri kullanıcı tercihleri ile içeriği eşleştirmeye çalışan sistemlerdir. Kullanıcı tercihleri (profili) de ontolojik olarak tanımlanabilir. Bu durumda ontolojik olarak etiketlen içerik ile kullanıcı tercihlerinin karşılaştırılması anlambilimsel olarak uygun içeriği bulmada kolaylık sağlar.

Bu projede yukarıda belirtilen ihtiyaçları ve tercihleri karşılayabilecek bir çokluortam veri yönetim sistemi elde edebilmek için gerekli teknolojiler araştırılıp detaylı altyapı çalışmaları yapıldı. Projede, hem ses hem de görüntü verilerini içerdiği için özellikle videoların yönetimi konusu çalışıldı. Bu kapsamda ilk önce yardımcı veri modellemesi üzerinde kapsamlı araştırmalar yapıldı. MPEG-7 standardı ve MPEG-7 ontolojileri hakkında bilgi edinildi. Birlikte çalışabilirlik amaçlandığı için MPEG-7 standardını MPEG-7 ontolojisi haline getirme çalışmaları başlatıldı. MPEG7 standardını tamamen kapsayan Rhizomik modeli MPEG-7 ontolojisini kullanmaya karar verildi (GARCIA, 2005).

Bu karardan sonra video yönetim sistemini gerçekleştirme çalışmaları başlatıldı. Bu proje kapsamında geliştirilen sistem, kullanıcılara genel olarak bir video içeriği yönetim çatısı sunmaktadır. Geliştirilen sistem MPEG-7 ontolojisi tabanlıdır ve alana özel ontolojilerin MPEG-7 ontolojisine entegre edilmesine olanak sağlamaktadır. MPEG-7 ontolojisi kullanımı sisteme, yaygın bir standart olan MPEG-7 standartlarıyla uyumluluk ve diğer sistemlerle ortak bir dil üzerinden haberleşebilme ve birlikte çalışabilirlik yeteneği kazandırmıştır. Ayrıca, alana özel ontolojilerin sisteme entegre edilmesiyle kullanıcıya ontoloji tabanlı video içeriği etiketleme ve sorgulama kabiliyetleri sunulmaktadır. Geliştirilen video yönetim sistemi bilgi alanlarından bağımsızdır, herhangi bir alana ve alan ontolojisine bağımlı değildir. Birbirinden farklı alan ontolojileri sisteme entegre edilerek video etiketleme ve sorgulama işlemlerinde bu alan ontolojilerinin içerdiği kavramlar kullanılabilmektedir. Geliştirilen video yönetim sistemi

modüler bir yapıdadır ve sistem ontoloji yönetimi, video etiketleme ve sorgu işleme alt modüllerinden oluşmaktadır.

Videoları yardımcı verilerle doğru ve detaylı etiketleme, sistemin başarısında anahtar rolü oynamaktadır. Çünkü videoların içeriği yardımcı veriler kullanılarak sorgulanmaktadır. Video içeriğini etiketlemek için en naif yol elle etiketlemektir. Kullanıcı istediği sahneleri, istediği kişi, nesne ya da olayları ontolojiye uygun olarak etiketleyebilir. Etiketlerin hangi sahne aralıklarında göründükleri ya da ekranın hangi bölgesinde göründükleri de açıkça işaretlenebilir. Ancak videoları elle etiketleme insan emeği gerektiren çok zahmetli ve vakit alıcı bir iş olduğu için tercih edilmemektedir. Bunun yerine içeriğin otomatik ya da yarı-otomatik etiketlenmesi arzu edilmektedir. Bu nedenle bu proje kapsamında otomatik video etiketleme çalışmaları yapılmıştır.

Otomatik etiketlemede değişik veri kaynakları kullanılabilir. Bunlardan en önemlisi videonun kendisidir. Dijital içerik görüntü işleme ve örüntü tanıma yöntemleri ile analiz edilip bilgi çıkarılabilir. Diğer kaynak ise varsa video ile birlikte gelen metinlerdir. Maç özetleri, film/haber alt-yazıları, işitme engelliler için altyazılar (closed caption) bunlara örnek olabilir. Bu metinlerden de bilgi çıkarım yöntemleri kullanılarak çok yararlı etiketler çıkarılabilir. Başka bir yöntem de hem metin, hem görüntü, hem de ses verilerini birlikte kullanmaktır. Bu kaynaklardan elde edilecek bilgiler birbirini tamamlayıcı nitelikte olduğu için çok daha kesin etiketler elde edilebilir. Bu projede, videoları otomatik etiketlemek için değişik video analiz ve metinden bilgi çıkarım yöntemleri çalışıldı. Video analiz yöntemlerini videoların içindeki insanları tanıma ve etiketleme amacı için kullandık. Metinden bilgi çıkarım yöntemleri ile maç özetlerini ve belgesel alt yazılarını analiz ettik. Maçlardaki önemli olayları gerçekleştikleri zaman aralıklarıyla birlikte otomatik etiketleyen yöntemler geliştirdik. Belgesel altyazılarını kullanarak videoların kategorilerini belirleyen ve videoların özetlerini hazırlayan sistemler geliştirdik. Hem video, hem metin analizi, hem de ses bilgisini kullanarak maçları daha doğru zamanlamalarla etiketlemeyi başardık. Proje kapsamında başlayan halen devam eden çalışmalarımızda ise video analiz yöntemleri ile insan vücudunu, belli kategorilerdeki nesneleri tanıyan programlar geliştirmekteyiz. Bu çalışmaların sonunda insanları ve belli nesneleri içeren olayları otomatik tanıma çalışmaları başlatmayı amaçlıyoruz.

Geliştirilen sistem, alan ontolojilerine özgü kavramlar kullanılarak video içeriği üzerinde ontolojik kavramsal sorgulama, uzay-zamansal sorgulama, birleşik sorgulama (birden fazla sorgu tipini birleştirilmesiyle oluşturulan sorgu) kabiliyetleri sunmaktadır. Ontolojik kavramsal sorgulamada bir videoda belirli kişilerin, nesnelerin ya da olayların göründükleri sahneleri sorgulamak mümkün olduğu gibi etiketlenen objeleri ontolojik olarak genellemeler yaparak da sorgulamak mümkündür. Örneğin, “Hakan Şükür” ontolojide insan-erkek-futbolcu zincirinde etiketlenmiş olsun. “Hakan Şükür’ün göründüğü sahneler”,“Futbolcuların göründüğü sahneler” ve “Erkeklerin göründüğü sahneler” sorgularının hepsinin sonuçlarında Hakan Şükür’ün göründüğü sahneleri görmek mümkün olmaktadır. Uzay-zamansal sorgulama yine ontolojik kavramların belli video zaman aralıklarında ya da ekranın belirli bir bölgesinde görünüp görünmediğini sorgulamayı sağlamaktadır. Birleşik sorgulamada ise farklı sorguları VE/VEYA bağlaçları ile içiçe yazmak mümkün olmaktadır.

Kullanıcının bütün bu sorgulama kabiliyetlerini kolaylıkla deneyimlemesi için farklı arayüzler geliştirildi. En temel arayüz bütün sorgu çeşitlerini farklı formlarla kullanıma açan form-tabanlı arayüzdü. Bu sorgulama aracının temel amacı sistemin bütün sorgulama işlevselliğini kanıtlamaktı. Ancak kullanım kolaylıyı ön plana alınınca kelime tabanlı ontolojik sorgulama ve doğal dille sorgulama yöntemlerini araştırmak söz konusu oldu. Kelime tabanlı ontolojik sorgulama futbol alanındauygulanmıştır. Genel olarak, kullanılabilirlik, ölçeklenebilirlik ve sorgulama performansı sorunlarıyla ilgilenilmiştir. Alana özgü bilgi çıkarma, anlamsal çıkarım yapma ve kurallar ile performans büyük ölçüde arttırılmıştır. Ölçeklenebilirlik ise, anlamsal indeksleme yöntemiyle elde edilmiştir.

Videoları doğal dille (İngilizce) anlambilimsel ve uzay-zamansal sorgulamak için ise MPEG-7 tabanlı alan ontolojilerini sorgulayabilen bir arayüz geliştirilmiştir. Kullanıcı sistemde kavramsal, karmaşık nesne (“VE”, “VEYA” ile bağlanmış birden fazla nesne), uzaysal (sağ, sol. . .), zamansal (önce, sonra, en az 10 dakika önce, 5 dakika sonra. . .), nesnesel yörünge ve yönsel yörünge (doğu, batı, güneydoğu, . . . , sağ, sol, yukarı . . .) sorguları yapabilmektedir. Ayrıca kullanıcının negatif anlam taşıyan sorgu yapabilmesini sağlamak amacıyla,doğal dil sorgu cümlesindeki negatif anlamlar da tespit edilebilmektedir. Girilen sorgu cümlesi link ayrıştırıcı ile ayrıştırılıp gerekli bilgiler çıkarıldıktan sonra sorgu içeriği bir SPARQL sorgusuna dönüştürülmektedir. Bu sorgu da ontoloji üzerinde yürütüldüğünde cevap olacak video parçalarının süre bilgileri sonuç olarak döndürülmektedir. Bu süreler video üzerine doğru sahnelerin bulunmasını sağlamaktadır.

Proje kapsamında paralel olarak yürütülen diğer önemli araştırma konusu da öneri sistemleri olmuştur. Kişiye özel içerik bulma konusunda yaptığımız araştırmalar kapsamına işbirlikçi, içerik odaklı ve melez öneri sistemleri çalışıldı. Genelde melez sistemlerin daha başarılı öneriler yapabildikleri gözlemlendi. Bu nedenle farklı ortamlardan kişiye özel öneri yapabilecek melez sistemler geliştirildi.

Yapılan bütün bu çalışmalar,projenin bu sonuç raporunda yenilikçi yönlerine ağırlık verilerek anlatılmıştır. Rapor proje kapsamında paralel olarak yürütülmüş olan temel çalışma alanlarına göre organize edilmiştir.2. Kısımda ontoloji-tabanlı video bilgi yönetim sistemi sunulmaktadır.3. Kısımda videoların (yarı-) otomatik olarak anlambilimsel etiketlenmesi için geliştirdiğimiz yöntemler anlatılmaktadır. 4. Kısımda anlambilimsel sorgulama yöntemleri gösterilmiştir. 5. Kısım kullanıcı tercihlerine göre içerik bulma konusunda yaptığımız çalışmaları anlatmaktadır. 6. Kısımda proje son döneminde yapmış olduğumuz entegrasyon işleri anlatılmaktadır. 7. Kısımda ise elde edinilen kazanımlar, sonuçlar ve gelecekteki ilintili araştırma konuları sunulmuştur.

2. Ontoloji-tabanlı Video Bilgi Yönetim Sistemi

Projede önerilen sistemin mimarisi ontoloji tabanlı çokluortam veritabanına dayanmaktadır. Çokluortam verileri (videolar, görüntüler, ses kayıtları vs.), anlambilimsel içeriklerinin anlatıldığı yardımcı verilerle birlikte saklanmaktadırlar. Anlambilimsel içerik çok değişik şekillerde saklanabilir. Literatürdeki çalışmalar sonucunda birçok çokluortam yardımcı veri modelleri ortaya çıkmıştır. Bu modelleri standartlaştırmak amacıyla MPEG tarafından bir ISO/IEC standardı olan MPEG-7 geliştirilmiştir. MPEG-7, bir bilgisayar kodu tarafından erişilebilen bilginin anlamını bir dereceye kadar yorumlamayı destekleyebilen çoklu ortam içerik verisini tanımlamak için bir standarttır. MPEG-7 standardının kapsamı çok geniş olduğundan aynı video içeriği çok farklı şekillerde tanımlanabilmektedir. Bu da birlikte işlerlik (interoperability) sorunlarına sebep olmaktadır.

Literatürde, birlikte işlerlik sorunlarına bir çözüm olarak videoların etiketlenmesindeMPEG7’ye dayalı ontoloji kullanımı önerilmektedir (ARNDT, 2007; TSINARAKI, 2007; GARCIA, 2005; HUNTER, 2001). Bu nedenleMPEG-7 standardını MPEG-7 ontolojisi haline getirme çalışmalarını başlattık (TARAKCI, 2008). Geliştirdiğimiz sistemde MPEG7 standardını tamamen kapsayan Rhizomik modeli MPEG-7 ontolojisini kullanmaya karar verdik(GARCIA, 2005). MPEG7 ontolojisinin temel amacı alan ontolojilerini MPEG-7 ile kolayca entegre etmektir. Bizim geliştirdiğimiz sistemde kullanıcının seçtiği herhangi bir alan ontolojisi ile etiketlenmiş video içeriği kullandığımız MPEG-7 ontolojisine çevrilmekte ve veritabanında OWL dosyaları olarak saklanmaktadır (TARAKÇI, 2008). Sistemin video etiketleme modülü çalışırken kullanıcı istediği alan ontolojisini seçebilmekte ya da yükleyebilmektedir. Kullanıcı video içeriğini bu seçilen alanontolojisindeki kavramlara göre etiketleyebilmektedir. Kullanıcının seçtiği alanontolojisi (ör. Futbol, aile, haber vs.) en üst seviyedeki MPEG-7 ontolojisi ile sarılabilimekte; ontolojik olarak etiketlenmiş video

görüntüleri yine ontolojik olarak sorgulanabilmekte; OWL dilinin özelliği olarak çıkarımsal sonuçlar da kullanıcıya sunulabilmektedir.

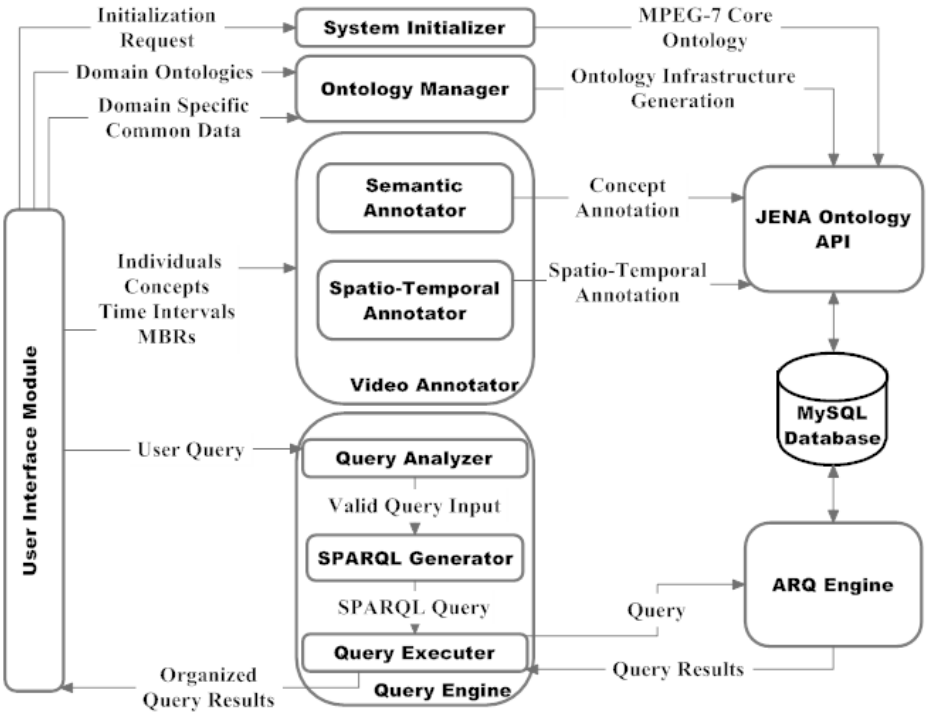
Geliştirilen bu temel altyapının en önemli özelliği mimarisinin herhangi bir alan ontolojisini yüklemeye izin verebilecek şekilde esnek olmasıdır. Kullanıcı futbol maçlarını etiketlemek istiyorsa bunun için bir futbol alan ontolojisini yüklemesi yeterli olacaktır. Kullanıcı filmleri etiketlemek istiyorsa film ontolojisi yükleyecektir.Videolar da bu ontolojilere göre etiketlenip sorgulanabilirler.Ayrıca geliştirilen çatının tasarımı modüler olduğu için yeni modüller kolayca eklenebilip, çıkarılabilmektedir. Örneğin otomatik yüz etiketleme modülü, doğal dille sorgulama modülü, metin bilgisinden otomatik bigi çıkarım modülü sisteme entegre edilebilmişlerdir.

Projenin ilk senesinde ortaya çıkan bu çoklu-ortam veri tabanı sistemi, genel amaçlı bir video bilgi yönetim sistemi elde edebilmek için sonraki dönemlerde daha da geliştirilmiştir. Geliştirilen video yönetim sistemi modüler bir yapıdadır. Sistemin genel mimarisi Şekil 1'de gösterilmiştir. Sistem temel olarak*ontoloji yönetimi*, *video etiketleme* ve *sorgu işleme* alt modüllerinden oluşmaktadır.

Ontoloji yönetim modülü, MPEG-7 ontolojisinin oluşturulması, ilgili alan ontolojilerinin sisteme eklenmesi, sistemden silinmesi ve eklenen alan ontolojilerinin MPEG-7 ontolojisiyle entegre edilmesi kabiliyetlerini sunmaktadır. Ayrıca, alan ontolojileri ile ana MPEG-7 ontolojisinin entegrasyonu sağlandıktan sonra, alana özgü ortak verilerin de ontolojik alt yapıya entegre edilebilmesi sağlanmıştır. Ontoloji yönetim modülünün tasarımında, ilk ontolojik altyapı temel alınmıştır. Daha sonra bu ontolojik alt yapıya, alana özgü uygulamalarda ortak olarak kullanılan verilerin de entegre edilmesi sağlanmıştır. Bu ortak verilerin entegre edilip kullanılmasıyla video içeriğini etiketleme çabasının azaltılması ve sorgu işlemi sırasında kullanıcıya alana özgü ortak bir dil sunulması sağlanmıştır.

Video etiketleme modülü, kullanıcıya alana yönelik ontolojilere özgü kavramları kullanarak nesneler tanımlama ve tanımlanan bu nesneleri video etiketleme işlemi sırasında kullanma imkanı sunmaktadır. Ayrıca ontolojik alt yapıya eklenmiş olan alana özgü ortak veriler de video etiketlemede kullanılabilmektedir. Video etiketleme işlemi elle gerçekleştirilirken, form tabanlı bir ara yüz üzerinden etiketlenmek istenen nesne ve bu nesnenin ait olduğu alan ontolojisi kavramı seçilir. Daha sonra üzerinde etiketleme yapılmak istenen video seçilir ve oynatılır. İlgili nesnenin göründüğü zaman aralıklarının başlangıcında video durdurulur ve video ara yüzü üzerinde etiketlenmek istenen nesne, en küçük çevreleyici dikdörtgen içine alınarak işaretlenir. Etiketlenmek istenen nesneyi çevreleyen dikdörtgen çizimi sayesinde ilgili nesnenin video üzerindeki koordinatları alınmış olur. Daha sonra video tekrar oynatılarak etiketlenmek istenen nesnenin ilgili alanda göründüğü tüm zaman aralığı bilgisi alınır ve etiketleme kaydedilir. Video üzerinde ilgili nesnelerin etiketleme işlemi tamamlandığında, etiketleme sırasında elde edilen nesne koordinatları ve nesnelerin göründüğü zaman aralığı bilgileri etiketlenmiş olan nesneler arasındaki uzay-zamansal ilişkilerin hesaplanmasında kullanılır. Bu sayede, seçilen videolar, içerdikleri nesnelerin birbirlerine göre uzaysal ve zamansal ilişkilerini de içerecek şekilde etiketlenmiş olmaktadır.

Oluşturulan uzaysal ilişkiler yönsel (kuzey, güney, doğu, batı), arayönsel (kuzeydoğu, kuzeybatı, güneydoğu, güneybatı), pozisyona dayalı (sol, sağ, yukarı, aşağı) ve topolojik (kesişen, içinde, içeren, çakışık, temas eden, ayrık, kapsayan, kapsanan) ilişkiler olarak alt gruplara ayrılmaktadır. Zamansal ilişkiler de Allen tarafından formülleştirilmiş olan zamansal ilişkiler kümesini içermektedir (ALLEN, 1983). Nesneler arasındaki uzay-zamansal ilişkilerin tamamı sorgu işleme aşamasından önce hesaplanarak sistemde saklanmaktadır, böylece sorgu işleme sırasında bu hesaplamaları yapmak ve ilişkileri yeniden oluşturmak için ekstra bir çaba harcanmamaktadır. Bu sayede sorgu işleme süreci daha hızlı ve verimli bir performans göstermektedir.



Şekil 1. Ontolojik Video Bilgi Yönetim Sistemi Mimarisi

Sorgu işleme modülü, kullanıcının form tabanlı bir ara yüz üzerinden oluşturduğu sorguların analiz edilmesi, işlenmesi ve eşleşen sonuçların gösterilmesi işlevlerini yönetmektedir. Geliştirilen sistemde alan ontolojilerine özgü kavramlar kullanılarak video içeriği üzerinde

- Ontolojik kavramsal sorgulama,
- Uzay-zamansal sorgulama,
- Birleşik sorgulama (Birden fazla sorgu tipini birleştirilmesiyle oluşturulan sorgu) kabiliyetleri gerçekleştirilebilmektedir.

Ontolojik kavramsal sorgulamatipi sayesinde ilgilenilen nesnelerin, olayların ve kavramların bulunduğu sahneler listelenebilir. Örneğin, "Ahmet'in göründüğü sahneler", "Herhangi bir erkeğin göründüğü sahneler" ve "Gol olayının gerçekleştiği sahneler" tarzındaki sorgular ontolojik olarak oluşturulabilir. Uzaysal ilişkilere dayalı sorgulamalarda nesnelerin birbirlerine göre uzaysal ilişkileri sorgulanır. Örneğin, "Ali'nin, Ayşe'nin solunda olduğu sahneler", "Herhangi bir futbolcunun topa temas ettiği sahneler" gibi sorgular oluşturulabilir. Zamansal ilişkilere dayalı sorgulamalarda nesnelerin ve kavramların birbirlerine göre zamansal ilişkileri sorgulanır. Örneğin, "Ali'nin Ayşe'den önce göründüğü sahneler", "Herhangi bir insanın herhangi bir arabayla aynı anda göründüğü sahneler" gibi sorgular oluşturulabilir. Birleşik sorgu kabiliyeti, sistemde ontolojik kavramlara yönelik sorguları ve uzay-zamansal sorguları "AND" / "OR" mantıksal bağlaçlarıyla bir araya getirip birleştirerek sorgulama yapmayı sağlamaktadır. Bu sorgu tipleri bu raporun 4.kısmında daha detaylı anlatılmaktadır.

Ontoloji-tabanlı video bilgi yönetim sistemi geliştirilmesi kapsamında yaptığımız çalışmalardan üç yüksek lisans tezi çıkmıştır (TARAKÇI, 2008; ŞİMŞEK, 2009; DEMİRDİZEN, 2010). Entegre edilmiş sistemi anlatan dergi makalesi yazım aşamasındadır.

3. Videoların (Yarı-) Otomatik olarak Anlambilimsel Etiketlenmesi

Bu projede video içeriğini otomatik olarak algılama ve yardımcı veri çıkarma konusunda çeşitli araştırmalar yapılmış, yeni yöntemler geliştirilmiştir Videoların saklanması ve sorgulanması için etiket dediğimiz yardımcı verilere ihtiyaç vardır. Bu yardımcı veriler videonun anlamsal içeriğini tarif eden bilgilerdir. Video içinde görünen kişi, olay, nesne ya da kavram gibi bilgileri içerebilir. Ancak çoğu video bu yardımcı veriden yoksun olduğu için içerik anlatan bu bilginin videodan tercihan otomatik olarak çıkarılması gerekir. Çokluortam verilerden (görüntü, video, ses gibi) bilgi çıkarımı oldukça çok çalışılan bir araştırma konusudur. Genelde video analiz yöntemleri, ses analizi, video/görüntü ile birlikte gelen metinden bilgi çıkarım yöntemleri ile yardımcı veriler çıkarılabilir.

Bu projede geliştirilen video yönetim sistemi istenen video sahnelerininelle etiketlenmesini sağlayan bir arayüz içermektedir. Bu arayüzle kullanıcı, istediği sahneleri durdurmakta, belli kişi, nesne ya da olayları ekranda göründükleri yerleri ve zamanları ile birlikte etiketleyebilmektedir. Bu etiketle işleminde alan ontolojilerindeki sınıflar, bireyler ve veri/sınıf özellikleri kullanılabilmektedir. Daha sonra bu biriken yardımcı veriler ontolojik olarak sorgulanabilmekte, çeşitli çıkarımsal sonuçlar da elde edilebilmektedir.

Ancak elle etiketleme insan emeği gerektiren, çok zaman alıcı, zahmetli bir iştir. Etiketlenen içeriğin zamanlarını doğru kaydetmekte hata payı yüksektir. Bu nedenle etiketlerin tercihan otomatik, olmuyorsa yarı-otomatik çıkarılması istenmektedir. Video verisi Internet, televizyon, DVD, kişisel kamera gibi değişik kaynaklardan elde edilebilir. Internet, T.V., DVD gibi kaynaklardan elde edilen videolarla birlikte gelen birtakım metin bilgisi de bulunabilir. Örneğin videonun indirildiği web sitesindeki yazılar, televizyondan kaydedilen programlarla ilgili elektronik program bilgisi, DVD'deki filmlerin altyazıları gibi metin kaynakları bilgi çıkarımı amacıyla kullanılabilir. Bu metinler tek başlarına kullanılabilecekleri gibi, video görüntüsü ve metin bilgisi birlikte kullanılarak daha doğru bilgiler çıkarılabilir. Kişisel videolar için ise metin bilgisi yoktur; içerik bilgisi sadece görüntülerden video analiz yöntemleriyle çıkarılabilir.

Bu projede videoların otomatik etiketlenmesi için elde olan bütün verilerin kullanımını içeren yöntemlerin çalışılması öngörülmüştür. Diğer bir deyişle, video ile gelen metin bilgisi, video görüntüsünün kendisi, hem video hem metin içeriğinin analizi gibi değişik yöntemler denemiş, her yöntem için uygun uygulama alanlarında deneyler yapılmıştır. Bu kısımda videoları otomatik etiketleme için yapılan çalışmalar anlatılmaktadır.

3.1. Metinden yardımcı veri çıkarılması

Videoların bir kısmı içeriğini açıklayan metin bilgisiyle bulunabilmektedir. Örneğin maç anlatımları, filmlerdeki altyazılar, haberlerin açıklamaları bu tür metin bilgileridir. Bu projede yardımcı veriyle etiketlenmemiş ama metinle birlikte gelen videoları bilgi çıkarım yöntemleri ile otomatik etiketleme çalışmaları yapılmıştır. Uygulama alanı olarak futbol maçları çalışılmıştır.

Internet'te UEFA, sporx gibi sitelerde dakika bilgisi ile maç özetlerini bulmak mümkündür. Bir web crawler yazılarak UEFA sitesinden UEFA kupası ve UEFA şampiyonlar ligi maçlarının İngilizce metin özetleri indirildi. İndirilen maç özetleri NEKOHtml ile parse edildi. Futbolcu isimleri, takım isimleri, dakika bilgileri regular expression'lar ile tanımlandı ve metin içinde bulundukları yerler saptandı. Her cümledeki oyuncu isimleri, takım isimleri, dakika bilgisi etiketlendi ve cümledeki diğer bütün kelimeler token olarak etiketlendi. Etiketlenmiş cümleye bir örnek aşağıdadır:

```
<Sentence>
    <Minute>87</Minute>
    <PlayerName>Crouch</PlayerName>
    <Token>(</Token>
    <Team>Liverpool</Team>
    <Token>)</Token>
    <Token>has</Token>
    <Token>an</Token>
    <Token>effort</Token>
    <Token>on</Token>
    <Token>goal</Token>
    <Token>.</Token>
</Sentence>
```

Etiketlenen cümlelerden önce elle yazılmış kurallar vasıtasıyla olay çıkarımı yapıldı. Maçlarda önemli olan gol, korner, faul, offside, penaltı gibi olaylar cümlelerden çıkartılıp bir XML dosyası halinde saklandı. Örneğin korner olayını tanıyan kural aşağıdaki gibi yazıldı.

```
<Rule>
    <Pattern>
            <Minute></Minute>
            <PlayerName></PlayerName>
            <Token>(</Token>
            <Team></Team>
            <Token>)</Token>
            <Token>delivers</Token>
            <Token>the</Token>
            <Token>corner</Token>
            <Token>.</Token>
    </Pattern>
    <MatchEvent>
        <CornerEvent>
        <Minute></Minute>
        <PlayerName></PlayerName>
          <Team></Team>
        </CornerEvent>
    </MatchEvent>
</Rule>
```

Futboldaki önemli olayları anlatmak için kullanılan her cümle şekli için kurallar yazıldıktan sonra etiketlenmiş cümleler haline getirilmiş maç özetleri işlenerek her maç için anlambilimsel yardımcı veriler otomatik olarak çıkarıldı. Çıkarılan bu anlambilimsel yardımcı veriler MPEG-7 formatına dönüştürülüp video ile birlikte saklandı. Bu yardımcı veriler sayesinde maçlar metin bilgisi ile senkrone bir şekilde saklanmış oldu. Böylece maçlar değişik şekillerde sorgulanabiliyor duruma geldi. Geliştirilen sorgulama sistemi ile maçların ilgili kareleri oyuncu adına, takım adına, dakika bilgisine ya da olaya göre sorgulanabilmektedir. Sorgulama sistemi XQuery dilini kullanmakta olup, sorgu tipine göre query değişmektedir. Bunun için de MPEG-7 üzerinde dinamik olarak sorgu oluşturan bir modül yazılmıştır. Buradan dönen sorgu sonuçları liste şeklinde görüntülenebilmekte ve olayla ilişkilendirilmiş video varsa olay anı seyredilebilmektedir.

Daha sonra kuralları elle yazmak yerine, kuralları metinden otomatik öğrenmek için algoritmalar denedik. WHISK algoritmasını kullanarak maçlardaki önemli her olay için kuralları supervised bir şekilde öğrenen bir program geliştirdik. Maç özetlerinin yapısı nispeten belirgin olduğu için öğrenme algoritması başarılı bir şekilde çalıştı. Böylece maç özeti ile birlikte gelen videoları otomatik olarak etiketlemek, bunları video ile senkrone etmek ve değişik şekillerde sorgulamak mümkün oldu.

Projede ayrıca dile bağlı olmadan maç özetlerinden bilgi çıkarımı çalışmaları yürütülmüştür. Birçok gramatik ve sözdizim hatası olabilen ham maç özetlerinden bilgi çıkarımı yapabilmek

için her maç olayı için elle oluşturulan şablonlar kullanılmıştır. Dilden bağımsız olduğu için çözümleyici, part-of-speech tagger, phrase chunker gibi herhangi bir dil işleme aracı kullanılmamıştır. Diğer otomatik bilgi çıkarım yaklaşımlarıyla karşılaştırıldığında önerilen sistemin kesinliği %85-%97 civarındadır. Bu yaklaşım başka alanlardaki metinlere ve dillere de uyarlanabilir. Geliştirilen yöntem hem Türkçe hem de İngilizce maç özetlerinde denenmiştir.Bu çalışmada üç ana adım vardır: Önce web tarayıcı ile varolan maç özetleri Sporx vs. gibi sitelerden indirilmektedir. Sonra bu sitelerdeki nizamlı veriler kullanılarak her maç özetindeki isimlendirilmiş varlıklar (named entities) etiketlenir. En son olarak iki seviyeli sözlüksel analiz her maç olayı için maç anlatımını ve şablonları analiz ederek maçtaki olayların çıkarımını yapar.

Bu çalışmalarımızı üç konferans makalesi olarak yayınladık (ALAN, 2008; GÖKTÜRK, 2008; TUNAOĞLU, 2009). Ayrıca bu konuyla ilgili bir yüksek lisans tezi ortaya çıkmıştır (GÖKTÜRK, 2008).

3.2. Metin bilgisi ve Video Analiz yöntemlerinin birlikte kullanılması

Futbol maç özetlerinden zamanlarıyla çıkardığımız olayların videodaki görüntüsel sınırlarını belirlerken karşılaştığımız sorunlardan birisi video zamanının, olay zamanıyla eşzamanlı olmamasıydı. Maç öncesi ve devre arasının da videoya dahil olması, maç zamanıyla video zamanının senkronizasyonunu bozuyordu. Çıkarılan olay etiketleri ve video görüntüsünün daha iyi eşzamanlı hale getirilmesi için metinden gelen bilgilere ek olarak video analiz yöntemleri kullanarak olay sınırlarını otomatik olarak belirleme çalışmaları yapılmıştır.

Televizyonda futbol karşılaşmaları yayınlarında birden fazla kamera kullanılır. Pozisyonların önemine, çeşidine göre belli aralıklarla kameralar arası geçişler yapılır, yayın değişik kameralardan verilir, gerektiğinde aynı pozisyonun değişik kamera açılarıyla birkaç kez tekrarı yapılır. Olayların yakalanmasında ve sınırlarının belirlenmesinde kamera değişimleri önemlidir. Bu projede metin bilgisinden çıkarılan olay ve yaklaşık dakika bilgisini ve videonun bu dakikası civarındaki kamera geçişlerini kullanarak olayın tam sınırları belirlenmeye çalışılmıştır.

Videoların kamera değişmeden bir seferde çekilen parçasına çekim diyoruz. Futbol videolarında çekimleri 3 sınıfa ayırabiliriz. Yakın plan, orta plan ve uzak plan çekim (Bkz. Şekil 2).

Şekil 2. Sırasıyla uzak plan, orta plan ve yakın plan görüntü

Olay sınırlarının bulunması için gerekli ilk adım, video analizi ile çekimlerin bulunması ve sınıflandırılmasıdır. Çekimlerin bulunması için video çerçeveleri (frame) teker teker okunup her beşinci frame bilgi çıkarmak için kullanılmıştır. Bilgi toplama için renk histogramından ve Canny kenar yakalama algoritmasından yararlanıldı. Renk histogramı, iki frame arasındaki renk histogram farkı, toplam histogram değişim değeri ve kenar pikselleri sayısı gibi özellikler belirlendi.

Çekimleri SVM'de sınıflandırmak için model oluşturma aşamasında öncelikle veriler incelenerek nasıl bir model oluşturulması gerektiğine dair yapılan değerlendirme sonucunda, uzak plan görüntülerini renk histogramı kullanarak diğer görüntülerden ayırt edebileceğimiz;

yakın ve orta plan görüntülerini de kenar piksel sayısını kullanarak ayırt edebileceğimiz gözlemlendi. Bu bilgiler ışığında modelimiz iki SVM'den oluşturuldu. İlk kısım renk histogram bilgisi ile eğitilerek uzak plan görüntülerin ayrılmasında kullanıldı, İkinci kısım da ilk kısımda ayrılamayan orta plan ve yakın plan görüntüleri ayırmak için kenar piksel sayısı ile eğitildi.

Olay sınırlarının bulunmasında ses analizi de yapılmıştır. Videonun her saniyesi icin bütün örneklerin rms(root mean square)i hesaplanıp sesin büyüklüğünü ifade eden bir değer elde edilmektedir.

Çekim bulma, sınıflandırma ve ses analizinden elimizde bulunan veriler şunlardır:

1- Çekimler(başlangıç zamanı, bitiş zamanı, uzunluğu).
2- Çekim Sınıfları(Uzak Plan çekim(F), Orta Plan Çekim(M), Yakın Plan Çekim(C)).
3- Ses büyüklüğü ortalaması (Videonun her saniyesi için).
4- Maç anlatımlarından çıkarılan, olay çeşidi ve olayın yaklaşık gerçekleşme dakikası.

Bir futbol yayınında, genel yayın uzak plan çekimle yapılır. Bir olay gerçekleşirken ya da gerçekleştikten sonra orta plana ve/veya yakın plana geçilir. Olay bittikten sonra tekrar normale dönülür ve uzak plan çekime geçilir. Bu nedenle olayları iki uzak plan görüntü arasında aramak cok mantıklıdır. Şekil 3'de bir gol olayına ait çekim dizisi bulunmaktadır. Görüldüğü gibi Gol olayı uzak plan çekimle başlayıp uzak plan çekimle sona ermektedir.

Şekil 3- Örnek Gol Çekim Dizisi

Bu nedenle olayların sınırlarını belirlerken, yukarıdaki gözlemden yararlandık. Metin bilgisinden çıkarılanyaklaşık dakika ve olay alınıp,gelen dakikanın öncesindeki ve sonrasındaki üçer uzak plan "shot" belirlendi. Elimizdeki bu altı uzak plan görüntü (Far view) arasında olayın olma ihtimali olan beş aralık oluşturuldu.

Örnek: F-----F-----F---referans dakika----F----F-----F

Elimizde bulunan ses ve video bilgileriyle her olay için kurallar oluşturuldu. Puanlama sistemiyle bu kurallara en yakın olan aralık seçilip bu aralıktaki "shot" lar birleştirilerek olayın sınırları belirlendi.

Yukarıda anlatılan çalışmanın detayları bir konferans makalesinde yayınlanmıştır (BAYAR, 2010).

Bütün etiketlemeyi tamamen otomatize etmek için videonun ilk yarı ve ikinci yarı başlangıçlarını bulmak gerekmektedir. Bu amaçla futbol videolarında bulunan zaman göstergesini kullanarak ilk yarı ve ikinci yarı başlangıç zamanlarını bulmaya çalıştık. Futbol maçları videolarında bulunan zamanlayıcıgenelde sağ üst veya sol üst köşede bulunmaktadır. Görüntüde rakam bulma algoritmaları kullanılarak dijital saat ve dolayısıyla videodaki frame belirlenebildiği için video zamanıyla maç zamanı senkronize edilmiş oldu.

Bu kapsamda yapılan çalışmalar Haziran 2011'de mezuniyeti planlanan bir yüksek lisans öğrencisinin tezinde anlatılacaktır.

3.3. Video Analiz ile Yüz Etiketleme

Projenin diğer bir önemli modülü de kullanıcı desteğiyle anlambilimsel etiket çıkarımının video analiz yöntemleri ile yarı otomatik yapmaya imkan veren etiketleme aracıdır. Etiketleme aracı kişisel kameralardan, cep telefonlarından ya da video kayıt cihazlarından elde edilen videoları anlambilimsel içeriklerine göre yardımcı verilerle etiketlemek amacıyla kullanılmaktadır. Videodaki insanların göründükleri sahneler kullanıcıya gösterildiğinde kullanıcı getirilen insanları yüklenen alan ontolojisi yardımıyla anlambilimsel içeriği tarif edecek kavramlarla etiketleyebilmektedir. Daha sonra sistem otomatik olarak o video içinde benzer insanları bulup kullanıcıya göstermekte ve kullanıcının fikrini sormaktadır. Kullanıcı yanlış bulunmuş etiketleri düzeltebilmektedir. Bu şekilde kullanıcının da öğrenme döngüsünün içinde olması ile kısa bir süre sonra pek çok insan yarı otomatik olarak etiketlenebilmektedir. Bir süre sonra ise etiketlemenin tamamen otomatik olarak yapılabilmesi mümkün olmaktadır.

Bu modülün gerçekleştirilmesinde iki yüksek lisans tezi yapılmıştır (YILMAZTÜRK, 2010; YAPRAKKAYA, 2010). İlk tez (YILMAZTÜRK, 2010) kapsamında videolar içindeki yüzlerin otomatik olarak bulunması, tanınması ve otomatik etiketlenmesi için kullanılabilecek literatürdeki en iyi algoritmaları uygulandı ve elde edilen sonuçlar analiz edildi.İkinci tezde (YAPRAKKAYA, 2010) ise elde edinilen tecrübelere göre etkin bir yüz tanıma ve etiketleme aracı geliştirildi. Bu iki tez kapsamında yaptığımız çalışmalar aşağıda özetlenmiştir.

OpenCV Viola-Jones(VIOLA, 2001) yüz tespit algoritması ve “Lucas-Kanade Optical Flow Feature Tracker” (BOUGUET, 2000) yüz takip algoritmalarıyla yüz resimleri toplandı.“Local Binary Patterns” (WOLF, 2008), “Discrete Cosine Transform” (EKENEL, 2005) ve “Histogram of Oriented Gradients” (DALAL, 2005) yöntemleriyle yüz resimlerinden öznitelik vektörleri çıkartılarak, bu vektörler bir öğrenme modeli geliştirmek için kullanıldı. Kullanılan sınıflandırma algoritmaları arasında “Nearest Neighbour”, “Linear Discriminant Analysis” ve “Support Vector Machine” bulunuyordu. Çevrimiçi olarak sıralı öğrenme yöntemi kullanıldı. Bu yöntemde videodaki bazı yüzler kullanıcı tarafından etiketlendikten ve sistem gerekli sayıda veri topladıktan sonra bir öğrenme modeli geliştiriliyor ve videonun ilerleyen bölümlerinde bu modele göre tespit edilen yüzlere isim önermesi yapılıyor. Kullanıcı ise, bir yandan yüz tanıma ve isim önerisi yapılırken bir yandan sistemi yeni karşılaşılan yüzlerle güncelleyerek öğrenilen modeli geliştirebiliyor. Çevrimiçi öğrenme yapılabilmesi için “Linear Discriminant Analysis” ve “Support Vector Machine” algoritmalarının verileri sıralı olarak işleyebilen türevleri kullanıldı. Sınıflandırma için geliştirilmiş olan “Linear Discriminant Analysis” algoritmasının, verileri sıralı gruplar halinde alarak güncellenebilen bir çeşidi, “Chunk Incremental Linear Discriminant Analysis” geliştirildi. Yine sınıflandırma için “Support Vector Machine” yönteminin yeni verileri öğrenerek güncellenmesini içeren bir algoritma hazırlandı.

Linear Discriminant Analysis, belli sınıflara ait olan yüksek boyutlu öznitelik vektör topluluklarının daha düşük boyutlu bir uzaya yansıtılması ve bu yansıtılan uzayda aynı sınıftaki vektörlerin birbirlerine yakın, farklı sınıftaki vektörlerin ise birbirlerinden uzak yansımalara sahip olmasını hedefleyen bir yöntemdir. Chunk Incremental Linear Discriminant Analysis (Chunk ILDA) (LI, 2003) ise yansıtılan bu uzayın her gelen yeni veri grubuyla tekrar tekrar sıfırdan başlanarak hesaplanması yerine sadece yeni gelen veriler için güncellenebilmesini sağlayan bir yöntemdir.

Normalde iki sınıflı sınıflandırma problemlerini çözmeyi hedefleyen SVM çoklu sınıf problemleri için de genişletilebilir. Bunun için “1 sınıf’a karşı diğerleri” yöntemi ele alındı. Bu yöntemde öğrenilecek her bir sınıf için bütün diğer sınıf örnekleri yanlış örnek olarak kabul edilir ve eğitim yapılır. N sınıf için N adet SVM modeli eğitilir. Sınıflandırmaya sokulacak bir örnek bütün SVM modellerine sokulur ve cevap olarak “bu sınıfa aittir (1)” ya da “bu sınıfa ait değildir (0 ya da -1 vb.)” kararları yerine reel sayı değerleri olan ve işaret fonksiyonuna

sokulmamış olan karar değerleri alınır. En yüksek değeri veren modelin sınıfına atama yapılır. SVM'i yeni gelen veriyle eğitmek için, bütün eğitim verisini baştan kullanmak yerine sadece önceden eğitilen SVM modellerinin sunduğu "destekçi vektörleri" (Support Vectors) yeni veriyle beraber tekrar eğitilir.

Standart olarak kullanılan "Single Kernel SVM"'e ilave olarak "Multiple Kernel SVM" ile verileri sıralı olarak işleyebilen türevi geliştirildi. Bu yöntem ile her bir yüz vektörünü daha küçük bloklara ayırarak, her bir blok için ayrı kernel oluşturuluyor. SVM'e girdi olarak verilecek kernel, bloklar için oluşturulan kernellerin ortalaması alınarak belirleniyor.

Sistem yüz tanıma yapılırken, önerdiği isimle beraber bir güvenilirlik puanı da sunuyor. Bu puan yapılan önerinin ne kadar emin olarak yapıldığının bir ölçütüdür. Bir video boyunca yapılan bütün tahminler ve güvenilirlik puanları hesaba katılarak tanıma başarısı oran olarak grafiklendirildi. Güvenilirlik puanları farklı eşik değerlerine tabi tutuldu ve her bir eşik değeri için, güvenilir kabul edilen tahminler arasındaki başarı oranları ölçüldü. İki farklı senaryo hesaba katıldı. Birincisinde video boyunca karşılaşılan yüz olarak tespit edilmiş bütün adaylar başarı oranı hesaplamasında dikkate alındı. İkinci yöntemde yalnızca yüzlerini öğrenmeyi hedeflediğimiz 6 başrol karakteri için yapılan tahminler üzerinden başarı oranları hesaplandı.

İşlem yükü ve yüz tanımada gösterilen başarı oranları dikkate alındığında Histogram of Oriented Gradients ve Nearest Neighbour çiftinin en uygun yöntem olduğu görüldü.

Bu çalışmaları özetleyen bir konferans makalesi yayınlandı (YILMAZTÜRK, 2010). Ayrıca makalenin uzun şekli Intelligent Data Analysis dergisine, bu çalışmalardan türeyen başka bir çalışmada Multimedia Tools and Applications dergisine gönderildi.

Otomatik etiketleme aracını geliştiren ikinci yüksek lisans tezinde insan yüzünün temel bileşenlerini (gözler, ağız, burun) otomatik olarak saptayan aktif görünüm modeli (AGM), local binary patterns (LBP) ve discrete cosine transform (DCT) metodlarına dayalı bir yüz tanıma yöntemi geliştirilmiştir. İnsan yüzünün temel bileşenlerinin yerleri Haar dalgacık benzeri öznitelikler kullanan Haar Peşpeşe Sınıflandırıcılar (HPS) ile saptanmaktadır. Saptanan yüzlerin cilt rengi analizi ve ağız, burun varlığı kontrolüyle, gerçek bir yüz olup olmadığı kontrol edilir. Yüzün yeri saptandıktan sonra Aktif Görünüm Modeli (AGM) ile yüze ait önemli noktalar bulunmaktadır. AGM'nin belirlediği 67 önemli nokta kullanılarak yüz belirli bir şablona oturtulur. İnsan yüzü tanımada yüksek başarım elde edebilmek için giriş resimlerinin hizalanması (alignment) gerekmektedir. Genellikle getirilen çözümler model tabanlı yaklaşımlardan oluşmaktadır. Bu model tabanlı yöntemler arasında insan yüzü hizalama işleminde en başarılı sonucu aktif görünüm modeli (active appearance model, AGM) vermektedir. Yöntem, şekil ve örüntü arasında kurduğu model sayesinde hızlı ve gürbüz bir şekilde bozulabilir (deformable) resim eşlemesi yapabilmektedir. Bu projede insan yüzüne ait önemli noktaların çıkartılması ve insan yüzüne ait modelin kurulması aşamasında AGM yöntemi kullanılmıştır. Farklı aydınlatma koşullarından etkilenmemek için bulunan yüzler historam eşitlemesi yöntemiyle belirli bir gri seviyeye getirilir. Bu ön işlemlerden sonra yüzün local binary pattern ve discrete cosine transform yöntemleriyle özellikleri çıkarılır.

Bu çalışmanın iki farklı çalışma modu vardır. İlki, eğitim modudur. Bu modda, bulunan yüzlerin LBP ve DCT Mod2 özellikleri bir dosyaya yazılır ve daha sonra çok sınıflı SVM ve Random Forest sınıflayıcılarına eğitim amaçlı verilir. Bu sınıflandırıcılar, çıkarılan özelliklerle eğitilir. İkinci mod, tanıma modudur. Bu modda, eğitilen sınıflayıcılar kullanılarak, videolarda bulunan yüzlerin kime ait olduğu saptanır. Girdi olarak kullanılan videolardan gelen görüntülerde bulunan yüzlerin önemli noktaları AGM metodu ile bulunduktan sonra yüz belirli şablona oturtulur ve sonra histogram eşitlemesi ile yüz belirli bir aydınlık değerine getirilir. Daha sonra LBP ve DCT mod2 özellikleri çıkarılır, SVM ve Random Forest sınıflandırıcılarıyla kime ait olduğu bulunur.

Farklı aydınlatma koşullarında çekilmiş birçok videoda ve farklı televizyon dizileriyle yapılan denemelerde, çalışmamızda kullanılan yöntemlerin oldukça başarılı olduğu görülmüştür. Video etiketlendirmede kullanılabilecek cinsiyet ve yaş grubu tayini yapabilen hazır modüller de bu çalışmaya entegre edilmiştir.

Bu çalışmalarımız da bir konferans makalesinde yayınlanmıştır (YAPRAKKAYA, 2010).

3.4. Videoların Sınıflandırılması ve Özetlenmesi

Projede futbol alanı dışındadiğer alanlarda da metin bilgisiyle gelen videolar üzerinde etiketlemenin nasıl yapılacağı üzerinde çalışmalar yapıldı. Bu konuda detaylı bir literatür taraması yapıldıktan sonra belgesel videolarının otomatik etiketlendirilme, sınıflandırılma ve mantıksal özetinin çıkarılması için uygun olduğu görüldü.

Video sınıflandırması, videoya, daha önceden tanımlanmış video kategorilerinden birinin atanmasıdır. Bu sınıflandırma, videodaki yazı, ses veya görsel bilgiler kullanılarak yapılmaktadır. Bu projede videoların sınıflandırılmasında iki ayrı yaklaşım denenmiştir. Birinci yaklaşım sırasıyla aşağıdaki adımları içermektedir.

İlk adımda, alt yazı dosyası analiz edilerek bu dosyadaki cümleler çıkarılmış, cümlelerin sözcük türleri (part of speech tag) atanmıştır. Bu atama için Stanford Log-linear Part-Of-Speech Tagger kullanılmıştır (TOUTANOVA, 2003). Bu cümlelerdeki “stop word”ler (örneğin; “about”, “him”) mantıksal bir anlam taşımadığından, metinden çıkarılmıştır.

İkinci adımda; altyazı dosyasındaki anahtar kelimeleri (keyword) seçmek için TextRank (MIHALCEA, 2004) algoritması kullanılmıştır. Bu algoritma uygulanarak; seçilen her kelimenin puanı hesaplandıktan sonra, puanı sıfırın üzerinde olan kelimeler anahtar kelime olarak seçilmiştir. Daha sonra anahtar kelimelerin hangi anlamlarında kullanıldığını bulmak için “word sense disambiguation” algoritması (BANERJEE, 2002) kullanılmıştır.

Üçüncü adımda; WordNet kullanılarak, anahtar kelimelerin alanları (domain)bulunmuştur. Bu domain’lerin toplam sayısı (anahtar kelimelerde kaç kere geçtiği) hesaplanmış ve domainler bu sayıya göre sıralanmıştır. Belgesellerin kategorilerinin belirlenmesinde belgesel adının önemli olduğu düşünülmüş ve belgesel adında geçen kelimelerin de WordNet domain’leri bulunmuştur. Domain’lerin toplam sayısı hesaplandıktan sonra, domain eğer belgesel adında geçiyorsa, bu sayı dörtte bir oranında artırılmıştır.

Bir sonraki adımda her bir video kategorisinin hangi WordNet domaini ile alakalı olduğu belirlenerek, anahtar kelimelerin domainine göre videoya bir kategori atanmıştır. Belgesel kategorisi olarak 14 tane kategori tanımlanmıştır ve bu kategorilerle ilgili WordNet domainleri bulunmuştur (KATSIOULI, 2007). Bu kategoriler; “Geography”, “Animals”, “Politics”, “History”, “Religion”, “Transportation”, “Accidents”, “Sports”, “War”, “Science”, “Music”, “Art”, “Technology” ve “People” kategorileridir. Videoya ait WordNet domainlerine sırayla bakılarak herhangi bir kategorinin ilk domaini ile aynı olup olmadığına bakılmıştır. Aynı olması durumunda kategori atanmış, iki kategoride aynı olması durumunda kategorilerin ikinci domainlerine bakılmıştır. Hiçbir kategorinin ilk domaini ile aynı olmaması durumunda videonun sıradaki WordNet domainine bakılarak kategori atanması yapılmıştır.

Bu yaklaşımın test edilmesinde 40 belgesel kullanılmış ve %75 başarı elde edilmiştir.

İkinci yaklaşımda kategorisi bilinen videolarla öğrenme gerçekleştirilmiş ve öğrenilen bilgilerle sınıflandırma yapılmıştır. Yaklaşım öğrenme ve sınıflandırma adımlarından oluşmaktadır. Öğrenme aşamasında her bir kategorinin genel domain dağılımı öğrenilmiştir. Öğrenme daha önce tanımlanan 14 kategori için yapılmıştır. Her kategoriye ait altyazı dosyaları birinci

yaklaşımdaki gibi işlenmiş ve böylece altyazı dosyalarının domain'leri ve bu domain'lerin sayıları belirlenmiştir. Kategorinin domain dağılımını belirlemek için, her bir domain'in Tf*Idf değeri hesaplanmıştır. Bu değer herhangi bir domain'in diğer tüm domain'lere oranını vermektedir. Böylelikle her bir kategorinin domain Tf*Idf değerlerinden oluşan bir matris elde edilmiştir

Sınıflandırma aşamasında bir belgeselin sınıflandırılması için daha önce öğrenilen bilgiler kullanılmıştır. Bir belgeseli sınıflandırmak için; belgesele ait altyazı dosyası birinci yaklaşımda anlatıldığı şekilde işlenmiş ve böylece altyazı dosyasının domain'leri ve bu domain'lerin sayıları belirlenmiştir. Belgeselin domain dağılımını belirlemek için, her bir domain'in Tf*Idf değeri hesaplanmıştır. Öğrenme aşamasında elde edilen matris kullanılarak, bu belgeselin domain dağılımıyla en çok benzeşen kategori domain dağılımı bulunmuştur. Bu benzerliğinin bulunmasında "Cosine Similarity" kullanılmıştır.

Bu yaklaşımın test edilmesinde 40 belgesel kullanılmıştır. Bunlardan 22 tanesi öğrenme için kullanılmış, diğer 18 tanesi sınıflandırılmıştır ve % 77 başarı elde edilmiştir.

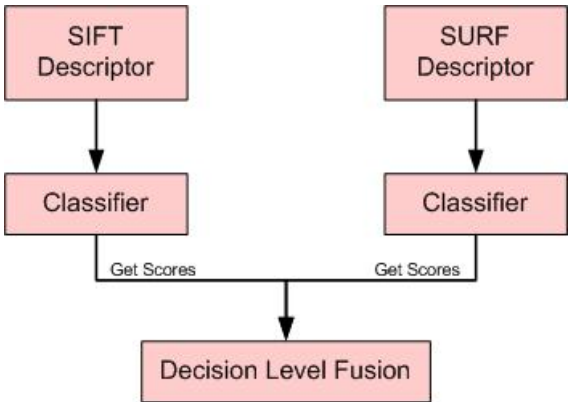
Videoların özetlenmesi, videoyu mantıksal olarak özetleyecek şekilde video kesitlerinin seçilmesi şeklinde tanımlanabilir. Bu çalışmadaki video özetlenmesi de video sınıflandırılmasında olduğu gibi alt yazı dosyaları kullanılarak yapılmıştır. Bu amaçla önce altyazı dosyasındaki cümleler çıkarılmış ve bu cümlelerin sözcük türleri Stanford Log-linear Part-Of-Speech Tagger kullanılarak atanmıştır. Bu cümlelerden videoyu özetleyecek önemli cümlelerin seçilmesi için TextRank algoritması kullanılmıştır. Özetleyen cümleler bulunduktan sonra, bu cümlelerin altyazı dosyasındaki başlangıç ve bitiş zamanları belirlenmiş ve bu zamanlara karşılık gelen video kesitleri arka arkaya gösterilmek suretiyle video özeti oluşturulmuştur.

Bu konuda yapılan çalışmalar bir konferans ve bir dergi makalesi olarak yayınlanmıştır(DEMİRTAŞ, 2010a; DEMİRTAŞ, 2010b). Çalışmaların detayları ayrıca bir yüksek lisans tezinde anlatılmaktadır (DEMİRTAŞ, 2009).

3.5. Nesne tanıma

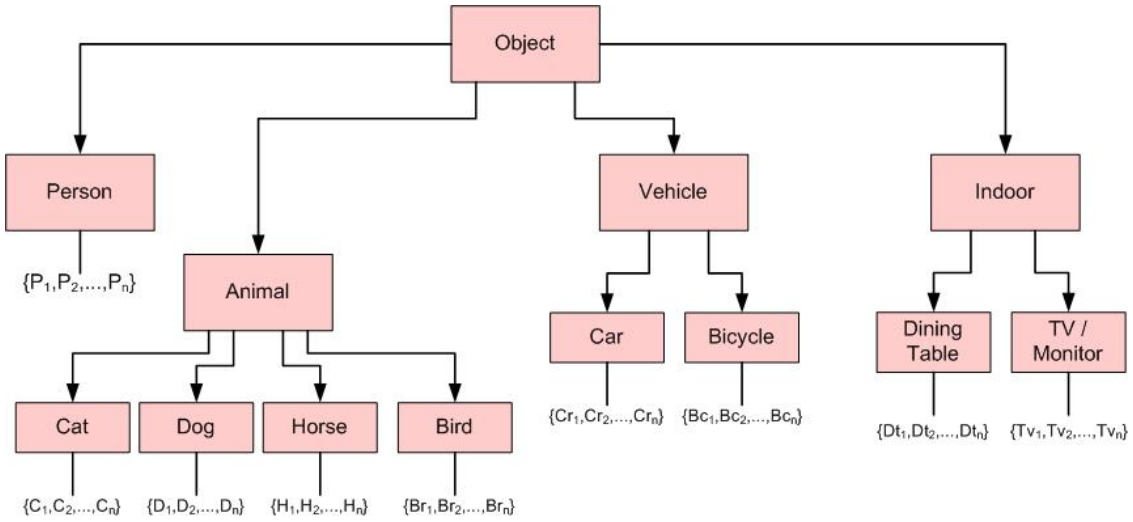
Videoların (Yarı-) otomatik anlambilimsel olarak etiketlenmesikapsamında insan yüzlerine ek olarak belirli nesneleri de otomatik tanıma ve ontolojik olarak etiketleme çalışmaları yapıldı. Bu amaçla projenin son döneminde bir yüksek lisans tezi kapsamında kişisel videolarda nesne takip etme, tanıma ve anlambilimseletiketleme (semantic recognition) çalışmaları yürütülmüştür.

Bu çalışmada genel bir nesne tespit edici kullanılmaktadır (ALEXE, 2010).Nesne tespit edici, her bir çerçevede olası nesneleri kutu içinde işaretler. Bu kutu içindeki alanlar için nesne hiyerarşine göre tanıma işlemi gerçekleştirilmektedir. Tanıma işlemleri SIFT(Scale-Invariant Feature Transform**)** (LOWE, 1999) ve SURF(Speeded Up Robust Features) (BAY, 2008) öznitelik tanımlayıcıları yardımıyla gerçekleştirilmektedir. SIFT ve SURF öznitelik tanımlayıcılarından gelen skorlar karşılaştırılarak daha yüksek oranlı sonuç veren öznitelik tanımlayıcısının tanıma işlemi sonucu kullanılmaktadır (Bkz. Şekil 4).



Şekil 4: Öznitelik tanımlayıcılarının kullanımı

Çalışmada kullanılan nesne hiyerarşisi Şekil 5'de gösterilmiştir. Bu kapsamda ilgilenilennesneler kişi, kedi, köpek, at, kuş, araba, bisiklet, masa ve TV/ekrandır.



Şekil5 : Nesne Hiyerarşisi

Tanıma işlemi gerçekleştirildikten sonra bu nesnenin kime ait olduğu bilgisi elde edilmeye çalışılmaktadır. Kime ait olduğu bilgisi nesne hiyerarşisinde en alt seviyede bulunan elemanlardır. Örneğin kedi için C1, C2 ve C3 farklı kişilere ait kedileri temsil etmektedirler. Bu seviyede ColSIFT öznitelik tanımlayıcısı kullanılmaktadır.

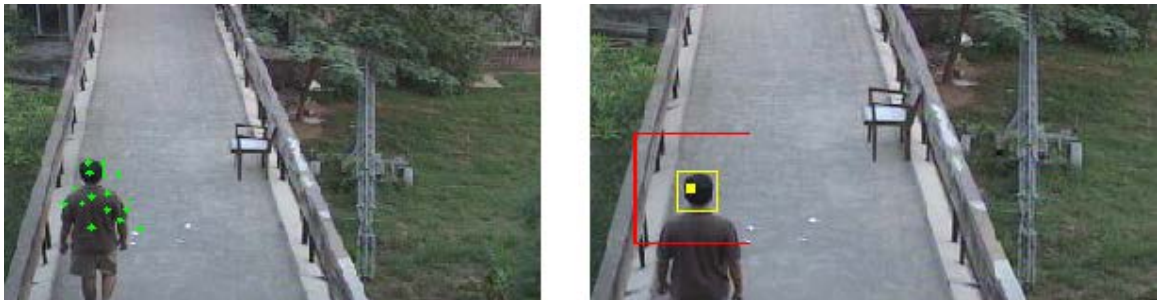
Şu an bu kapsamda geliştirilen algoritmalar TRECVID (SMEATON 2006) ve PASCAL (EVERINGHAM, 2008) veri setleri ile denenmektedir. TRECVID veri seti proje kapsamında video veri seti olarak kullanılmakta; PASCAL veri seti ise nesne tespiti için kullanılan genel nesne tanımlayıcının(ALEXE, 2010)eğitimi için kullanılmaktadır. Bu tez çalışmasının sonuçlarının Haziran 2011'de tamamlanması beklenmektedir.

3.6. İnsan tanıma

Proje kapsamında sürdürülen ve projenin son döneminde başlayan başka bir tez çalışmasında da güvenlik kameralarından elde edilmiş videoların işlenerek insanların dahil olduğu olaylar ya da aktivitelerin otomatik olarak tanımlanması hedeflenmektedir. Bu problemi üç ayrı süreç altında çözmeyi planlıyoruz: İnsan tanıma, İnsan takibi ve olay/aktivite tanıma. Tanımlanan her bir süreç Bilgisayarlı Görme literatüründe ayrı birer çalışma

sahasıdır. Tez çalışması süresince ilk olarak problemin çözümüne yönelik bu sahalarda doğruluğu/performası kanıtlanmış tekniklerin tespiti, ikinci olarak bu teknikleri kullanarak bir uygulama geliştirilmesi hedeflenmektedir.

Şu ana kadar tez çalışması kapsamında insan tanıma ve insan takibi konuları üzerine literatür taramaları yapılmış ve tarama sonucu elde edilen yayınlar analiz edilip ilgili tekniklerin avantaj ve dezvantajları değerlendirilmiştir. İnsan tanıma ve insan takibi ile ilgili OpenCV kütüphanesi kullanarak Optik Akış, Şablon Eşleştirme (Bkz. Şekil 6), Kalman filtresi ile nesne takibi, HOG tanımlayıcısı ile insan tanıma konularında örnek kodlamalar yapılıp ilgili tekniklerin pratik olarak zayıf/güçlü yönleri gözlemlenmiştir. Ayrıca ilgili makaleler ile birlikte paylaşılan kodlar incelenmiştir. Şu ana kadar insan takibi yerine daha geniş bir küme olan hareketli nesne takibi üzerine çalışılmıştır. İnsan tanıma ile ilgili kodlamaların tamamlanmasıyla birlikte önce insanın tanınıp sonra o insanın otomatik takip edilmesi çalışmaları güncel olarak devam etmektedir.



Şekil 6. Optik Akış ve Şablon Eşleştirme ile ilgili örnek resimler

İnsan tanıma ve insan takibi konuları Bilgisayarlı Görme dünyasında olgunlaşmış konulardır ve literatürde bu konular üzerine yüzlerce yayın bulunmaktadır. Ancak video'dan olay tanıma, insan aktivitesi tanıma gibi üst seviye anlamsal bilgi çıkarımı sahaları henüz olgunlaşmamıştır. Bu kapsamda olay tanıma ile ilgili olarak literatür taraması devam etmektedir. Kodlama aşamasına henüz geçilmemiştir.

4. Anlambilimsel Sorgulama

Geliştirilen sistemde videoların kendileri ile birlikte içeriklerini tanımlayan yardımcı veriler ontolojik olarak tanımlanmakta ve saklanmaktadır. Bu bilgilerin detaylı ve kolay bir biçimde sorgulanabilmesi gerekir. Bu projede video arşivinin detaylı ve anlambilimsel olarak sorgulanması için değişik sorgulama yaklaşımları denenmiş, form tabanlı, kelime bazlı ve doğal dil (İngilizce) ile sorgulama yapabilen sistemler geliştirilmiştir. Bu kısımda anlambilimsel sorgulama konusunda yapılan çalışmalar özetlenmiştir.

4.1. Form-tabanlı uzay-zamansal sorgulama

Projede geliştirilen sistemde, form tabanlı bir arayüz üzerinden kullanıcının oluşturduğu sorguların analiz edilmesi, işlenmesi ve eşleşen sonuçların gösterilmesi mümkün olmaktadır. Geliştirilen arayüz ile alan ontolojilerine özgü kavramlar kullanılarak video içeriği üzerinde

- Ontolojik kavramsal sorgulama,
- Uzay-zamansal sorgulama,
- Birleşik sorgulama (Birden fazla sorgu tipinin birleştirilmesiyle oluşturulan sorgu) kabiliyetleri gerçekleştirilebilmektedir.

Ontolojik kavramsal sorgulama, alan ontolojilerine ait kavramları ve bu kavramlardan oluşturulan nesneleri kullanarak oluşturulan sorguları içermektedir. Bu sorgu tipi sayesinde ilgilenilen nesnelerin, insanların, olayların, ya da ontolojide tanımlanmış kavramların bulunduğu/göründüğü video parçaları listelenebilir. Örneğin, "Başbakanın göründüğü sahneler", "Bir kadının göründüğü sahneler" ve "araba kazasının olduğu sahneler" tarzındaki sorgular ontolojik olarak oluşturulabilir.

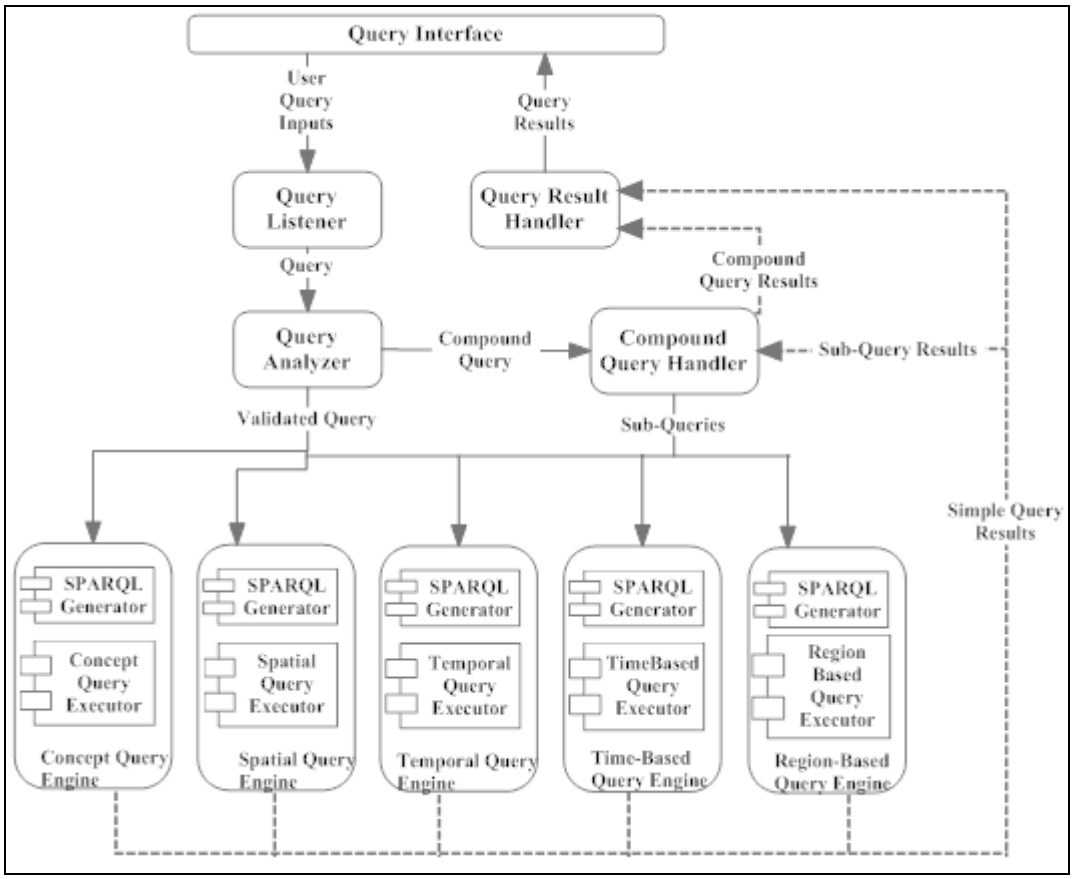
Uzaysal sorgulamalar, *uzaysal ilişkilere dayalı* sorgulama *ve bölgesel* sorgulama olmak üzere ikiye ayrılmaktadır. Uzaysal ilişkilere dayalı sorgulamalarda nesnelerin/insanların birbirlerine göre uzaysal ilişkileri sorgulanır. Örneğin, "Ali'nin arabanın solunda olduğu sahneler" gibi sorgular oluşturulabilir. Bölgesel sorgulamalarda ise videonun oynatılacağı alan üzerinde dikdörtgensel bir bölge seçilerek ilgili bölgeye yönelik sorgulamalar yapılabilmektedir. Bölgesel sorgu işleme sonucunda sorgu ara yüzü üzerinden seçilen belirli bir nesnenin ya da alan ontoloji kavramının ilgili bölge içerisinde yer aldığı sahneler kullanıcıya dönülür. Örneğin "Bir oyuncunun ekranın sağında göründüğü sahneler" bu tür bir sorgulamadır.

Zamansal sorgulamalar, *zamansal ilişkilere dayalı* sorgulama ve *zaman tabanlı* sorgulama olmak üzere ikiye ayrılır. Zamansal ilişkilere dayalı sorgulamalarda nesnelerin ve kavramların birbirlerine göre zamansal ilişkileri sorgulanır. Örneğin, "Hırsızın polisten önce göründüğü sahneler", "Bir hayvanın bir insanla aynı anda göründüğü sahneler" gibi sorgular oluşturulabilir. Ayrıca, zamansal sorgulama olaylar için ve olaylara göre de yapılabilmektedir. Örneğin, "Araba kazasından sonra kavga olan sahneler" gibi bir sorgu oluşturulup işlenebilmektedir. Zaman tabanlı sorgularda ise sorgu ara yüzünden girilen bir zaman ya da zaman aralığı girdisi alınır ve girilmiş olan bu zaman bilgilerinden önceki, sonraki ya da girilen zaman aralığındaki olaylar, nesneler ve kavramlar sorgulanabilir ve ilgili sahneler kullanıcıya dönülür.

Birleşik sorgu kabiliyeti, sistemde ontolojik kavramlara yönelik sorguları ve uzay-zamansal sorguları bir araya getirip birleştirerek sorgulama yapmayı sağlamaktadır. Sistemde desteklenmekte olan ontoloji tabanlı kavramsal, uzay-zamansal, bölgesel ve zaman tabanlı sorgu tipleri "AND" / "OR" mantıksal bağlaçlarıyla birleştirilerek geçerli birleşik sorgular oluşturulur. Elde edilen birleşik sorgu, ilk olarak bir ön işlemeden geçirilir ve oluşturulan sorgu ağacı üzerinde sonuç kümesini daraltacak şekilde düzenleme yapılır. Sorgu ağacının dalları üzerinde ilk önce "AND" bağlacıyla bağlı alt sorgular işlenecek şekilde bir yapı oluşturulur. Daha sonra sorgu ağacı üzerindeki alt-sorgular ayrı ayrı işlenir ve ilgili "AND" / "OR" mantıksal bağlacına göre alt-sorgu sonuçları birleştirilir. Ana sorgu ağacı üzerinden ilk önce "AND" bağlacıyla bağlı alt-sorgular işlenip sonuçları birleştirileceği için daha küçük bir sonuç kümesi üzerinde çalışılır. Çünkü "AND" ile bağlı olan sorgu sonuçlarının kesişim kümesi alınırken "OR" ile bağlı kümenin birleşim kümesi alınmaktadır. Bu nedenle ilk önce "AND" ile bağlı sorgular birleştirilerek sonuç kümesinin olabildiğince küçük kalması hedeflenmiştir. Alt-sorguların sorgu tiplerine göre ayrı ayrı işlenip birleştirilmesiyle ana sorgunun sonucu adım adım elde edilmektir.

Sistemde, tüm sorgu tipleri için hem genel sorgulama, hem de videoya özel sorgulama özelliği desteklenmektedir. Oluşturulan sorgu için ara yüz üzerinden herhangi bir video seçilmezse, sorgu sonuçları bütün veritabanı üzerinde aranıp ilgili tüm videolardan sahneler gösterilir. Ancak, özel bir video seçilerek, ilgili sorgunun sadece o videoya ait verilerin üzerinde çalıştırılması da sağlanabilir ve sadece belirtilen videodaki sorguya eşleşen sahneler gösterilir. Bu sayede, kullanıcıya video veritabanındaki tüm videolar üzerinde arama ve veri erişimi kabiliyeti sağlanmasının yanı sıra sadece belirli bir video üzerinde arama yapma kabiliyeti de sunulmuş olur. Kullanıcı sadece belirli bir video ile ilgileniyorsa, istediği verilere daha hızlı bir şekilde ulaşmış olur ve birden fazla videodan gelen farklı sahneler arasında ayrıştırma yapma işlemleriyle de uğraşmamış olur.

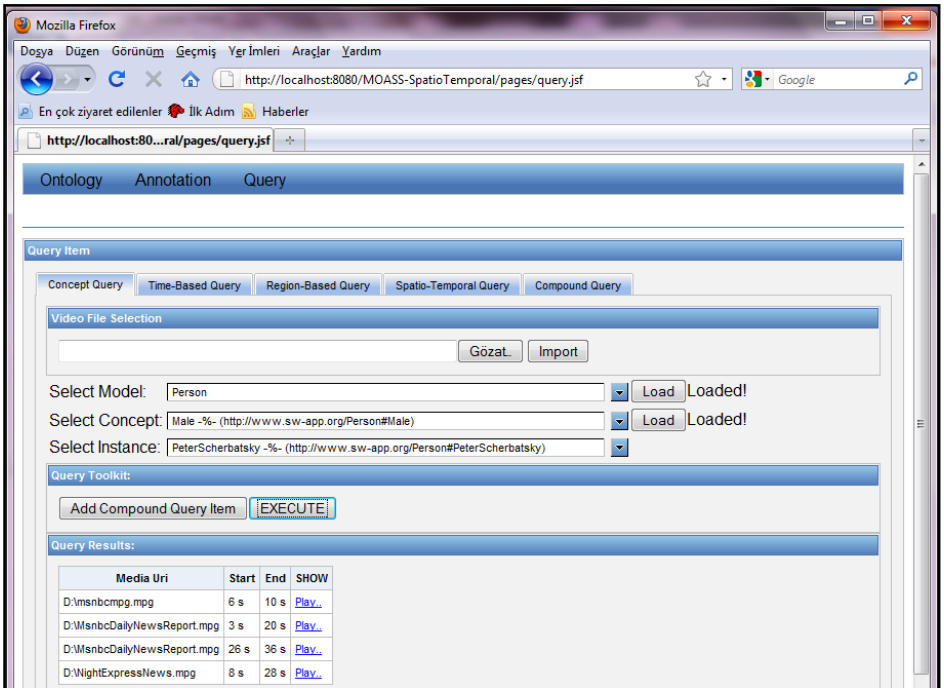
Sorgulama mekanizması, sorgu oluşturma ve sorgu işleme kısımlarından oluşmaktadır. Sorgu oluşturma kısmında form tabanlı bir kullanıcı ara yüzünden alınan sorgu verileri analiz edilerek SPARQL sorgusu oluşturulur ve oluşturulan SPARQL sorgusu sorgu tipine göre ilgili sorgu işleyicisine gönderilir (bkz. Şekil 7).



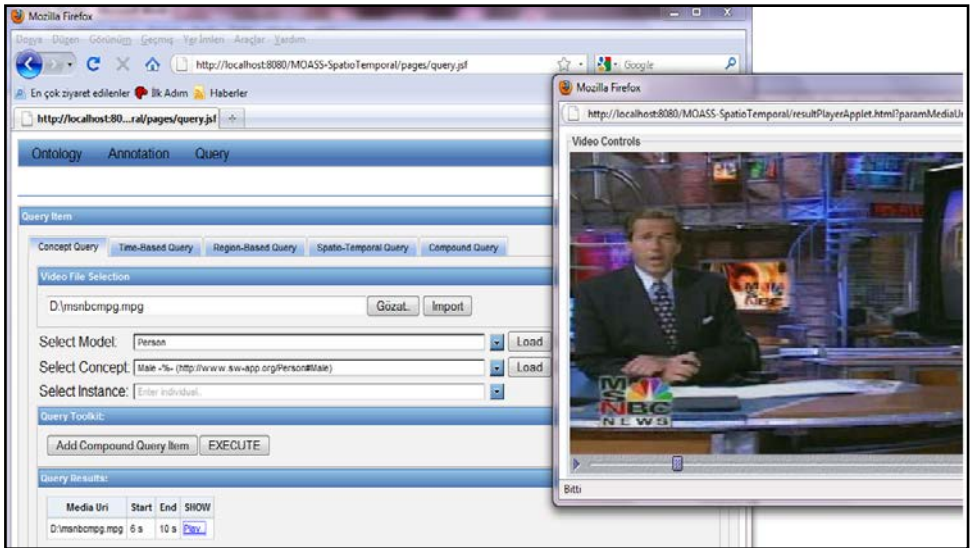
Şekil 7. Sorgu işleme mimarisi

Sistemde sorgu işleyicileri girdi olarak bir SPARQL sorgu cümlesi alır ve bu sorguyu işleyerek eşleşen sorgu sonuçlarını döndürür. Bu nedenle, sorgu oluşturma ve sorgu işleme kısımları birbirine bağımlı değildir. Bu sayede, sistemin sorgu işleme modülü, ilgili form tabanlı ara yüzden ve sorgu oluşturma kısmından ayrıştırılıp SPARQL sorgu cümlesi oluşturan başka bir sisteme bağlandığında da modüler olarak çalışabilmektedir. Örneğin, doğal dille sorgu arayüzü kullanılarak girilen doğal dil cümlelerini SPARQL sorgu cümlelerine dönüştüren bir sistem, geliştirdiğimiz sistemin sorgu işleme mekanizmasıyla entegre bir şekilde çalışabilmektedir. Aynı şekilde, sistemdeki manüel video etiketleme mekanizması da MPEG-7 ontolojik yapısına uyumlu etiketleme yapan otomatik etiketleme modülüyle değiştirilerek yarı-otomatik veya otomatik etiketleme yapan bir yapıya dönüştürülebilir. Geliştirilen sistemin bir başka avantajı da, MPEG-7 ontoloji ve alana özgü ontoloji entegrasyonu detaylarının otomatik olarak yapılması ve kullanıcıya görünür olmamasıdır. Kullanıcı etiketleme ve sorgulama işlemleri sırasında sadece alana özel ontolojileri ve onlara ait kavramları görmekte ve kullanmaktadır. Böylece, MPEG-7 ontoloji yapıları ve detaylarıyla uğraşmamaktadır.

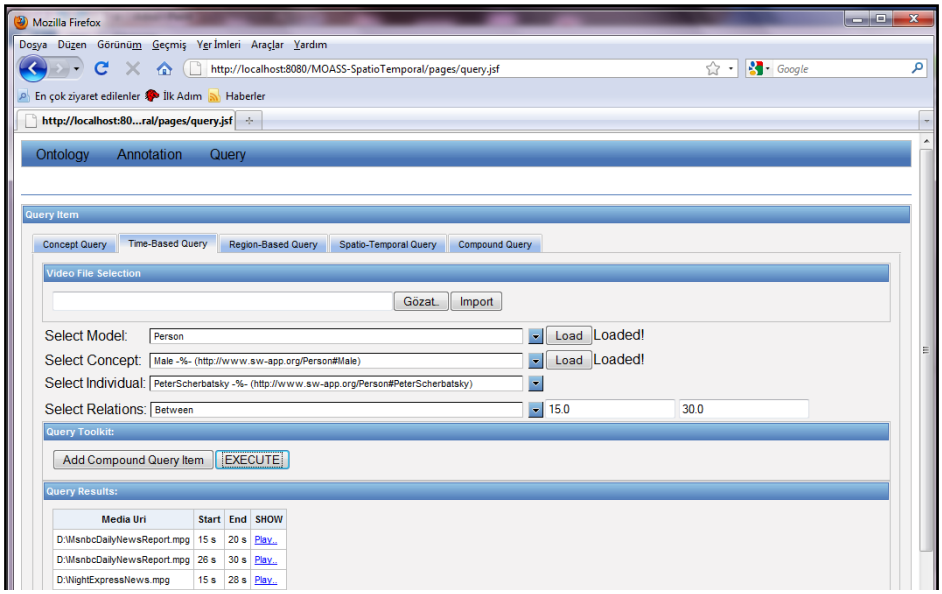
Şekil 8 ve Şekil 9'de sistemin form tabanlı arayüzünden kesitler verilmiştir. Bu şekillerde kavram sorgulamaları örneklendirilmiştir. Sorgular belirli kişi bazında olabileceği gibi kişilerin ait olduğu sınıflar bazında da sorulabilmektedir. İlk şekilde Peter Scherbatsky sorgulanmakta, ikinci de ise herhangi bir erkeğin göründüğü sahneler sorgulanmaktadır. Şekil 10 ise aynı kişinin videonun belirli zamanlarında görünüp görünmediğini sorgulamaktadır.



Şekil 8. Sorgu arayüzü



Şekil 9. Sorgu ve gelen yanıt



Şekil 10. Zamansal sorgulama

Bu çalışmaların detayları bir yüksek lisans tezinde anlatılmaktadır (DEMİRDİZEN, 2010).

4.2. Kelime-tabanlı ontolojik sorgulama

Form-tabanlı sorgulama her ne kadar detaylı aramalara izin veriyor olsa da, kullanıcıların belirli sorgu formlarını doldurmaları gerekmektedir. Kullanıcıların büyük bir kısmı Google tarzı kelime-tabanlı aramaya alışık olduğu için, aynı işlevselliği kelime tabanlı sorgulama yöntemi ile elde etme çalışmaları yaptık. Bu yöntem ile kullanıcılar çok daha rahat biçimde sorgu üretebilmektedirler. Örneğin, "Arda'nın Alex'e yaptığı faulller" 'i arıyorsak, sorgu satırına "faul arda alex" yada "arda alex faul" yazmak yeterli olacaktır. Bu yöntemin tek kusuru bazı sorguların anlam belirsizliğine (ambiguity) neden olmasıdır. Örneğin, yukarıdaki sorgular fazlalık olarak "Alex'in Arda'ya yaptığı faulller" 'i de getirebilir.

Bu proje kapsamında kelime-tabanlı arama yöntemini geliştirip zenginleştirmek için, ontoloji ve anlamsal çıkarımdan (inference) faydandık. UEFA ve SporX sitelerindeki maç anlatımlarının sisteme yüklenip aranabilir hale gelmesi için aşağıdaki adımlar izlendi.

1. "Web Crawling" yöntemiyle sözü geçen sitelerdeki maç bilgileri ve anlatımlar önce XML'lerde daha sonra Lucene index'lerinde saklandı.
2. Web'den çekilen bu bilgi ontoloji instance'ları oluşturmak için kullanıldı. Bu bilginin içinde maça ait oyuncular, takımlar, hakemler, goller, kartlar gibi temel bilgiler yer almaktadır.
3. Bu temel bilgilerin yanında maç anlatımları "doğal dilden bilgi çıkarma" yöntemiyle ontolojiye uygun olarak işlenerek daha karmaşık bilgiler çıkarıldı. (Örn: köşe vuruşu, faul, vs)
4. Çıkarılan her bilgi, bir ontoloji nesnesi olarak OWL dosyalarında saklandı. Burada, tüm maça bir OWL dosyası dersek, maçın eylemleri de ontoloji objeleri (instance) olmaktadır.
5. Bu aşamada oluşan OWL dosyaları anlamsal sorguya hazırdır. Daha önce bahsedilen SPARQL ya da form-tabanlı sorgulama aracı bu OWL dosyaları üzerinde arama yapabilmektedir.
6. Kelime-tabanlı arama yapabilmek için ise, maç eylemleri, Lucene[1] kullanılarak indekslendi. Bunun için, OWL dosyalarındaki bilgiler otomatik olarak okunup, her

[1]Apache Lucene. http://lucene.apache.org

OWL objesi bir index entry'sine denk gelecek şekilde indexlendi. Anlamsal aramalara izin verebilmek için eylemlerin ontolojik bilgileri de index'e eklendi. Ontolojik bilgiler, o eylemin sınıfı ve diğer özellikleri (property) hakkında bilgi vermektedir. Sorgu başarısını yükseltmek amacıyla index'in belirli field'ları aramalarda ön plana çıkarıldı.

7. Anlamsal çıkarımın (inference) sorgu başarısına etkisini gözlemlemek amacıyla, bir önceki adımda oluşan OWL dosyaları anlamsal çıkarım işleminden geçirildi. Bunun için açık kaynak kodlu Pellet[2]yazılımı kullanıldı. Oluşan OWL dosyaları 6. Adımda anlatıldığı gibi indexlendi. Anlamsal çıkarım sonucunda elde edilen bilgiler de index'te uygun field'lara eklenerek aranabilir içerik zenginleştirildi.

Performans kıyaslaması amacıyla, sadece düz metin kullanılarak ayrıca bir index oluşturuldu. Elde edilen tüm indeksler 10 farklı sorguyla test edildi. Doğal dilden bilgi çıkarma yönteminin ve anlamsal çıkarımın sorgu performansına etkileri gözlemlendi. Yapılan testlerde bazı sorguların anlamsal çıkarım olmadan cevaplanamadığı ortaya çıktı. Benzer şekilde, doğal dilden bilgi çıkarma yönteminin sorgu kalitesini artırdığı görüldü. Bu çalışmanın detayları bir konferans makalesinde anlatılmıştır (KARA, 2010).

Daha sonra, önerilen bu yöntemi başka yöntemlerle karşılaştıran çeşitli deney ve analizler yapılmıştır. İlk etapta aynı başarının sorgu genişletme yöntemleriyle de elde edilip edilemeyeceğini görmek gerekiyordu. Bu yüzden domain bilgisi kullanarak sorgu genişletme yapabilen bir modül geliştirildi ve bu önerilen yöntemimizle karşılaştırıldı. Sorgu genişletmeye örnek verecek olursak: "goal" kelimesini barındıran sorgular, "score", "miss" ve bunların türevleriyle, "punishment" sorgusu ise, "yellow card" ve "red card" gibi alt-sınıflarla genişletilmiştir. Genişleyen sorgular doğrudan free-text üzerinde çalıştırılmıştır. Yapılan karşılaştırmalarda, bu yöntemin sorgu perfomansını bir miktar artırdığı, fakat anlamsal indeksleme metoduna yaklaşamadığı gözlemlenmiştir.

İkinci etapta belirsizlikler (ambiguities) ile ilgili analizler yapıldı. Belisizlikler kelime-tabanlı sorgulama yöntemlerinin en büyük sorunudur. Genel olarak iki çeşit belirsizlikten söz edebiliriz: sözcüksel (lexical) ve yapısal (structural) belirsizlikler. Sözcüksel belirsizlikler eşsesli kelimelerden kaynaklanmakla beraber yapısal belirsizlikler bir cümlenin birden fazla anlamı olduğu durumlarda ortaya çıkar. Sözcüksel belirsizlikleri çözebilmek için word disambiguation yöntemleri gereklidir. Ancak yapısal belirsizlikler anlamsal indeksleme sayesinde çözülebilir. Örnek olması amacıyla, mevcut uygulamamıza sözcük takımı desteği ekleyerek basit yapısal belirsizlikleri çözmeye çalıştık.

Yapısal belirsizliklere örnek olarak, "foul Alex Ronaldo" sorgusu verilebilir. Bu sorguda kimin kime faul yaptığı sistem tarafından anlaşılamamaktadır. Bu sorunu çözmek için, "to X", "by X" ve "of X" gibi sözcük takımlarıyla indeksi genişlettik. Bunun için, indekse iki yeni field ekledik: bir adet özne için ve bir adet nesne için. Bu field'lere öznenin veya nesnenin ismi ile buna karşılık gelen edat birleştirilerek konulmuştur.

Yeni indeksin performansı eskisiyle karşılaştırıldı. Bunun için, 3 adet yapay sorgu hazırlandı. İlk sorgu, özneyi belirleme performansını ölçmektedir (Faul'ü yapan oyuncu daniel). İkinci sorgu, ilk sorguya bir nesne ekleyerek belirsizlik yaratmaktadır. Üçüncü sorguda özne ile nesnenin yerleri değiştirilmiştir. Eski index belirsizlikleri çözmede zorlanmaktadır; sürekli aynı oyuncuyu özne olarak seçmektedir. Fakat yeni index her durumda belirsizlikleri başarılı bir şekilde çözmektedir.

Kelime tabanlı ontolojik sorgulama kapsamında yaptığımız bu çalışma ve analizler bir yüksek lisans tezinin kapsamındadır (KARA 2010). Tezin içeriği bir dergi makalesi olarak anlatılıp Information Systems dergisine gönderilmiştir. Makale halen değerlendirme aşamasındadır.

[2]Pellet: The Open Source OWL DL Reasoner. http://clarkparsia.com/pellet/

4.3. Doğal Dille sorgulama

Projede, sisteme doğal dille sorgulama kabiliyeti ekleyen bir arayüz geliştirilmiştir. Temel olarak amaç MPEG-7 tabanlı alan ontolojilerini anlamsal ve uzay-zamansal sorgulamak için bir doğal dil (İngilizce) kullanmaktır. Geliştirilen video bilgi yönetim sisteminde sorgulanan temel ontoloji, alan ontolojilerini Rhizomik MPEG-7 ontolojisine ekleyerek oluşturulur. Kullanıcı sistemde kavramsal, uzaysal, zamansal, nesnesel yörünge ve yönsel yörünge sorguları yapabilmektedir. Kavramsal sorguda, kullanıcı ilgilendiği nesneleri sorgulayabilir. Bununla birlikte, uzaysal sorgu tipi kullanılarak nesnelerin birbirine göre yerleri (sağ, sol, vs.) sorgulanabilir. Kullanıcı, nesnelerin birbirine göre görünüş sırasını sorgulamak için zamansal (önce, sonra, vs.) sorgu tipini kullanır. Nesnelerin yörüngelerini ve belirli bir yöne giden (doğu, batı, güneydoğu, vs.) nesneleri bulmak için ise, sisteme yörünge sorgusu yöneltir. Bütün bu sorgu tipleri doğal dille sorulduğunda cevap olan video görüntülerini döndüren bir modül geliştirildi.

Kullanıcı sisteme doğal dilde bir sorgu cümlesi girdiğinde, bu cümle link ayrıştırıcı kullanılarak çözümlenmektedir (EROZEL, 2008). Sorgu tipine göre, sistemde önceden tanımlı olan kurallar kullanılarak, sorgu cümlesinden nesneler, nesne özellikleri, uzaysal, zamansal, ve yörüngesel ilişkiler ile zaman bilgileri çıkarılır. Çıkarılan bilgiler kullanılarak, doğal dil sorgu cümlesi, SPARQL sorgu cümlesine çevrilmektedir. Daha sonra SPARQL sorgusu ontoloji üzerinde çalıştırılıp, elde edilen sorgu sonuçları, nesneler arasında uzaysal, zamansal, ve yörüngesel ilişkiyi hesaplamak için kullanılmaktadır. Sorgu cümlesinde sorulan ilişkiyi sağlayan sonuçlar kullanıcıya gösterildikten sonra kullanıcı, sorgu sonuçlarını kullanarak video içeriği üzerinde gezinebilmektedir.

Geliştirilen bu sisteme kompleks nesne sorgusu da eklenmiştir. Kullanıcı kompleks nesne sorgusunu kullanarak, “VE” ve “VEYA” ile bağlanmış birden fazla nesneye ait bilgileri sorgulayabilir. Ayrıca, kullanıcının negatif anlam taşıyan sorgu yapabilmesini sağlamak amacıyla, doğal dil sorgu cümlesindeki negatif anlamlar da tespit edilebilmektedir. Böylece kullanıcı, bir nesnenin videoda göründüğü zaman aralıklarını sorgulayabildiği gibi, görünmediği aralıkları da sorgulayabilir.

Sistemdeki zamansal sorgu tipi, zaman filtresi eklemek suretiyle geliştirilmiştir. Zaman filtresi kullanarak, iki nesne arasındaki “en az 5 dakika/saniye önce/sonra”, “en fazla 5 dakika/saniye önce/sonra”, “5 dakika/saniye önce/sonra” ve “5 ile 10 dakika/saniye önce/sonra” zamansal ilişkileri sorgulanabilmektedir. Yönsel yörünge sorgusu ise, “aşağı”, “yukarı”, “sağa”, ve “sola” ilişkilerini destekler hale getirilmiştir. Bu kapsamda yapılan çalışmalardan bir yüksek lisans tezi tamamlanmıştır (ALACA-AYGÜL, 2010). Ayrıca bir dergi makalesi yazılmış ve Knowledge-based systems dergisine gönderilmiştir. Makale halen değerlendirme aşamasındadır.

5. Kullanıcı tercihlerine göre içerik bulma

İçerik kişiselleştirme uygulamaları varolan veri miktarı ile birlikte artmaktadır. Önem kazanan bu uygulama alanlarından biri de çokluortam içeriğinin kişiselleştirilmesidir. Bu çalışma alanı makine öğrenme ve istatistiksel veri analizi yöntemleri uygulanarak kullanıcıların içerik arama uzayını daraltmayı ve kullanıcıya kendi geçmişi göz önünde bulundurularak kendisi ile en fazla ilgili olan içeriğe ulaşmasını amaçlamaktadır. Bu, kullanıcıların geçmiş tercihlerinin elde edilmesini ve kullanıcı modellerinin veri üzerinden öğrenilerek yaratılmasını gerektirir.

Projede bu konuda beş yüksek lisans tezi çalışması yapıldı. Bu tezlerden dört tanesi tamamlandı (ÖZBAL, 2009; KARAMAN, 2010; ERYOL, 2010; ÖZTÜRK, 2010), beşincisi de tamamlanma aşamasındadır. Kişiselleştirme ve tercihe göre öneri sistemleri çok popüler araştırma konuları olduğu için literatürde (genelde aynı veri seti üzerinde) çok farklı

yöntemler denenmektedir. Projede bu konuda farklı öğrencilerle farklı boyutlarda araştırmalar yapıldı ve değişik yöntemler geliştirildi. Edinilen tecrübelere göre projede öngörüldüğü şekilde ontoloji-tabanlı kullanıcı tercihlerine göre içerik bulma yöntemini son tez kapsamında tamamlama aşamasındayız. Aşağıda bu konuda yaptığımız araştırma ve geliştirme çalışmaları özetlenmiştir.

5.1. Melez sosyal önerisistemleri için olasılıksal saklı anlam analizi

Projede ilk etapta işbirlikçi (collaborative) ve içerik-bazlı (content-based) öneri sistemlerinin birleşiminden oluşan melez (hybrid) yöntemler üzerinde çalışıldı. İşbirlikçi yöntemler, aynı filmlere yakın düzeyde oy veren kullanıcıların birbirine yakın olduğu öngörüsü üzerine kurulur. Bu sistemlerde iki kullanıcının birbirine benzerliği hakkında bilgi elde etmeye çalışılırken, bu iki kullanıcının çok sayıda izlediği ortak filmin bulunması gerekir. Veri seyrekliği nedeniyle bu bilgi her zaman elde edilememektedir. Seyreklik problemine parallel olarak, popular filmlerin en çok ortak izlenen filmler olmaları sebebiyle sistem az sayıdaki popüler filmi tekrar tekrar önerecektir. Bir diğer problem de soğuk başlama problemidir. Bir kullanıcı sisteme dahil olduğunda izlediği filmler sistemdeki bir başka kullanıcının izlediği filmlerle örtüşmedikçe, bu kullanıcı sistemden tatmin edici öneriler alamayacaktır.

Tüm bu problemler, ilişkisel bilginin sadece ‘ortak izlenen filmler’ verisinden hareketle elde edilmeye çalışılmasından kaynaklanmaktadır. Ortak izlenen filmler verisi, filmlerin içeriği hakkında daha detaylı bilgi elde edilerek desteklenebilir. Bu sayede yukarıda bahsedilen seyreklik, popüler öneriler ve soğuk başlama problemlerinin önüne geçilebilmektedir. İçerik bazlı yöntemler sayesinde zor elde edilen ‘ortak izlenen filmler’ verisi yerine ‘benzer içeriğe sahip’ filmler verisi kullanılabilmektedir. Böylece sistem çok sayıda aynı filmi izlemiş kullanıcıları aramak yerine benzer içerikli filmleri izlemiş kullanıcıları aramaktadır ve bu verinin erişimi daha kolay olmaktadır. Aynı zamanda popüler olan bir film ile aynı konuda olan popüler olmayan bir filmi de sistem önerilebilmektedir. Fakat içerik bazlı yöntemlerin faydalı olabilmesi için içeriğin öğrenilebilir bir fonksiyon ile ifade edilebiliyor olmasını gerektirir. Bu amaçla makine öğrenme yöntemleri kullanılarak içerik, öznitelikleri ile ifade edilir.

Proje kapsamında yapılan ilk çalışmalarımızdaki amaç kullanıcı ve filmlerin aynı anlamsal uzaya taşınması ve bu uzayın tüm elemanları arasındaki uzaklığı ifade edecek metrikleri öğrenecek algoritmaların uygulanmasıydı. Bu sayede kullanıcıların birbirleri arasındaki uzaklık, filmlerin birbirleri arasındaki uzaklık ve kullanıcı film arasındaki uzaklık hakkında nicelik bilgisi edinilmektedir. Uzaklık bilgisinin önemi, sistemin verdiği önerinin hangi kaynaklar tarafından ne oranda etkilendiğini bildirebilmesinden dolayıdır. Böylece verilen öneri, sistem tarafından açıklanabilir hale gelir. Hangi etkenlerin önerinin verilmesinde etkili olduğunun kullanıcıya bildirilebilmesi sayesinde kullanıcıdan önerinin verilmesinde etkili olan etken veya etkenlerin doğru/yanlış olduğu ile ilgili geri bildirim alınabilmesini sağlar. Bu bilgi sayesinde öneri sisteminin öğrendiği kullanıcı-kullanıcı/film-film/kullanıcı-film ilişkilerindeki hatalar daha isabetli bir şekilde giderilir.

Kişiselleştirme konusundaki en büyük problemin verinin özetlenmesi (modellenmesi) olduğu görülmüştür. Bu problemle ilgili farklı yaklaşımlar incelenmiştir. Projede bu amaçla bir istatistiksel öneri modeli geliştirilmiştir. Bu öneri modeli bir dizi log-benzerlik formülünü kapsar. Bu formüller bir istatistiksel dağılımı temsil eder. Kullanıcı ve filmler farklı parametrelerle ifade edilen istatistiksel dağılımlar olarak temsil edilir. Film ve kullanıcılar hakkında bilgi edinilmesi de bu parametrelerin güncellenmesini içerir. Sistem daha sonra bu istatistiksel dağılım fonksiyonları üzerinde çalıştırılacak sorgular ile kullanıcı-kullanıcı/film-film/kullanıcı-film benzerliği hakkında olasılıksal çıktı üretir.

Bu çalışmalar sonucunda ortaya çıkan yüksek lisans tezinde (ERYOL, 2010), melez sosyal öneri problemi için çatı olarak, istatistiksel temel sağlayan olasılıksal saklı anlam analizi yöntemini (Probabilistic Latent Semantic Analysis) önerilmiştir. Farklı veri melezleştirme yaklaşımları üzerinden deneyler hazırlanmıştır. Bu esnek olasılıksal model üzerinde ağ düzenleme ve model harmanlama yaklaşımları sosyal güven ağının kolektif filtreleme sürecinde kullanımı için önerilmiştir. Deneylerde, önerilen yöntemler başarılı bir şekilde temel seviye yöntemlerden daha yüksek başarı göstermiştir. Araştırma sonucunda, önerilen yöntemlerin oy ve sosyal güven ağı verilerini teoriye uygun olarak bir arada modellediği gösterilmiştir.

5.2. İçerik destekli işbirlikçi filtreleme yaklaşımı

Bu çalışmada, ağ tabanlı bir film öneri sisteminin tasarımı, gerçekleştirimi ve değerlendirilmesi yapılmıştır. Bu sistem, kullanıcılarına IMDb'deki bütün film bilgilerine ulaşmanın yanında, bu filmler hakkında yorumda bulunup ilgilerini çekebilecek film önerileri alabilme imkanı tanımaktadır.

Sistemde kullanılan film verileri, 'Bilgi Çıkarımı' teknikleri ile internetteki en geniş kapsamlı film veritabanı olan IMDb'den elde edilen tür, konu, kast, yönetmen, yıl, oy, isim, slogan, dil, süre, şirket ve anahtar kelime bilgilerini içermektedir. Bu verilerin çıkarımı için IMDbPy adlı bir Python paketinden yararlanılarak IMDb'nin güncel bir yerel kopyası oluşturulmuştur.

Geleneksel önerme sistemlerinin, kullanılacak veri seyrekken, yani komşu nesne ya da kullanıcı sayısı az olduğunda başarısız olduğu görülmüştür. Bu başarısızlığın önüne geçebilmek için, bu çalışma kapsamında geliştirilen sistem, eksik veri tahmini ve lokal/global benzerlik kavramlarına dayalı ve de hem kullanıcı, hem de nesne tabanlı bir işbirlikçi (kolaborative) filtreleme tekniğinden yararlanmaktadır. Kullanıcı tabanlı tahminler, benzer kullanıcıların oyları işlenerek, nesne tabanlı tahminler ise benzer nesnelerin oyları ile bu nesnelerin içerik olarak benzerliğinin harmanlanması sonucunda oluşturulmaktadır. Bu işlem sonucunda elde edilen tahminler, ancak belirli bir eşik değerinin üzerine ulaştığında yapılmaktadır. Bu sırada kullanılan bütün benzerlik hesaplamalarında 'Pearson Correlation Coefficient' tan faydalanılmıştır.

İki filmin içerik olarak benzerliğinin hesaplanması için, Bilgi Çıkarımı modülü sayesinde elde edilen bütün verilerden yararlanılmıştır. Bunlardan metin tabanlı olanları, çeşitli NLP teknikleri ile işlenmiştir. Uygulanan işlemler sonucunda her film, bir vektör ile ifade edilir hale getirilmiştir. Böylelikle iki film arasındaki benzerlik, Cosine Benzerliği metodu ile bulunabilmektedir.

Kullanıcı tabanlı tahminlerin yapılmasında kullanılan 'global benzerlik' kavramı sayesinde, aktif herhangi bir kullanıcının lokal komşu sayısının yetersiz olması durumunda, daha fazla komşunun ortaya çıkması sağlanmaktadır. Bunu başarmak için, düğümleri kullanıcılar, kenarları ise kullanıcılar arasındaki lokal benzerlik miktarı olarak belirlenen bir çizge oluşturulmuş ve bu çizgedeki iki kullanıcı arasındaki global benzerlik, ait oldukları iki düğüm arasındaki maksimum uzaklık olarak belirlenmiştir. Bu hesaplamalar için Floyd-Warshall algoritmasından faydalanılmaktadır.

Oluşturulan sistemin başarısını var olan sistemlerle karşılaştırabilmek için MoviLens veri kümesinden yararlanılmıştır. Bunun için tahmin edilen oyların gerçekte verilmiş oylarla karşılaştırılması yöntemi kullanılmıştır. Bu yüksek lisans çalışmasının (ÖZBAL, 2009) detayları hem bir konferans hem de bir dergi makalesinde yayınlanmıştır (ÖZBAL, 2010; ÖZBAL, 2011).

5.3. İşbirlikçi yöntemlerle desteklenmiş içerik tabanlı öneri sistemi

Bu çalışmada, işbirlikçi ve içerik tabanlı filtreleme yöntemlerinin her ikisine de dayanan bir film öneri sistemi olan *ReMovender*(KARAMAN, 2010)geliştirilmiştir. Bu çalışmanın ayırt edici noktaları, kullanıcılar arasında ilişki kurma metodolojisi ve filmlerin içerik bilgilerinin kullanılma şeklidir. *ReMovender*, kullanıcılara filmleri bir ve beş aralığındaki oylarla oylama fırsatı vermektedir. Eksik verileri tamamlamak amacıyla, bu oy bilgilerini işbirlikçi yöntemle kullanıcılar arasında benzerlik bulmada kullanmaktadır. İçerik tabanlı kısımda ise, filmlerin içerik bilgileri kullanılarak bu filmler ilişkilendirilmekte ve kullanıcılara öneriler sunulmaktadır.

Film öneri sistemlerinin karşılaştığı en büyük problemlerden biri olan veri seyrekliği, sistemin ileri sürdüğü yöntemle kısmen çözülmüştür. Verilen oylar kısıtlı olduğunda, örneğin bir kullanıcı az sayıda filme oy verdiğinde, bu kullanıcının zevkini tahmin edebilmek oldukça zordur. Bu noktada içeriği bir kenara bırakıp, işbirlikçiişbirlikçi bir şekilde bu kullanıcının oy vermediği bazı filmlere verebileceği muhtemel oylar tahmin edilerek veri seyrekliği biraz da olsa yenilmiş olmaktadır. Yeterli miktarda veriyi elde ettikten sonra, filmlerin içerikleri (yönetmenleri, türleri, anahtar kelimeleri, özetleri, vs.) kullanılarak, bu filmler arasındaki benzerlikler bulunmakta ve bu benzerlik değerleri kullanılarak, henüz oylanmamış filmler [0,5] aralığında sınıflandırılmaktadırlar.

Filmler sınıflandırılırken, bir filmin içeriğinde bulunan alanların kullanıcı açısından farklı ölçülerde önem taşıdığı göz önünde bulundurularak, filmler arasındaki benzerlikler hesaplanırken her alana farklı ağırlıklar verilmektedir.

Alabileceği değerler sınırlı olan tür, yönetmen, dil gibi özellikler sınıflandırılırken, öneri sistemlerinde kullanılmaya oldukça uygun olan “Naive Bayes Classifier” yöntemi kullanılmaktadır. Anahtar kelime, özet gibi metin özelliği taşıyan özellikler arasındaki benzerlikler ise metinler arasında benzerlik bulmaya yarayan araçlar kullanılmaktadır.

İçerik üzerine kurulmuş olmasına rağmen veri seyrekliğini işbirlikçi yollarla engelleyen bu sistem, şu ana kadar yapılmış diğer film öneri sistemlerine oranla daha başarılı olmuştur. Bu çalışmanın detayları bir yüksek lisans tezinde ve bir dergi makalesinde anlatılmaktadır (ÖZBAL, 2010; ÖZBAL, 2011).

5.4. Bir çizge algoritmasına dayalı melez video öneri sistemi

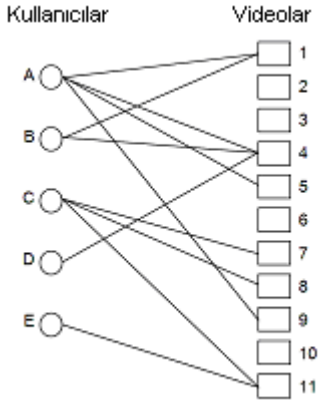
Bu çalışmada da yeni içerik bulma ve öneri yapma konusunda içerik bazlı filtreleme ve işbirlikçi filtreleme teknikleri birleştirilerek kullanılmıştır. Bu çalışmada, günümüzün popüler video paylaşım sitesi olan YouTube ortam olarak kullanılmıştır. YouTube, üzerinde geliştirme yapacak kişiler için Uygulama Programı Arabirimi (UPA) sağlamaktadır[3]. UPA, YouTube üzerindeki içeriğin belirli bir bölümüne ulaşıma olanak sağlamaktadır. Ancak, bu çalışma için gerekli olan veri kümesini desteklememektedir. Bu nedenle kullanılacak veri kümesinin bir modül yardımıyla toplanması gerekmiştir. Bu modülün geliştirilmesi için Java Platformu ve YouTube tarafından sağlanan UPA kullanılarak, YouTube kullanıcıları bilgilerini, izledikleri ve oy verdikleri videoları ve bu oy değerlerini kapsayan bir veritabanı oluşturulmuştur ve bu veritabanının sürekli güncel tutulması sağlanmıştır.

Literatürü taradığımızda YouTube üzerinde en ümit verici çalışmanın sadece kolaboratif yaklaşım kullanarak geliştirilen Adsorption algoritması(BALUJA, 2008)olduğuna karar verdik.

[3]YouTube Developer's Guide: Data API Protocol. http://code.google.com/apis/youtube/2.0/developers_guide_protocol.html

Adsorption algoritması bir çizge (graph) algoritmasıdır. Geliştirdiğimiz melezsistem,çizge tabanlı bu algoritma esas alınarak yapıldı.

Çalışmada,öncelikle Adsorption algoritması aynen gerçekleştirilmiştir. Daha sonra algoritmaya, YouTube'da her videonun etiketlerinden yararlanılıp, içerik bazlı filtreleme eklenmiştir. Böylece, elde edilen melez (hybrid) sistemle daha doğru öneriler elde edilmiştir. Adsorption algoritmasına göre kullanıcılar ve videolar bir grafiğin düğümlerinden her birini oluşturmaktadır. Örnek bir çizge aşağıdaki şekilde görülebilir:



Orijinal Adsorption algoritmasınıkullanılan öneri sistemi üç fikri içermektedir. Buna göre bir videonun bir kullanıcı için uygunluğu aşağıdaki koşullarla belirlenmektedir:

1. Kullanıcı ve video arasında kısa bir yol varsa
2. Kullanıcı ve video arasında birkaç yol varsa
3. Kullanıcı ve videonun arasında yüksek dereceli olmayan yollar varsa

Kullanılan ana mantıkta bazı düğümlerin etiketleri mevcuttur. Algoritma, çizge üzerinde gezinerek bilinen etiketlerin bilinmeyen düğümlere aktarılması esasına dayanır. Bunun sonucunda da her düğüm için bir olasılık dağılımı elde edilmektedir. Video öneri sistemlerinde de çıktı olarak elde edilen bu olasılık dağılımından yararlanılmaktadır.

Gelişigüzel yürüme sistemiyle Adsorption'da etiketleri bilinen her düğümün gölge düğümü oluşturulur. (BALUJA, 2008)'de gölge düğüm ve asıl düğüm arasındaki ağırlığın çeşitli parametrelerle değişebileceği belirtilmiştir. Ancak gerçekleştirimi yapılmamıştır. Bu projede, var olan algoritma gerçekleştirilmiş, ek olarak gölge düğümler ve asıl düğümler arasındaki yakınlığa karar verirken değişik elementlerden yararlanılmıştır. Ayrıca bir katkı da elde edilen olasılık dağılımı üzerinde yapılmaktadır. Öncelikle her kullanıcının daha önce izlemiş olduğu videolardan ve bu videolara verdiği oylardan yararlanılarak bir kullanıcı profili oluşturulmuştur. Bu profil, YouTube'da her video için var olan video etiketleriyle karşılaştırılıp profile benzer videoların yararına önerme oranı arttırılmıştır.

Son katkı da bir eşik değerinin belirlenip uygulanmasıdır. Buna göre, bu değerin üzerinde kalan videolar öneri sisteminde kullanılmaktadır. Değerlerde eşitlik söz konusu olduğunda ise videonun ortalama beğeni değeri ve izlenme sayısı gibi nicelikler gözönünde bulundurularak öneri gerçekleştirilmektedir.

Adsorption algoritması, bu eklemelerle YouTube veri tabanı üzerinde test edilmiştir. Elde edilen sonuçlar beklendiği gibi makuldur, ancak veri kümesinin dağınıklığı sebebiyle düşük seviyede çıkmıştır. Bu nedenle algoritmanın başarısını teyit etmek için verileri daha düzenli olan bir veri tabanının kullanılması gündeme gelmiştir. Yapılan araştırmalarla MovieLens veri kümelerinden birinin bu amaçla kullanılabileceğine karar verilmiştir. MovieLens, GroupLens tarafından geliştirilen, web-tabanlı ve şu anda aktif olarak çalışan bir film öneri

sistemidir[4]. Kullanıcıların çeşitli filmlere oy (rating) vermesiyle çalışır. Bu sebepten zaman içinde gerçek kullanıcılardan ve filmlerden oluşan güçlü bir veri tabanı oluşmuştur. MovieLens geliştiricilere destek olmak amacıyla, değişik ihtiyaçlara karşılık verebilecek çeşitli veri tabanlarını paylaşmıştır.

Projede, YouTube'den elde edilen veri tabanının yanı sıra MovieLens'in hazır olarak sunduğu veri tabanlarından biri kullanılmıştır. Bunun sebebi, daha önce belirtildiği gibi, Youtube veri tabanının dağınık olması ve çalışmanın ikinci bir platform kullanılarak daha değerlendirilme isteğidir. MovieLens veri kümesi kullanılarak yapılan değerlendirmeler sonucunda, beklendiği gibi daha başarılı sonuçlar elde edilmiştir.

Bu aşamaya kadar önce sadece Adsorption algoritmasının değerlendirilmesi yapıldı. Bu noktadan sonra içerik bazlı tekniklerin özelliklerinden de yararlanma çalışmaları yapıldı. Adsorption algoritması kullanıcılar için dağılım listesi dönmektedir. Adsorption'dan elde edilen dağılımlar daha önce kullanıcıların geçmişi kullanılarak oluşturulan profillerle birleştirme fikri denendi. Değerlendirme öncellikle YouTube veri kümesi üzerinde yapıldı. Elde edilen sonuçlar yine yüksek değildi ancak sadece Adsoption kullanılarak oluşturulan sonuçlardan daha başarılı oldu. Melez sistem, MovieLens veri kümesi üzerinde de değerlendirildi. Elde edilen sonuçlara göre Adsorption algoritması tek başına önerilerde %10 gibi bir başarı gösterirken göre melez sistem %15-%20 arasında başarı gösterdi.

Bu çalışma bir yüksek lisans tezinin konusu olmuştur (ÖZTÜRK, 2010). Çalışma ayrıca bir konferans makalesi olarak kabul olmuştur (ÖZTÜRK, 2011).

5.5. Ontoloji-tabanlı kullanıcı tercihlerine göre video öneri sistemi

Bu çalışmada kullanıcı tercihlerini kullanan bir içerik öneri sistemi geliştirilmiştir. En verimli sonucun alınabilmesi için içerik tabanlı ve işbirlikçi öneri mekanizmalarını birlikte kullanan melez bir strateji uygulanmıştır. Sistem, kullanıcı profillerinin oluşturulması, kullanıcıların gruplanması ve içerik önerme olmak üzere üç temel fonksiyonu yerine getirir. Kullanıcı profillerinin hazırlanması ve aday içeriğin değerlendirilip kullanıcıya önerilmesi aşamalarında içerik tabanlı; aday içeriklerin bulunması aşamasında ise işbirlikçi yaklaşım ağırlıklı olarak kullanılır. Kullanıcıların gruplanması ise kullanıcı geçmişi veya içerik üzerinde kullanıcı tercihlerine göre yapılabilir.

Sistem, kullanıcıların daha önceki davranışlarından yola çıkarak önerilecek içerikle ilgili tercihlerini belirler ve bu tercihlere göre kullanıcının profilini oluşturur. Kullanıcı profillerinde, kullanıcının daha önce değerlendirip oy verdiği içeriklerden oluşan kullanıcı geçmişi ve kullanıcı tercihleri yer alır. Kullanıcı tercihi içeriklere ait ayırt edici özellikler, bu özelliklerin kullanıcının seçimindeki etkisi ve her özelliğin alabileceği değerlerin kullanıcı tarafından beğeni derecesi yer alır. Kullanıcının her özellik ve değeri için beğenisi, o kullanıcının, özelliklerin ait olduğu içeriği beğeni düzeyi yani verdiği oy ile belirlenir. Kullanıcının yüksek oy verdiği içeriklerde ortak değerlere sahip olan fakat her kullanıcı için farklılık gösteren özellikler, kullanıcının seçimlerinde önemli etkiye sahip demektir. Özelliklerin değerlerinin beğeni düzeyi ise, içinde yer aldıkları tüm içeriklerin beğeni düzeyi ile orantılıdır.

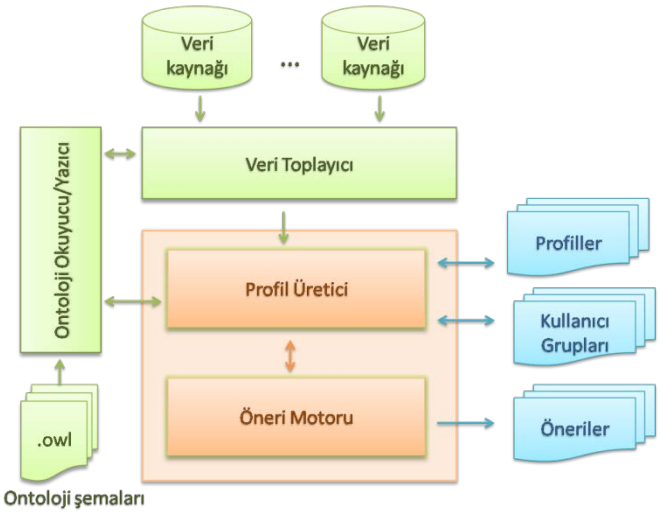
İşbirlikçi öneri mekanizmasının kullanılabilmesi için kullanıcı profillerinden yola çıkılarak, tercihleri benzer olan kullanıcılar gruplar altında toplanır. Gruplama doğrudan içeriklerin aldıkları oyların ya da kullanıcıların profillerinin benzerliğine göre yapılabilir. Kullanıcı profillerinin benzerliği temel alındığında, kullanıcıların tercihlerinde öne çıkan elemanlar belirlenir. Gruplar en çok ortak özelliği paylaşan profiller bir araya getirilerek oluşturulur.

[4]MovieLens veri kümesi, http://www.grouplens.org/node/73, son erişim 13.07.2010.

Kullanıcıya önerilecek aday içerikler, kullanıcının ait olduğu grup içerisindeki diğer kullanıcıların beğendikleri içerikler arasından seçilir. Aday içerikler, öneri yapılacak kullanıcının profili ile karşılaştırılır. Kullanıcı profilinde yer alan tercihlere en benzer özellikler içeren adaylar kullanıcıya önerilir. Hem işbirlikçi önerinin uygulanacağı grupların içerik tabanlı olarak oluşturulması hem de işbirlikçi öneri mekanizması ile belirlenen adayların içerik tabanlı olarak filtrelenmesi ile bu iki mekanizma birlikte kullanılmış olur.

Kullanıcı profilleri, içerik tabanlı tercihleri daha iyi temsil edebilmek üzere, bir ontoloji ile temsil edilir. İçerikler de iyi tanımlanmış bir alan ontolojisinden seçilir. Bu, içerikler ve özellikleri arasında anlamsal bağların kurulmasını mümkün kılar. Bu çalışmada iyi tanımlanmış olması ve yeterli veriye sahip olması nedeniyle film alanı seçilmiştir. Film alan ontolojisi olarak Freebase ontolojisinin[5]film alanı kullanılmıştır. Kullanıcı oyları için ise MovieLens veri setleri kullanılmıştır. MovieLens kullanıcı profilleri ile Freebase'den elde edilen içerikler eşleştirilerek kullanıcıya ait gerekli veriler elde edilir. Bu verilerin elde edilmesinden sonra kullanıcı profillerinin oluşturulması, kullanıcıların gruplanması ve öneri işlemleri girdi olarak kullanılan ontolojiler ve verilerin yapısından bağımsızdır. Dolayısıyla sistem, farklı ontolojiler ile farklı içeriklerin önerilmesi için kullanılabilir.

Yukarıda belirtilen işlevselliği gerçekleştirecek sistemin çatı mimarisi Şekil 11'de gösterilmiştir. Profil Üretici ve Öneri Motoru sistemin ana fonksiyonları gerçekleştiren temel elemanlardır. Veri toplayıcı ve Ontoloji Okuyucu/Yazıcı farklı ontoloji ve içeriklere göre verilerin toplanmasını ve temel elemanlara gereken girdinin sağlanmasından sorumludur.



Şekil 11 – Öneri sistemi çatı mimarisi

Profil Üretici, kullanıcı profillerinin yaratılmasından sorumlu *Profil Oluşturucu* ve profillerin gruplanmasından sorumlu *Profil Gruplayıcı*'dan oluşur.

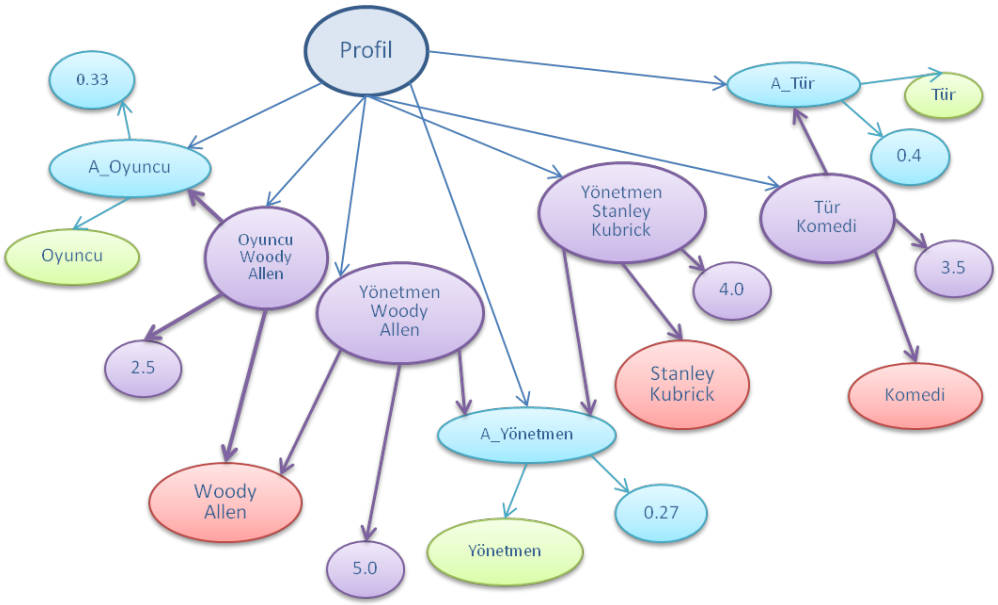
Profil Oluşturucu, girdi olarak veri toplayıcıdan kullanıcının içeriklere verdiği oyları, içeriğe ait ayırt edici özellikler ve oylanan içeriklerin bu özelliklerinin değerlerini alır. Film öneri sisteminde kullanıcının izlediği filmler ve filmlere verdiği oylar; filmlerin "oyuncu", "tür", "yönetmen", vb. özellikleri ve bu özelliklerin örnekleri girdiyi oluşturur. Her özellik için en çok tekrarlanan örneklerin tekrarlanma sayıları belirlenir (i_n). Her bir özellik için i_n değerinin, tüm i_n değerlerinin toplamına oranı o özelliğin ağırlığını gösterir. Tüm özelliklerin ağırlıkları toplamı 1'e eşittir. Örneklerin aldığı oylar filmin hangi özelliği olarak yer aldıklarına göre değişir. Örneğin, A kişisinin farklı filmler veya aynı filmde birden fazla rolü olması durumunda,

[5] http://www.freebase.com

"oyuncu" olarak oyu ile "yönetmen" olarak oyu farklı olacaktır. Şekil **12** 12'de bir kullanıcı profilinin bir bölümü gösterilmiştir.

Profil Gruplayıcı, kullanıcıları profillerinin benzerliklerine göre istenen sayıda gruplara ayırır. Kullanıcıların gruplanmasında *k-means* algoritmasının bir uyarlaması kullanılır. Her kullanıcının en öne çıkan (oyu en yüksek olan) tercihleri seçilerek her kullanıcıya ait birer profil vektörü oluşturulur. Beğeni sınırı 3.5 olarak belirlendiğinde, Şekil 12**Şekil 12**'deki gibi bir kullanıcı profilinden oluşacak profil vektörü basitçe Şekil 13'teki gibidir. Kullanıcılar arasından grupların merkezlerini oluşturacak rastgele profiller seçilip başlangıç grup profilleri oluşturulur. Başlangıçta grup profilleri merkez olarak kullanıcının profil vektörü ile belirlenir. Gruplama algoritmasının her yinelemesinde kullanıcılar kendi profil vektörleriyle en çok ortak tercihlere sahip gruba eklenir; tüm kullanıcılar eklendikten sonra, gruptaki kullanıcıların profil vektörlerinde ortak olarak en çok yer alan tercihlerle grup profili güncellenir. Bir grup profili, gruptaki kullanıcıların ortak tercihlerini belirten bir profil vektörü, gruba ait kullanıcılar ve bu kullanıcıların en çok oy verdikleri içeriklerden meydana gelir.

Profil Üretici, son ürün olarak *kullanıcı profilleri* ve *kullanıcı grupları*nı oluşturur. Bu ürünler *Öneri Motoru* tarafından kullanılır.

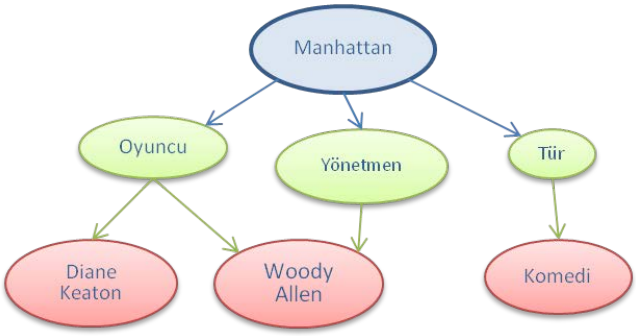


Şekil 12 – Kullanıcı profili

Tür Komedi	Yönetmen Woody Allen	Yönetmen Stanley Kubrick

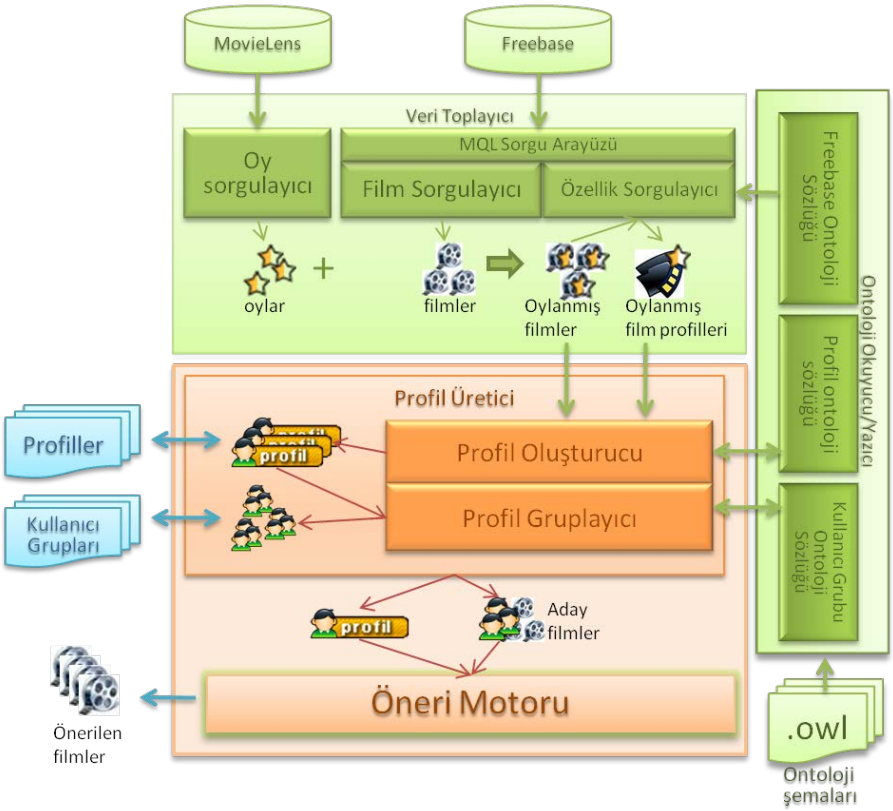
Şekil 13 – Profil vektörü

Öneri Motoru, kullanıcıya ait profili ve kullanıcının içinde bulunduğu grubun profilini kullanarak kişiye içerik önerisinde bulunur. Kullanıcının ait olduğu grubu bularak, o gruptaki kullanıcılar tarafından yüksek oy alan yani tercih edilen içerikleri önermek üzere aday olarak belirler. Aday bir içeriğin her özelliğinin(f) tüm değerlerinin(i) kullanıcının profilindeki oyunu(r_{fi}) ve özelliğin ağırlığını(w_f) kullanarak, $\sum (\mu r_{fi} \times w_f)$ formülü ile kullanıcının aday içeriğe verebileceği oyu tahmin eder. En çok oyu alan adayları sırayla kullanıcıya önerir. Şekil 14'teki gibi bir film önerilmek üzere aday olarak belirlendiğinde, bu filmin Şekil 12'deki kullanıcı profiline göre alacağı oy $0{,}33 \times (0 + 2{,}5) + 0{,}27 \times 5 + 0{,}4 \times 3{,}5 = 3{,}575$'tir.



Şekil 14 - Film örnek ontolojisi

Veri Toplayıcı, sistemin, kullanılan farklı veri türleri ve kaynaklarının farkında olan kısmıdır. Profili oluşturmak için gerekli verileri hazırlar. Film öneri sisteminde bu modül, Profil Oluşturucu'nun kullanılan veri setlerine bağımlı işlemler eklenek *MovieLens/Freebase Profil Oluşturucu olarak* genişletilmesi ile gerçekleştirilir. Modül, buna ek olarak verileri çekmek için FilmSorgulayıcı, Özellik Sorgulayıcı ve Oy Sorgulayıcı alt modüllerini içerir. Tasarlanan film öneri sistemine özel olarak gerçekleştirilmiş sistem mimarisi Şekil 15'te gösterilmiştir.



Şekil 15 - Film öneri sistemi mimarisi

Oy Sorgulayıcı, MovieLens veri setinden kullanıcıların izledikleri filmleri ve verdikleri oyları alır. MoviLens verileri kullanıcı oyları ve filmler için olmak üzere iki ayrı sıradan erişimli dosya halinde bulunurlar. Buradan elde edilen filmler sadece adları ile tanımlandıklarından sistem için gerekli anlamsal bilgiyi sunamazlar. *Film Sorgulayıcı*, kullanıcıların izledikleri filmleri Freebase ontolojisinde bularak, ontolojik tanımlarla oylamaları eşleştirir. Bu da *Özellik Sorgulayıcı*'nın oylanmış olan filmlerin istenen özelliklerinin değerlerini yine Freebase ontolojisinden bulmasına olanak sağlar.

Ontoloji Okuyucu/Yazıcı, kullanılan ontolojilerdeki (sistem dışında veya içinde tanımlı) gerekli kaynakların URI'lerini ve gerekirse ontoloji şemalarını içeren sözlüklerin tümünü belirtmek için kullanılmıştır. Ontolojileri okuyup yazma, ontolojilerde yer alacak kaynakları oluşturma işlerinde görev alır.

6. Sistem Entegrasyonu

Projenin son döneminde, proje süresince paralel yürümüş olan çalışmaların sonuçlarını, tek bir sistem altında, beraber işleyecek şekilde bir araya getirmek için entegrasyon çalışmaları yapılmıştır. İlk etapta (DEMİRDİZEN 2010)'deki yüksek lisans tezi kapsamında geliştirilen MPEG-7 ontolojisine dayalı, alan ontolojilerinin entegrasyonuna izin veren, ontolojik kavramlar kullanılarak video etiketleme ve sorgulama yapılabilen video yönetim sistemine, (GÖKTÜRK, 2009)'deki yüksek lisans tezi kapsamında geliştirilen futbol videolarından MPEG-7 standardına uygun üst veri çıkaran ve XML formatında çıktı üreten programı ve (ALACA-AYGÜL, 2010)'deki yüksek lisans tezinde önerilen MPEG-7 tabanlı alan ontolojilerini anlamsal ve uzay-zamansal sorgulamak için geliştirilen doğal dil sorgu arayüzü entegre edilmiştir. İkinci etapta da (YAPRAKKAYA, 2010)'deki yüksek lisans tezinde geliştirilen yüz tanıma aracı sisteme dahil edilmiştir. Şekil 16'de, entegrasyon planı özetlenmiştir.

Şekil 16- *Entegrasyon Planı*

Video yönetim sistemi MPEG-7 ontolojisine dayalı bir yapıda olduğu için, benzer şekilde MPEG-7 ontoloji tabanlı sistemlerle uyumlu bir şekilde çalışabilir olması ve ortak bir dil üzerinden haberleşebilmesine imkan sağlamaktadır. Ancak, sistemin MPEG-7 *standardında* etiketlenmiş videoları da arşivleyip sorgulamaya izin vermesi sistemin daha geniş video kaynaklarınca kullanabileceğini sağlayacağı için tercih sebebidir. Örneğin (GÖKTÜRK, 2008)'de geliştirilen futbol videolarından yardımcı veri çıkarma programının çıktısının MPEG-7 standardı ile uyumlu XML dosyalarıdır.Bu tip videoları da sisteme ekleyebilmek için video yönetim sisteminin MPEG-7 standardına uygun XML dosyalarını da kabul edebilmesi gereksinimi doğmuştur.

MPEG-7 standardı ile uyumlu XML dosyalarının, MPEG-7 ontolojisine içeri aktarılmaları (import) işlemi, teknik farklılıklardan dolayı bir yordamın tanımlanmasını gerektirmiştir. Bu yordamın ilk adımı için, öncelikle XML dosyaları RDF formatında elde edilmelidir. Bu işlem ReDeFer projesi (http://code.google.com/p/redefer/) kapsamında düzgün bir biçimde gerçekleştirilmektedir. Bu projenin ekibinden kaynak kodu istenerek, işlemi gerçekleştirmek için gerekli olan kod parçası projede kullanılmıştır.

MPEG-7 ontolojisindeki ve üretilen RDF dosyasındaki isim uzayları (namespace) uyuşmamaktadır. İsim uzaylarındaki bu farklılık, içeri aktarım sırasında aynı sınıf (class), veri özelliği (data property) veya nesne özelliği (object property) olan unsurların, farklı isim uzaylarında olmaları nedeniyle farklı unsurlar gibi davranmalarına yol açmaktadır. Bu durum da, çıkarsama (inference) işlemi sırasında ontolojinin yararlarından faydalanamamaya sebep olmaktadır. Bu sorunu çözmek için, video yönetim sistemine içeri aktarılacak RDF dosyalarına, sistemin kullandığı MPEG-7 ontolojisinin isim uzayını enjekte etmek gerekmektedir. Bunun için bir isim uzayı değiştiricisi ile önce MPEG-7 uyumlu XML dosyası değiştirilmekte ve ReDeFer aracı ile RDF dosyasına çevirme işlemi değiştirilmiş XML dosyasına uygulanmaktadır. Bu sayede yapılan içeri aktarma işleminde aynı sınıf, veri özellikleri ve nesne özellikleri aynı unsur olarak ele alınabilmekte ve çıkarsama işlemi sırasında doğru sonuç alınmaktadır.

MPEG-7 standardı ile uyumlu XML dosyalarının içeri aktarılması işlemini kısaca özetlemek gerekirse, öncelikle sistemin MPEG-7 ontolojisinin isim uzayını enjekte ederek XML dosyasını yeniden yazmak, değiştirilmiş XML dosyasını ReDeRer aracı ile RDF formatına çevirmek ve ana sistemde MPEG7 modeline RDF dosyasının içeri alınması işlemini uygulamak olarak adımlayabiliriz. Bu işlem her ne kadar futbol videolarından yardımcı veri üretme aracının entegrasyonu için tanımlanmış olsa da, aslında genel tanımlı olması nedeniyle herhangi bir MPEG-7 uyumlu XML dosyasını sisteme dahil edebilir. Bu durumda, MPEG-7 XML'i ile çalışan herhangi bir sistemle kolayca entegre olunabilir, veri alış verişi sağlanabilir.

(ALACA-AYGÜL, 2010)'deki yüksek lisans tezi kapsamında sunulan doğal dil sorgu arayüzünün entegrasyon çalışmaları da tamamlanmıştır. Bu tezin geliştirilmesinde .NET ortamı kullanılması, video yönetim aracının ise Java teknolojileri ile geliştirilmesi nedeniyle, sistemlerin entegrasyonu için en kolay yol web servisleri olarak ortaya çıkmıştır. Bu nedenle, video yönetim sisteminin temel fonksiyonları web servisleri olarak yeniden yazılmıştır. Doğal dil sorgu arayüzü için de web servisleri hazırlanmış, bu web servislerini kullanarak, entegre edilen sistemin doldurulduğu (populate) veritabanında doğal dil ile sorgulama yapmaya imkan veren bir web arayüzü hazırlanmıştır.

Ayrıca (YAPRAKKAYA, 2010)'deki yüksek lisans tezi kapsamında hazırlanan yüz tanıma ve etiketleme aracının entegrasyonu da, video yönetim sisteminden açılan web servisler aracılığıyla gerçekleştirilmiştir. Bu entegrasyon işinin detayları aşağıda açıklanmıştır.

(DEMİRDİZEN, 2010)'deki yüksek lisans tezi kapsamında Java platformu üzerinde geliştirilen web tabanlı video yönetim sistemi videolardaki kişilerin ve olayların el ile anlambilimsel olarak etiketlenmesi esasına dayanmaktadır. (YAPRAKKAYA, 2010)'deki yüksek lisans tezi kapsamında *C++ ile Windows işletim sistemine özel olarak geliştirilen yazılım da* videolarda görünen yüzleri tespit edip öğrenen, öğrendiği yüzleri aynı videonun devamında ya da başka videolarda bulan bir yazılımdır. El ile yapılan işlemlerin en azından bir kısmını otomatik yapabilmek amacıyla insan yüzü etiketleme işleminin otomasyonunun bu iki tez çıktısının entegrasyonu ile yapılabileceği düşünüldü.

Yazılımların birbirlerinden bağımsız olarak farklı platformlarda farklı yazılım dilleri ile geliştirilmiş olmalarından dolayı entegrasyon için faklı çözümler üretme gereksinimi doğdu. Öncelikle entegrasyon için iki alternatif belirlendi: yazılımların tamamen web tabanlı çalışması ya da yarı kullanıcı tarafında çalışan yarı web tabanlı melez bir sistem geliştirilmesi. Öncelikle tamamen web tarafında çalışabilecek bir yazılım için ön araştırmalar yapıldı. CGI (Common Gate Interface), Fast CGI ve server pushing teknolojileri kullanılarak örnek yazılımlar geliştirildi. Başka bir alternatif olarak da yüz tanıma yazılımının JAVA ile tekrar geliştirilmesi planlandı. Ancak etiketlenecek videoların upload gereksinimi, fazla bant genişliğine ihtiyaç duyması ve görüntü işleme işleminin yüksek performansa ihtiyaç duyması nedenleri ile bu çözüm yolundan vazgeçildi. İkinci alternatif olan melez bir sistem geliştirme

üzerinde karar kılındı. Ancak burada da JAVA tabanlı bir sistem ile C++ tabanlı diğer sistemin haberleşmesi sorunu ortaya çıktı ve bu sorunun web servis teknolojisi kullanılarak çözüme ulaştırılmasına karar verildi.

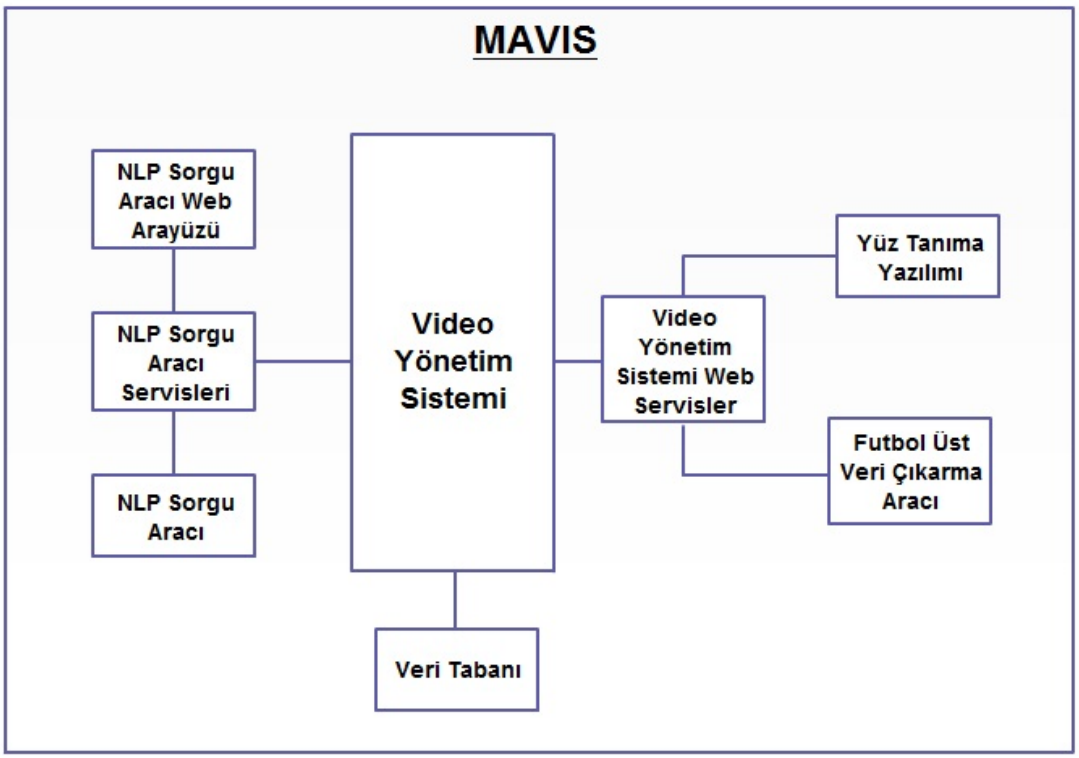
Sistemlerin otomatik hale getirilmesindeki amaç daha kısa sürede daha hızlı çözümler üretmektir. Bu bağlamda yüz tanıma sistemindeki performans sorunları giderildi, yazlımın bir video karesini işleme süresi 0,5 – 0,8 sn aralığından 0,03 – 0,05 sn gibi bir aralığa düşürülerek yazılımın çalışma hızı arttırıldı, böylece gerçek zamanlı video işleme olanağı sağlandı.

Java tabanlı video yönetim sistemi ile C++ tabanlı yüz tanıma yazılımının haberleşmesi için ortak web servis oluşturulmasına karar verildi. Bu nedenle video yönetim sisteminin bazı fonksiyonlarının kullanılmasına izin veren bir web servis geliştirildi.Web Servis sunucusunun sunduğu fonksiyonların C++ tabanlı yüz tanıma yazılımında kullanılabilmesi için GSoap2 yazılım kütüphanesi kullanılarak istemci (client) tarafı web servis yazılımı geliştirildi.İstemci tarafı web servis yazılımı yüz tanıma yazılımına entegre edilerek, yüz tanıma yazılımına otomatik etiketleme yapabilmesi için yeni fonksiyonlar eklendi ve yazılım bu çerçevede yeniden düzenlendi.Yüz tanıma yazılımına yeni fonksiyonların eklenmesi ve etiketleme sisteminin gerektirdiği bir takım arayüz zorunlulukları nedeni ile yüz tanıma yazılımına yeni bir arayüz geliştirildi ve entegrasyon tamamlandı.

Bu entegrasyon işlemi sonucunda kişilerin otomatik olarak etiketlenmesini sağlayacak melez bir sistem geliştirilmiş oldu. Geliştirilen sistemin şeması Şekil 17'de verilmiştir.

Şekil 17.Yüz tanıma ve etiketleme modülünün entegrasyonu

Son dönem gerçekleştirilen entegrasyon işlemleri sonucunda oluşan sisteme METU Advanced Video Information System (MAVIS) adını verdik.Sisteminin son halinin şeması ise Şekil 18'de gösterildiği gibidir. Bu sistemin tanıtımına http://mavis.ceng.metu.edu.tr adresinden ulaşılabilmektedir.



Şekil 18.MAVIS – Genel Mimari

7. Sonuçlar

Bu projede, ontoloji tabanlı, anlambilimsel içeriğe yönelik etiketleme ve sorgulama yapılmasına olanak sağlayan bir video bilgi yönetim sistemi çatısı geliştirilmiştir. Sunulan sistem, MPEG-7 ontolojisi tabanlıdır ve bu sisteme, diğer MPEG-7 ontolojisi uyumlu sistemlerle ortak bir dil üzerinden haberleşebilme ve birlikte çalışabilirlik yeteneği kazandırmıştır. Geliştirilen sistemde, ontoloji tabanlı kavramsal sorgulama, uzay-zamansal sorgulama, bölgesel ve zaman tabanlı sorgulama kabiliyetleri basit sorgu tipleri olarak gerçekleştirilebilmektedir. Bunların yanı sıra, basit sorguların "(", ")", "AND" ve "OR" operatörleri ile birleştirilmesiyle birleşik sorgular oluşturulabilmektedir. Tüm bu sorgu tipleri için, sistem, hem genel hem de videoya özel sorgulama yapabilmeyi desteklemektedir. Bu sayede, kullanıcıya video veritabanındaki tüm videolar üzerinde arama ve veri erişimi kabiliyeti sağlanmasının yanı sıra sadece ilgilenilen belirli bir video üzerinde arama yapma kabiliyeti de sunulmaktadır. Sorgular hem form-tabanlı hem de doğal dille sorguya izin veren arayüzlerle yapılabilmektedir.

Metin bilgisinden yardımcı veri çıkartma alanında yapılan çalışmalarla belli alanlardaki videolar için metin bilgisi kullanılarak yardımcı veri çıkarmak, videoları sınıflandırmak ve özetlemek mümkün olmuştur. Ayrıca hem metin bilgisi hem video analiz yöntemlerinin birleştirilmesi çalışmaları futbol alanında güzel sonuçlar vermiştir. Bu konularda yapılan çalışmalar uluslararası konferans ve dergilerde yayınlanmıştır.

Video analiz yöntemleriyle videoların yarı-otomatik etiketlenmesi kapsamında yapılan yüz bulma, tanıma ve etiketleme çalışmaları ile bu amaç için en elverişli algoritmalar belirlenmiş ve uygulanmıştır. Elde edilen sonuçlara göre, video bilgi yönetim sistemine gerekli girdileri

sağlayabilecek, yüzleri yarı-otomatik bir şekilde tanıyıp etiketleyen, yaş ve cinsiyet sınıflandırması yapabilen bir modül geliştirilmiştir. Bu konuda yapılan çalışmalaruluslararası konferanslarda yayınlanmış ve iki dergi makalesi değerlendirmeye sunulmuştur.

Geliştirilen video yönetim sistemi çatısı modüler bir yapıdadır ve sistem ontoloji yönetimi, video etiketleme ve sorgu işleme alt modüllerinden oluşmaktadır. Bu nedenle yeni modüller kolayca eklenebilip, çıkarılabilmektedir. Proje kapsamında otomatik yüz etiketleme modülü, doğal dille sorgulama modülü,metin bilgisinden otomatik bigi çıkarım modülü sisteme entegre edilerek, çalışan bir prototip sistem geliştirilmiştir.Bu sisteme "METU Advanced Video Information System" adı verilmiştir.Entegre edilen sistem ve alt modüllerin demoları http://mavis.ceng.metu.edu.tr adresinde sunulmaktadır.

Bu projede temel olarak üç ana araştırma konusunda çalışmalar yapılmıştır. Bunlar çokluortam veritabanı sistemleri, video ve metinden bilgi çıkarımı ve öneri sistemleridir. Proje çalışmaları sonucunda bu alanlarda elde edilen kazanımlar yeni projeler oluşturmak için yeterlidir. Yeni projeler bu üç temel alanda daha da detaylı çalışmalar başlatabilir. Örneğin metinden yardımcı veri çıkarma çalışmaları çeşitli yönlerde ilerleyebilir. Farklı alanlar için yeni algoritmalar geliştirilebilir. Belgeseller yerine haberler ya da filmler üzerinde çalışmalar yapılabilir. Televizyon yayınlarından yardımcı veri çıkarılabilir. Video analizi ile metin bilgisi birlikte kullanılarak daha senkrone yardımcı veri çıkarılabilir. Video analizi içine ses bilgisi de eklenebilir.

Video içindeki yüzleri, nesneleri, insan vücudunu tanıma ve etiketleme çalışmaları daha da geliştirilip insan görsel zekasının sahip olduğu çıkarım (İng. Inference) ve tahmin (İng. Prediction) gibi kabiliyetlerin geliştirilmesine çalışılabilinir. Bu projede edinilen kazanımlar daha akıllı bir seviyede çalışabilen, insan görsel zekasını taklit ederek nesne/kişi/olay tespiti, tanıması, çıkarımı ve tahmini yapabilen bir sistemin geliştirilmesi için bir alt yapı oluşturmaktadır. Bu kapsamda ilerlemek için yeni yüksek lisans ve doktora tezleri başlatmış durumdayız.

Kişiselleştirme ve öneri sistemleri konusunda yapılan çalışmalar genelde deneysel çalışmalar olarak nitelendirilebilir. Proje süresince elde edinilen tecrübeleri kullanarak geliştirilen ontolojik öneri sistemleri konusundaki yüksek lisans tezi bitmek üzeredir. Bu alandaki kazanımlar video bilgi yönetim sistemine yeni içeriği, Internet'i kullanıcı tercihlerine göre otomatik olarak tarayıp, bulacak akıllı ajanların geliştirilmesinde kullanılabilir. Bu konuda da yeni bir doktora tezi başlamıştır.

Bu proje kapsamında 13 öğrenci burs aldı. 13 yüksek lisans tezi tamamlandı. Bunlardan başka 3 tane daha yüksek lisans tezinin Haziran 2011'de bitirilmesi planlanmaktadır. Proje çalışmalarını anlatan 11 tanesi uluslararası konferans yayını, 3 tanesi dergi yayını olmak üzere toplam 14 yayın yapıldı.Ayrıca 4 tane yeni makale değişik dergilere teslim edildi ve şu an değerlendirme aşamasındadırlar. Buna ek olarak bütün sistemive entegrasyonu anlatan bir dergi makalesi de şu an hazırlanma aşamasındadır.

Proje kapsamında dört yurtdışı bir yurtiçi seyahat yapıldı. ISCIS 2008 konferansı için İstanbul'a, eChallenges 2008 için Stockholm'e, ICSC 2009 konferansı için San Francisco'ya, ICME 2010 konferansı için Singapur'a,ISCIS 2010 konferansı için Londra'yagidildi. Proje konusundaki çalışmalarımızın tanıtılması sonucunda yeni ortaklıklar başlatıldı ve yeni projeler hazırlanmaktadır.

Referanslar

ALACA-AYGÜL, F.,*Natural language query processing in ontology basedmultimedia databases,*(Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

ALAN, Ö, Sabuncu, O., Akpınar, S., Cicekli, N.K., Alpaslan, F.N., Ontological Video Annotation and Querying System for Soccer Games, ISCIS, İstanbul, (2008).

ALEXE, B., Deselaers, T., Ferrari, V., What is an object ?, CVPR, CA, (2010).

ALLEN, J. F., Maintaining Knowledge About Temporal Intervals, Communications of ACM 26, (1983) pp: 832–843.

ARNDT, R., Troncy, R., Staab, S., Hardman, L., Vacura, M., COMM: designing a well-founded multimedia ontology for the web, In the Proceedings of the 6th International Semantic Web Conference (ISWC'2007), (2007).

BALUJA, S.,Seth, R., Sivakumar, D., Jing, Y., Yagnik, J., Kumar, S., Ravichandran, D., Aly, M., Video suggestion and discovery for youtube: taking random walks through the view graph. In the Proceedings of WWW, (2008).

BANERJEE, S., Pedersen, T.: An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet. In the Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLING-02) Mexico City, Mexico (2002).

BAY, A., Ess, T. Tuytelaars, Gool, L. V., Speeded-up robust features (SURF), CVIU, vol. 110, no. 3, (2008) pp. 346–359.

BAYAR, M.,Alan, Ö, Sabuncu, O., Akpınar, S., Cicekli, N.K., Alpaslan, F.N., Event Boundary Detection Using Audio-Visual Features and Web-casting Texts with Imprecise Time Information, in IEEE International Conference on Multimedia & Expo (ICME), (2010).

BOUGUET, J.,Pyramidal implementation of the Lucas-Kanade feature tracker: description of the algorithm, Technical report, OpenCV Document, Intel Microprocessor Research Labs, (2000).

DALAL, N., Triggs, B., Histogram of Oriented Gradients for human detection, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, (2005)pp: 886-893.

DEMİRDİZEN, G.,*An ontology-driven video annotation and retrieval system,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

DEMİRTAŞ, K.,*Automatic video categorization and summarization*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2009).

DEMİRTAŞ, K., Cicekli, N.K., Cicekli, I., Automatic Categorization and Summarization of Documentaries, *Journal of Information Science*, 36 (6),(2010), pp: 671–689.

DEMİRTAŞ, K., Cicekli N.K., Cicekli, I., Summarization of Documentaries, In Proceedings of the 25th Intl. Symposium on Computer and Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) Lecture Notes in Electrical Engineering (62), London, (2010), pp: 105-108.

EKENEL, H.K.,Stiefelhagen, R., Local Appearance Based Face Recognition Using Discrete Cosine Transform,Proceedings of the 13th European Signal Processing Conference (EUSIPCO), September, (2005).

ERYOL, E.,Probabilistic latent semantic analysis based framework for hybrid social recommender systems, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

EVERINGHAM, M., Van~Gool, L., Williams, C. K. I., Winn, J., Zisserman, A.,The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results, http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html.

GARCIA, R., Celma, O., Semantic Integration and Retrieval of Multimedia Metadata, In the Proceedings of the Knowledge Markup and Semantic Annotation Workshop, Semannot'05, (2005).

GÖKTÜRK, Ö.,*Metadata extraction from text in soccer domain*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2008).

GÖKTÜRK, Ö., Cicekli, N.K., Cicekli, I., Metadata extraction from text in soccer domain, ISCIS, İstanbul, (2008).

HUNTER, J.,Adding Multimedia to the Semantic Web - Building an MPEG-7 Ontology, In International Semantic Web Working Symposium (SWWS 2001), Stanford University, California, USA, July 30 - August 1, (2001).

KARA, S.,*An Ontology-Based Retrieval System Using Semantic Indexing,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

KARA, S.,Alan, Ö., Sabuncu, O., Akpınar, S., Çiçekli, N.K., Alpaslan, F.N., An Ontology-based Retrieval System Using Semantic Indexing.DESWeb, IEEE ICDE Workshop,(2010).

KARAMAN, H.,*A content based movie recommendation system empowered by collaborative missing data prediction,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010*).*

KATSIOULI P., Tsetsos V., Hadjiefthymiades S. Semantic video classification based on subtitles and domain terminologies, SAMT'07 Workshop on Knowledge Acquisition from Multimedia Content (KAMC), Genova, Italy, December, (2007).

LI, T.,Zhu, S., Ogihara, M., Using Discriminant Analysis for Multi-class Classification, Proceedings of theThird IEEE International Conference on Data Mining, (2003)pp: 589- 592.

LOWE, D., Object recognition from local scale-invariant features, ICCV '99, vol.2, (1999) pp: 1150–1157.

MIHALCEA, R., Tarau, P., TextRank - bringing order into texts. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Barcelona, Spain, (2004).

ÖZBAL, G.,*A content boosted collaborative filtering approach for movie recommendation based on a local and global user similarity and missing data prediction*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2009).

ÖZBAL, G., Karaman, H, Alpaslan, F.N., A Content Boosted Collaborative Filtering Approach ForMovie Recommendation Based On Local & Global Similarity and Missing Data Prediction, In Proceedings of the 25th Intl. Symposium on Computer and Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) Lecture Notes in Electrical Engineering (62), London, (2010) pp: 109-112.

ÖZBAL, G., Karaman H., Alpaslan, F.N., A Content-Boosted CollaborativeFiltering Approach for MovieRecommendation Based on Localand Global Similarity and Missing DataPrediction, to appear in Computer Journal, Published by Oxford University Press on behalf of The British Computer Society, doi:10.1093/comjnl/bxr001, (2011).

ÖZTÜRK, G., *A hybrid video recommendation systembased on a graph-based algorithm,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

ÖZTÜRK, G., Cicekli N.K., A Hybrid Video Recommendation System Using a Graph-Based Algorithm, to appear in the proceedings of the Twenty-fourth International Conference on Industrial, Engineering

and Other Applications of Applied Intelligent Systems (IEA/AIE 2011), June 28 - July 1, Syracuse, New York, USA, (2011).

SMEATON, A. F., Over, P., Kraaij, W., Evaluation Campaigns and TRECVid, in Proceedings of ACM International Workshop on Multimedia Information Retrieval, (2006) pp: 321–330.

ŞİMŞEK, A.,*Ontology-Based Spatio-Temporal Video Management System*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2009).

TARAKCI, H., CICEKLI, N.K., Ontological Multimedia Information Management System, eChallenges, Stockholm, (2008).

TARAKÇI, H.,*An ontology-based multimedia information management system*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2008)

TOUTANOVA, K., Klein, D., Manning, C., Singer, Y, Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network, In Proceedings of HLT-NAACL, (2003) pp: 252-259.

TSINARAKI, C., Polydoros, P., Christodoulakis, S., Interoperability support between MPEG-7/21 and OWL in DS-MIRF, In IEEE Transactions on Knowledge and Data Engineering (IEEE-TKDE), Special Issue on the Semantic Web Era, (2007).

TUNAOĞLU, D., Alan, Ö, Sabuncu, O., Akpınar, S., Cicekli, N.K., Alpaslan, F.N., Event Extraction from Turkish Football Web-casting Texts Using Hand-crafted Templates, IEEE International Conference on Semantic Computing, Berkeley, San Francisco, (2009).

VIOLA, P.,Jones, M., Rapid object detection using a boosted cascade of simple features, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, (2001)pp: 511-518.

WOLF, L., Hassner, T., Taigman, Y., Descriptor Based Methods in the Wild, Faces in Real-Life Images workshop at the European Conference on Computer Vision (ECCV), October,(2008).

YAPRAKKAYA, G.,*Face identification, gender and age groups classifications for the semantic annotation of videos,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

YAPRAKKAYA, G., Cicekli, N.K., Ulusoy, I., Gender and Age Groups Classifications for Semantic Annotation of Videos, In Proceedings of the 25th Intl. Symposium on Computer and Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) Lecture Notes in Electrical Engineering (62), London, (2010) pp: 227-230.

YILMAZTÜRK, M.C.,*Online and Semi-automatic Annotation of Faces in Personal Videos* (Kişisel Videolardaki Yüzlerin Çevrimiçi ve Yarı Otomatik İsimlendirilmesi), (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

YILMAZTÜRK, M., Ulusoy, I., Cicekli, N.K., Analysis of Face Recognition Algorithms for Online and Automatic Annotation of Personal Videos, In Proceedings of the 25th Intl. Symposium on Computer and Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) Lecture Notes in Electrical Engineering (62), London, (2010) pp: 231-236.

TÜBİTAK
PROJE ÖZET BİLGİ FORMU

Proje No: 107E234

Proje Başlığı: Kişisel Video Bilgi Yönetim Sistemleri için bir Çatı Geliştirilmesi

Proje Yürütücüsü ve Araştırmacılar:

Doç. Dr. Nihan Kesim Çiçekli

Prof. Dr. İlyas Çiçekli

Doç. Dr. Ferda Alpaslan

Y.Doç. Dr. İlkay Ulusoy

Projenin Yürütüldüğü Kuruluş ve Adresi: Ortadoğu Teknik Üniversitesi

Bilgisayar Mühendisliği Bölümü, Ortadoğu Teknik Üniversitesi, 06800, ANKARA

Destekleyen Kuruluş(ların) Adı ve Adresi: Ortadoğu Teknik Üniversitesi

Ortadoğu Teknik Üniversitesi, 06800, ANKARA

Projenin Başlangıç ve Bitiş Tarihleri: 1/2/2008 – 1/2/2011

Öz (en çok 70 kelime)

Bu projede, ontoloji tabanlı, anlamsal içeriğe yönelik etiketleme ve sorgulama yapılmasını mümkün kılan bir video bilgi yönetim sistemi çatısı önerilmektedir. Geliştirilen sistemde, ontoloji tabanlı kavramsal sorgulama, uzay-zamansal sorgulama, bölgesel ve zaman tabanlı sorgulama kabiliyetleri basit sorgu tipleri ya da birleşik sorgular olarak gerçekleştirilebilmektedir. Sorgular hem form-tabanlı hem de doğal dille sorguya izin veren arayüzlerle yapılabilmektedir. Ayrıca kullanıcı tercihlerini kullanan bir içerik öneri sistemi geliştirilmiştir.

Bu projede yardımcı verilerin otomatik çıkarılması için video analiz yöntemleri ve video ile birlikte gelen metinden bilgi çıkarım yöntemleri çalışılmıştır. Futbol, film ve haber bültenleri gibi açıklayıcı metinlerle birlikte gelen videoları otomatik etiketlemek için metinden bilgi çıkarım yöntemleri gerçekleştirilmiştir. Kameralardan ya da cep telefonlarından edinilen ve metin bilgisi olmayan videoların ise video analiz yöntemleri ile yarı-otomatik etiketlenmesi için algoritmalar geliştirilmiştir. Bu bağlamda insanların yaş ve cinsiyetlerinin kategorilendirilmesi de yapılmıştır. Ayrıca belgesellerin, altyazıları kullanılarak otomatik sınıflandırılması ve özetlenmesi de gerçekleştirilmiştir.

Anahtar Kelimeler:

Çokluortam veri yönetimi, metinden bilgi çıkarımı, anlambilimsel arama, öneri sistemleri, video analizi, doğal dille sorgulama

Fikri Ürün Bildirim Formu Sunuldu mu? **Evet** ☐ **Gerekli Değil** ☒

Fikri Ürün Bildirim Formu'nun tesliminden sonra 3 ay içerisinde patent başvurusu yapılmalıdır.

Projeden Yapılan Yayınlar:

[1] ALAN, Ö, Sabuncu, O., Akpınar, S., Cicekli, N.K., Alpaslan, F.N.Ontological Video Annotation and Querying System for Soccer Games, ISCIS, İstanbul, (2008).

[2] BAYAR, M.,Alan, Ö, Sabuncu, O., Akpınar, S., Cicekli, N.K., Alpaslan, F.N., Event Boundary Detection Using Audio-Visual Features and Web-casting Texts with Imprecise Time Information, in IEEE International Conference on Multimedia & Expo (ICME), (2010).

[3] DEMİRTAŞ, K., Cicekli, N.K., Cicekli, I., Automatic Categorization and Summarization of Documentaries, *Journal of Information Science*, 36 (6),(2010), pp: 671–689.

[4] DEMİRTAŞ, K., Cicekli N.K., Cicekli, I., Summarization of Documentaries, In Proceedings of the 25th Intl. Symposium on Computer and Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) Lecture Notes in Electrical Engineering (62), London, (2010), pp: 105-108.

[5] EROZEL, G., Cicekli, N.K., Cicekli, I., Natural language querying for video databases, Information Sciences, 178, (2008) pp: 2534–2552.

[6] GÖKTÜRK, Ö., Cicekli, N.K., Cicekli, I., Metadata extraction from text in soccer domain, ISCIS, İstanbul, (2008).

[7] KARA, S.,Alan, Ö., Sabuncu, O., Akpınar, S., Çiçekli, N.K., Alpaslan, F.N., An Ontology-based Retrieval System Using Semantic Indexing.DESWeb, IEEE ICDE Workshop,(2010).

[8] ÖZBAL, G., Karaman, H, Alpaslan, F.N., A Content Boosted Collaborative Filtering Approach ForMovie Recommendation Based On Local & Global Similarity and Missing Data Prediction, In Proceedings of the 25th Intl. Symposium on Computer and Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) Lecture Notes in Electrical Engineering (62), London, (2010) pp: 109-112.

[9] ÖZBAL, G., Karaman H., Alpaslan, F.N., A Content-Boosted CollaborativeFiltering Approach for MovieRecommendation Based on Localand Global Similarity and Missing DataPrediction, to appear in Computer Journal, Published by Oxford University Press on behalf of The British Computer Society, doi:10.1093/comjnl/bxr001, (2011).

[10] ÖZTÜRK, G., Cicekli N.K., A Hybrid Video Recommendation System Using a Graph-Based Algorithm, to appear in the proceedings of the Twenty-fourth International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE 2011), June 28 - July 1, Syracuse, New York, USA, (2011).

[11] TARAKCI, H., CICEKLI, N.K., Ontological Multimedia Information Management System, eChallenges, Stockholm, (2008).

[12] TUNAOĞLU, D., Alan, Ö, Sabuncu, O., Akpınar, S., Cicekli, N.K., Alpaslan, F.N., Event Extraction from Turkish Football Web-casting Texts Using Hand-crafted Templates, IEEE International Conference on Semantic Computing, Berkeley, San Francisco, (2009).

[13] YAPRAKKAYA, G., Cicekli, N.K., Ulusoy, I., Gender and Age Groups Classifications for Semantic Annotation of Videos, In Proceedings of the 25th Intl. Symposium on Computer and Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) Lecture Notes in Electrical Engineering (62), London, (2010) pp: 227-230.

[14] YILMAZTÜRK, M., Ulusoy, I., Cicekli, N.K., Analysis of Face Recognition Algorithms for Online and Automatic Annotation of Personal Videos, In Proceedings of the 25th Intl. Symposium on Computer and Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) Lecture Notes in Electrical Engineering (62), London, (2010) pp: 231-236.

TEZLER

1. ALACA-AYGÜL, F.,*Natural languagequeryprocessing in ontologybasedmultimediadatabases,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

2. DEMİRDİZEN, G.,*An ontology-driven video annotationandretrievalsystem,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

3. DEMİRTAŞ, K.,*Automatic video categorizationandsummarization*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2009).

4. ERYOL, E.,Probabilisticlatentsemanticanalysisbasedframeworkforhybridsocialrecommendersystems, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

5. GÖKTÜRK, Ö.,*Metadataextractionfromtext in soccer domain*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2008).

6. KARA, S.,*An Ontology-BasedRetrievalSystem Using SemanticIndexing,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

7. KARAMAN, H.,*A contentbasedmovierecommendationsystemempoweredbycollaborativemissing data prediction,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010*).*

8. ÖZBAL, G.,*A contentboostedcollaborativefilteringapproachformovierecommendationbased on a localand global usersimilarityandmissing data prediction*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2009).

9. ÖZTÜRK, G., *A hybrid video recommendationsystembased on a graph-basedalgorithm,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

10. ŞİMŞEK, A.,*Ontology-BasedSpatio-Temporal Video Management System*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2009).

11. TARAKÇI, H.,*An ontology-basedmultimediainformationmanagementsystem*, (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2008)

12. YAPRAKKAYA, G.,*Faceidentification, genderandagegroupsclassificationsforthesemanticannotation of videos,* (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

13. YILMAZTÜRK, M.C.,*Online and Semi-automaticAnnotation of Faces in PersonalVideos* (Kişisel Videolardaki Yüzlerin Çevrimiçi ve Yarı Otomatik İsimlendirilmesi), (Yüksek Lisans Tezi), Ortadoğu Teknik Üniversitesi Mühendislik Fakültesi, (2010).

YAYINLAR

1. ALAN, Ö, Sabuncu, O., Akpınar, S., Cicekli, N.K., Alpaslan, F.N.Ontological Video AnnotationandQueryingSystemfor Soccer Games, ISCIS, İstanbul, (2008).

2. BAYAR, M., Alan, Ö, Sabuncu, O., Akpınar, S., Cicekli, N.K., Alpaslan, F.N., EventBoundaryDetection Using Audio-Visual Featuresand Web-castingTextswithImprecise Time Information, in IEEE International Conference on Multimedia & Expo (ICME), (2010).

3. DEMİRTAŞ, K.,Cicekli, N.K., Cicekli, I., AutomaticCategorizationandSummarization of Documentaries, *Journal of Information Science*, 36 (6), (2010), pp: 671–689.

4. DEMİRTAŞ, K.,Cicekli N.K., Cicekli, I., Summarization of Documentaries, InProceedings of the 25th Intl. Symposium on Computerand Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) LectureNotes in ElectricalEngineering (62), London, (2010), pp: 105-108.

5. EROZEL, G.,Cicekli, N.K., Cicekli, I., Natural languagequeryingfor video databases, Information Sciences, 178, (2008) pp: 2534–2552.

6. GÖKTÜRK, Ö.,Cicekli, N.K., Cicekli, I., Metadataextractionfromtext in soccer domain, ISCIS, İstanbul, (2008).

7. KARA, S., Alan, Ö., Sabuncu, O., Akpınar, S., Çiçekli, N.K., Alpaslan, F.N., An Ontology-basedRetrievalSystem Using SemanticIndexing. DESWeb, IEEE ICDE Workshop, (2010).

8. ÖZBAL, G., Karaman, H, Alpaslan, F.N., A Content Boosted Collaborative Filtering Approach For Movie Recommendation Based On Local & Global Similarity and Missing Data Prediction, InProceedings of the 25th Intl. Symposium on Computerand Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) LectureNotes in ElectricalEngineering (62), London, (2010) pp: 109-112.

9. ÖZBAL, G., Karaman H., Alpaslan, F.N., A Content-Boosted Collaborative Filtering Approach for Movie Recommendation Based on Local and Global Similarity and Missing Data Prediction, to appear in Computer Journal, Published by Oxford University Press on behalf of The British Computer Society, doi:10.1093/comjnl/bxr001, (2011).

10. ÖZTÜRK, G.,Cicekli N.K., A Hybrid Video RecommendationSystem Using a Graph-BasedAlgorithm, toappear in theproceedings of theTwenty-fourth International Conference on Industrial, EngineeringandOther Applications of AppliedIntelligentSystems (IEA/AIE 2011), June 28 - July 1, Syracuse, New York, USA, (2011).

11. TARAKCI, H., CICEKLI, N.K., Ontological Multimedia Information Management System, eChallenges, Stockholm, (2008).

12. TUNAOĞLU, D., Alan, Ö, Sabuncu, O., Akpınar, S., Cicekli, N.K., Alpaslan, F.N., EventExtractionfromTurkish Football Web-castingTexts Using Hand-crafted Templates, IEEE International Conference on Semantic Computing, Berkeley, San Francisco, (2009).

13. YAPRAKKAYA, G.,Cicekli, N.K., Ulusoy, I., Genderand Age GroupsClassificationsforSemanticAnnotation of Videos, InProceedings of the 25th Intl. Symposium on Computerand Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) LectureNotes in ElectricalEngineering (62), London, (2010) pp: 227-230.

14. YILMAZTÜRK, M., Ulusoy, I., Cicekli, N.K., Analysis of FaceRecognitionAlgorithmsfor Online andAutomaticAnnotation of PersonalVideos, InProceedings of the 25th Intl. Symposium on Computerand Information Sciences (ISCIS), E. Gelenbe et al. (Eds.) LectureNotes in ElectricalEngineering (62), London, (2010) pp: 231-236.

Available online at www.sciencedirect.com
ScienceDirect
Information Sciences 178 (2008) 2534–2552
INFORMATION SCIENCES
AN INTERNATIONAL JOURNAL
www.elsevier.com/locate/ins

Natural language querying for video databases

Guzen Erozel [a], Nihan Kesim Cicekli [a], Ilyas Cicekli [b,*]

[a] Department of Computer Engineering, METU, Ankara, Turkey
[b] Department of Computer Engineering, Bilkent University, Ankara, Turkey

Received 13 November 2006; received in revised form 31 January 2008; accepted 6 February 2008

Abstract

The video databases have become popular in various areas due to the recent advances in technology. Video archive systems need user-friendly interfaces to retrieve video frames. In this paper, a user interface based on natural language processing (NLP) to a video database system is described. The video database is based on a content-based spatio-temporal video data model. The data model is focused on the semantic content which includes objects, activities, and spatial properties of objects. Spatio-temporal relationships between video objects and also trajectories of moving objects can be queried with this data model. In this video database system, a natural language interface enables flexible querying. The queries, which are given as English sentences, are parsed using link parser. The semantic representations of the queries are extracted from their syntactic structures using information extraction techniques. The extracted semantic representations are used to call the related parts of the underlying video database system to return the results of the queries. Not only exact matches but similar objects and activities are also returned from the database with the help of the conceptual ontology module. This module is implemented using a distance-based method of semantic similarity search on the semantic domain-independent ontology, WordNet.

Keywords: Natural language querying; Content-based querying in video databases; Link parser; Information extraction; Conceptual ontology

1. Introduction

The current technological developments offer convenient ways for storing and querying video files that include movies, news clips, sports events, medical scenes, security camera recordings, to people and researchers working in the areas of media, sports, education, health, security, and many others. There are mainly two problems in the implementation of video archives. The first problem is related to the modeling and storage of videos and their metadata. The second problem is how to query the content of the videos in a detailed and easy manner.

* Corresponding author. Tel.: +90 312 2901589; fax: +90 312 2664047.
E-mail addresses: guzen.erozel@tcmb.gov.tr (G. Erozel), nihan@ceng.metu.edu.tr (N.K. Cicekli), ilyas@cs.bilkent.edu.tr (I. Cicekli).

0020-0255/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2008.02.001

Unlike relational databases, spatio-temporal properties and rich set of semantic structures make it more complex to query and index the video content. Due to the complexity of video data, there have been many video data models proposed for video databases [1,2,8,10,15,16,22–24]. Some of the existing work use annotation based modeling. Some use physical level video segmentation approach [37,44], and some have developed object based modeling approaches which use objects and events as a basis for modeling the semantic information in video clips [1,30]. The object-oriented approach is more suitable to model the semantic content of videos in a more comprehensive way.

There have been several methods proposed to query the content of video databases in the literature. It is possible to divide these methods into mainly two groups: graphical interfaces and textual interfaces. In the graphical user interfaces, the user generates queries by selecting proper menu items, sketching graphs, drawing trajectories and entering necessary information with the help of a mouse like in WebSEEK [25], SWIM [43] and VideoQ [7]. These are in general easy to use systems but they are not flexible enough. On the other hand, textual interfaces that require the user to enter queries via SQL-like query languages or extensions to SQL are difficult to use, since the user has to learn the syntax of the language [10,26]. Other approaches for textual interfaces are not so flexible for the reason that Boolean operators or category-hierarchy structures are used for querying like in VideoSTAR [17] and VISION [27]. The most flexible method among all these approaches is the use of a natural language.

The aim of this paper is to present a natural language query interface over a content-based video data model which has spatio-temporal querying capabilities in addition to the basic semantic content querying. The video data model [22] identifies spatial properties of objects with rectangular areas (regions) resembling MBRs (minimum bounding rectangles). It is possible to compute and query spatial relationships between two rectangular areas, hence the objects covered by those rectangles. It is also possible to handle spatial relations such as left, right, top, bottom, top-left, top-right, bottom-left, bottom-right, as directional relations, and overlaps, equal, inside, contain, touch, and disjoint as topological relations. The model also supports querying the trajectory of an object given the starting and ending regions. The model allows us to perform spatio-temporal queries on the video and also provide the inclusion of fuzziness in spatial and spatio-temporal queries. Our video data model [22] previously had only a graphical query interface. Later, the system is integrated with a natural language interface in which the user can express queries in English. In this paper, we present this natural language query interface to the video database system. The capability of querying the system in a natural language instead of an artificial language can be exemplified with the following kinds of queries.

- *Find the frames where the prime minister meets the minister of foreign affairs.* (A journalist may be posing this kind of query frequently.)
- *Show all intervals where the goals are scored.* (This query may be used in a sports event archive.)
- *Show all cars leaving the parking lot.* (A security camera recording can be queried in this fashion.)

Our natural language interface can handle such queries and other forms of queries given in English.

There has been a considerable amount of work in querying the video frames in natural languages. They use syntactic parsers to convert the media descriptions (or annotations) and build semantic ontology trees from the parsed query [29]. However, these are usually application specific and domain-dependent (e.g. querying only the recordings of street cameras in SPOT [19] or querying only the parts of news broadcast in Informedia [18]). Not every system using natural language can capture high-level semantics. The video system Informedia which is using keyword-matching natural language interface, cannot answer detailed queries nor handle structures with attributes. In this paper we propose a general-purpose video database querying system by adding a natural language interface to a video data model [22]. Another contribution of the querying facility of the system is the usage of information extraction techniques in order to find the semantic representation of user queries [12,13]. In SOCIS system, the crime scene photographs are annotated with text and keywords are extracted to index the photos [11,31]. However, only spatial relations in images are extracted in that system. In our system, on the other hand, many other query types can be extracted from sentences and their semantic representations are mapped to the underlying video data model.

It is an important problem to match a given query with the underlying video data in the systems that use natural language interfaces. When natural language queries are parsed, the first aim is to extract the entities

that occur in the query and match them with entities in the database. However sometimes, an exact match cannot be obtained for the query from the database. For example, the user may query a *car* where a *car* entity does not exist but instead *Mercedes* and *Fiat* exist as video entities. In order not to reply with an empty result set, ontology-based querying is used after the parsing phase. The similarity between entities in the database and parsed entities from query is evaluated by using an *is-a* hierarchy. The root of the tree is semantically more generalized than the leaves. The highest similarity value of the entity is selected to be in the result. Therefore, a natural language interface that uses ontology-based querying returns close-match results in addition to exact matches [3,21].

Another important contribution of the natural language query interface presented in this paper is to perform an ontological search by using a domain-independent general-purpose ontology that holds the ontological structures of English words. In querying, the system will not only search the given words but also perform semantic similarity search based on the ontological structure of the given words. For instance, when the user poses a query like "*Show all frames where vehicles are seen*", the system will be able to return videos which include *vehicles* and all semantically similar words such as *cars*, *buses* or *trains*. Although many different semantic similarity algorithms exist, none of the methods gives the best result. In our system, we preferred a combined method of an edge counting method of Wu and Palmer [41] with a corpus based method, because it gave the best results in our tests [12]. The ontology-based querying has previously been used in some other video systems, but these systems construct their own ontology that needs to be changed whenever the domain changes [29]. In this paper, syntactic parsing with a general lexicon and domain-independent ontology search are preferred for the flexibility of the developed interface. Hence, no additional dictionary or semantic similarity algorithm is needed when the domain and the video data entities change.

The rest of the paper is organized as follows: The related work is given in Section 2. In Section 3, the video data model that is used as the basis of this paper is summarized by introducing the types of queries supported by the model. The proposed system maps the given English queries into their semantic representations. The semantic representations are built from the output of the parsing module, by the information extraction module of the system. Section 4 describes the extraction module and the parsing technique used in query processing. Section 5 presents the details of the ontology-based querying that provides close-match results. In Section 5, we also explain the expansion of the semantic representations that are extracted from the natural language processing module. We give evaluation results for ontology-based searching in Section 6. Finally, Section 7 presents the conclusions and future work.

2. Related work on natural language query processing

A natural language interface is desirable to query the content of videos, in order to provide a flexible system where the user can use his/her own sentences for querying. The user does not have to learn an artificial query language, which is a major advantage of using a natural language in querying.

Although natural language interfaces provide the most flexible way of expressing queries over complex data models, they are limited by the domain and by the capabilities of parsers. The main issue is the conversion of a given natural language query into the semantic representation of the underlying query language. This process is not a simple task and different NLP techniques can be employed in order to map queries into their semantic representations. Before we describe NLP techniques that are used in our system, we review the related work in this section.

2.1. Natural language querying over databases

Early studies of natural language query processing depend on simple pattern-matching techniques. These are simple methods that do not need any parsing algorithm. SAVVY [5] is an example of this approach. In this system, some patterns are written for different types of queries and these patterns are executed after the queries are entered. For example, consider a table consisting of country names and their capitals. Suppose that a pattern is written as "*Retrieve the capital of the country if the query contains the word 'capital' before a country name*". Then the query "*What is the capital of Italy?*" will answer "*Rome*" as the result. However, since the results of this technique were not satisfactory, more flexible and complex techniques have been investigated.

The method used in the system LUNAR [39] supports a syntax-based approach where a parsing algorithm is used to generate a parse tree depending on user's queries. This method is especially used in application-specific database systems. A database query language must be provided by the system to enable the mapping from parse tree to the database query. Moreover, it is difficult to decide the mapping rules from the parse tree to the query language (e.g. SQL) that the database uses.

The system LADDER [5] uses semantic grammars where syntactic processing techniques and semantic processing techniques are used together. The disadvantage of this method is that semantic approach needs a specific knowledge domain, and it is quite difficult to adapt the system to another domain. In fact, a new grammar has to be developed when the system is configured for a different domain.

Some intermediate representation languages can be used to convert the statements in natural language to a known formal query language. MASQUE/SQL [4] is an example for this approach. It is a front-end language for relational databases that can be reached through SQL. User defines the types of the domain which database refers using an *is-a* hierarchy in a built-in domain-editor. Moreover, words expected to appear in queries with their logical predicates are also declared by the user. Queries are first transformed into a Prolog-like language LQL, then into SQL. The advantage of this technique is that the system generating the logic queries is independent from the database and therefore, it is very flexible in domain replacements.

2.2. Natural language techniques over video databases

Because of rich set of semantic structures and spatio-temporal properties in video data models, it is more complex to support querying in video databases. Natural language querying systems should be able to handle more complex query structures. This means that NLP techniques used in video databases should be more sophisticated so that queries can be mapped into the underlying query language. Syntactic parsers can be used to parse the given natural language queries, mapping systems can be used to map queries into their semantic representations, and ontologies can be used to extend the semantic representations of the queries.

The video system, SPOT [19], can query moving objects in surveillance videos. It uses the natural language understanding in the form of START (a question-answering system) [20], which has an annotation based natural language technique. Annotations which are English phrases or sentences, are stored in the knowledge base. The phrases are used to describe question types and information segments. Queries are syntactically parsed to match with these annotations. When a match is found between annotations and parsed query phrases, the segment in the annotation is shown to the user as a result. In SPOT, a track is the basic unit of data, which traces the motion of a single object in time. The queries are translated into symbolic representations that tracks are formulated. These representations are also in the sense of matching the annotations in the knowledge base. However, this system is incapable of capturing high-level semantics of the video content.

In [29], media data in the query is extracted into *description data* by using a matcher tool that uses the lexicon. These descriptions are semantically parsed with a domain specific lexicon in order to be matched with the data in the database. This method is used for exact matching. However, since in natural language, same descriptions can have different semantics, approximate matching is performed in the system by using semantic network model. When the query is parsed, the semantic representation is translated into a semantic network in which nouns and the actions in the query are the major nodes. There are also domain-dependent verb and noun hierarchies stored in the knowledge base of the system. The semantic networks are tried to be matched node by node according to the hierarchies. Weights are used in the hierarchy trees to enable better matching results. The main drawbacks of the system are the difficulty of the weights for approximate matches and the usage of a domain-dependent lexicon.

Informedia [18] uses a natural language interface in searching a news-on-demand collection. When a user poses a natural language query, the system searches and retrieves the best 24 news stories that match the query. Text summary headlines appear for the selected stories and the user can select the story that he is most interested. Both text headlines and video "skims" are generated by extractive summarization. All stories are scanned for words that have a high inverse document frequency in order to determine distinguishing stories. The major concern in this study is to search the text annotations by using summarization techniques. They mainly use a keyword-based search engine to find the video clips. Their approach is quite different from ours, since we store the video content using a video data model not as text annotations.

VideQA [42] uses a natural language interface in a question-answering system on news videos. In VideQA, a short question is mapped into one of the eight general question classes with a rule-based question classifier. In order to cope with the imprecise information in short questions, they also use WordNet to expand short questions.

In [14], a natural language query processor based on conceptual analysis is described. Their conceptual analyzer first tries to find nouns in the given query and then tries to fill the templates induced by verbs with the found nouns. A filled template represents the semantic structure of a given natural language query. They use a domain-dependent lexicon for nouns and verbs.

BilVideo [10] is expanded to support a natural language interface [23]. Natural language queries are mapped into Prolog-based fact representations. Since this system does not support ontology-based querying, it is not possible to get close-match results.

Zhang and Nunamaker use natural language processing techniques in video indexing [43]. They use a natural language approach to a content-based video indexing to find video clips. Their technique is similar to information retrieval techniques. They did not use any ontology to find close-match results.

The natural language query interface described in this paper uses a wide-coverage shallow parser for English to parse the given queries. Then, the information extraction module is used to map the parsed queries into the underlying formal query forms. Since the coverage of the parser and the information extraction module is high, the system can handle a lot of different query forms. These two steps can be seen as first figuring out the question template from the query and then filling in this template. Later, the found template is expanded using WordNet which is a wide-coverage domain-independent ontology in order to handle approximate matches.

Domain dependence is an important subject to consider for every natural language interface method. Domain dependency should be kept to minimum in order to enable a more flexible system when deciding on a technique to implement a natural language interface. The systems that use domain-dependent ontology only, need different ontologies for databases in different domains. In our system, however, we use WordNet which is a wide-coverage domain-independent ontology for English in order to provide maximum flexibility in querying.

3. Video data model

The video data model used in this paper is a content-based spatio-temporal video data model [22]. The basic elements of the data model are *objects*, *activities* and *events*. The video clip is divided into time-based partitions called *frames*. Objects are real world entities in these frames (e.g. book, Elton John, football, etc.). They can have properties (or attributes) like name and quantifiers (size, age, color, etc.). Activities are verbal clauses like playing football, singing, etc. Events are detailed activities that are constructed from an activity name and one or more objects with some roles. For instance *John plays football*, and *the cat is catching a mouse* are events.

Frame sequences contain a set of continuous frames that include any semantic entity like an object, an event, etc. Each entity in the video data model is associated with a set of frame sequences in which they occur. Frame contents can be queried by giving the object/event or activity of interest, and the system will return the relevant frames with the display option.

The basic type of queries supported in our system is occurrence queries. Occurrence queries are used to retrieve frames in which a given object, event or an activity occurs. In addition, it is possible to retrieve objects, events and activities occurring in a given frame (time) interval.

The support for spatio-temporal queries is the main concern in this video data model. Spatial properties contain the location of an object in the video frame. It is more difficult to represent the spatial relationships between two objects in a video than images since video data has time-dependent properties. In this model, a two-dimensional coordinate system is used for spatial properties of objects. The most preferred method to define the location of an object is to use an MBR (minimum bounding rectangle) which is an imaginary rectangle that bounds the area of an object at the minimum level. With the spatial properties added to the model, temporal properties are combined with them by defining region-interval tuples for objects. Thus, it is possible to define and query spatio-temporal relationships in a frame sequence between any two objects. A rule base covering the relations *top, bottom, right, left, top-right, top-left, bottom-right, bottom-left*, etc. is defined to help calculations of spatial relationships. Since the objects may move in a given interval, the spatial relationships may change over time. For instance, the *cat is to the left of the table* may change in a given time interval. This problem introduces fuzzy definitions of spatial relations in the model [22].

Fig. 1. Graphical query interface of the video database system.

The relative positions of two objects or an object's own position in the frame can be queried using spatial relations. The spatial relations between objects can be fuzzy since the objects may be moving in a video stream. The data model incorporates fuzziness in the querying of spatial relations by introducing a threshold value in their definition. Temporal properties are given as time intervals described in seconds and minutes. In the implementation, spatial queries are called regional queries; temporal queries are called interval queries. It is also possible to query the trajectory of a moving object. Starting from one region, an object's trajectory can be queried if its positions are adjacent up to an ending region in consecutive video frames.

A graphical user interface is used to query the videos in a previous implementation of the video database system. Pull-down menus and buttons are used to select objects, events, activities and spatial relations to express a query as seen in Fig. 1. When a spatial relation is queried, related objects, the spatial relation and also a threshold value are chosen from the drop down lists, and the type of the query must be selected using buttons. However, this interface has not been very flexible. Therefore, we have decided to use a natural language interface for querying the video contents.

4. Query processing

The idea is to map English queries into their semantic representations by using a parser and an information extraction module. The semantic representations of queries are fed into the underlying video database system to process the query and show the results. The main structure of the system is given in Fig. 2.

4.1. Semantic representations of queries

In order to extract the semantic representation of a query, it is sufficient to find the type of the given query and its parameters. The structure of the semantic representations is similar to the underlying data model structures. Therefore, in order to obtain the semantic representation of a query, we should be able to determine which parts of the query determines the type of the query and which parts correspond to the parameters of the query.

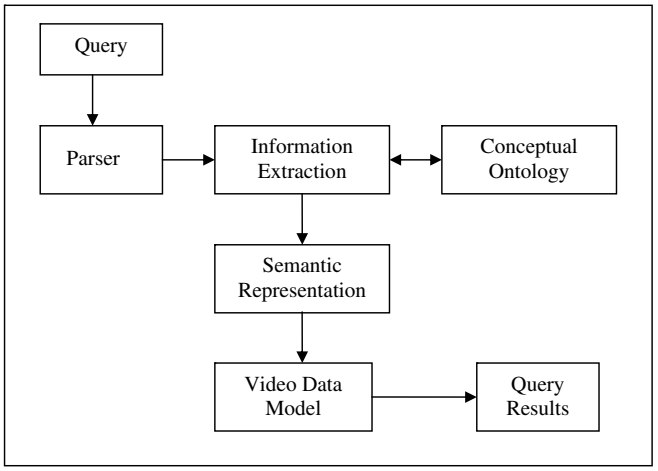


Fig. 2. The main structure of the natural language interface of the video data model.

Every query should include at least an object or an activity. Objects and activities are atomic particles that form an event. Objects can also have parameters like its name and attributes that qualify its name in the query, such as color, size, etc. In the implementation, the object representation is restricted to have only two attributes described by any adjectives in the query. Therefore, the atomic representation of an object is:

- *Object(name, attribute1, attribute2)*

where 'Object' is the predicate name used in the semantic representation of the query, 'name' is the name of the object, attributes (if they exist) are the adjectives used to describe the object in the query.

Activities are just verbs that are focused in the video frames. So they are also atomic and have representations like:

- *Activity(activity_name)*

where *Activity* is the predicate name used in the semantic representation of the query, *activity_name* is the activity verb itself.

Events are not atomic, because every event has an activity and the actors of that activity as its parameters. Thus, an event will be represented as:

- *Event (activity, object1, object2...)*

where *Event* is the predicate name, *activity* is the activity of this event, and the subsequent objects are the actors of this activity. The full semantic representation of an event occurrence query can be constructed only after the activity and objects are extracted.

There may be other kinds of semantic representations for spatial and temporal properties in the queries. Some of them are atomic structures such as coordinates and minutes, and some of them are relations between any two object entities. Regional queries include some rectangle coordinates to describe a region. During information extraction, the phrases representing the rectangles are converted to two-dimensional coordinates. Thus, regional semantic representation is:

- *Region(x1, y1, x2, y2)*

where *Region* is the predicate name that is used in the query semantic representation. $x1$ and $y1$ are the coordinates of the upper-left corner; $x2$ and $y2$ are the coordinates of the lower right corner of MBR.

Temporal properties are encountered as intervals in the query, so an interval is represented as follows:

- *Interval(start, end)*

where *start* and *end* are the bounding frames of the interval.

Spatial relations are extracted as predicates and the extracted objects involved in the spatial relations become the parameters of the predicates in the semantic representations. Semantic representations of the supported spatial relations are:

- *ABOVE (object1, object2, threshold)*
- *RIGHT (object1, object2, threshold)*
- *BELOW (object1, object2, threshold)*
- *LEFT (object1, object2, threshold)*
- *UPPER-LEFT (object1, object2, threshold)*
- *UPPER-RIGHT (object1, object2, threshold)*
- *LOWER-LEFT (object1, object2, threshold)*
- *LOWER-RIGHT (object1, object2, threshold)*

In these predicates, *threshold* value is used to specify the fuzziness in the spatial relations. The threshold value is between 0 and 1, and it indicates the acceptable correctness percentage for the relation. For example, the query "Find the frames, in which object A is seen to the left of object B, with a threshold value of 0.7" is a fuzzy spatial relationship query. The system finds the frames in which A and B occurs and regions of A and B satisfies the spatial relationship *LEFT* with at least 70% correctness, and these frames are returned as the result of this fuzzy query.

Supported query types and their semantic representations are given in Table 1. Each query in Table 1 has a different semantic representation with a different set of parameters. Hence, the data to be extracted depend on the type of the query. The semantic representations of the parameters are extracted, and they are combined to get the final semantic representation of the query.

4.2. Parsing queries

A syntactic parser is needed to extract information from the user query. We only need specific kinds of word groups (like objects, activities, start of the interval, etc.) to obtain the semantic representations. A light-parsing algorithm such as shallow parser [38], chunk parser [33] and link parser [28,36] is enough for our purposes, since there is no need to find the whole detailed parse tree of a query. We have chosen to use a link parser to parse given queries in our implementation, because of its ability to give more accurate results. Another advantage of this parser is to have the ability to get the grammatical relations between word groups, and these relations are used in the extraction of semantic representations.

Link parser is a kind of light parser which parses one English sentence at a time [28,36]. When a sentence is given as an input to the parser, the sentence is parsed with *linkages* using its grammar and its own word dictionary. As described in [28,36], a link grammar links every word in the sentence. A link is a unit that connects two different words. The sentence can be described as a tokenized input string by links which are obtained by the sentence splitter. When the sentence is parsed, it is tokenized with linkages (a group of links that do not cross). In the following example, Ds is a link that connects the singular determiner with its noun.

```
+---Ds---+
a        cat
```

A determiner (here it is a) must satisfy a Ds connector to its right. A single noun (here it is cat) must satisfy a Ds connector to its left. When the connectors are plugged, a link is drawn between a word pair.

When a natural language query is parsed by the link parser, the output of the parser includes the linkage information between the words of the query. For instance in Fig. 3 no two links cross and no words are left as unlinked. The links are represented in capital letters. The semantics of the links are as follows:

O	connects transitive verbs to objects
D	connects determiners to nouns
M	connects nouns to post-nominal modifiers

Table 1
Query types supported by the system, semantic representations and their examples

Query types	Semantic representations of queries	Query examples in natural language	Semantic representation of the examples
Elementary object queries	RetrieveObj (objA): *frame_list*	Retrieve all frames in which John is seen	• RetrieveObj (Obj_A): *frames.* • Obj_A (John, NULL, NULL)
Elementary activity type queries	RetrieveAct (actA): *frame_list*	Find all frames in which somebody plays football	• RetrieveAct (Act_A): *frames* • Act_A (play football)
Elementary event queries	RetrieveEvnt (evtA): *frame_list*	Show all frames in which Albert kills a policeman	• RetrieveEvnt (Evnt_A): *frames* • Evnt_A (Act_A, Obj_A, Obj_B) • Act_A (kill) • Obj_A (Albert, NULL, NULL) • Obj_B (policeman, NULL, NULL)
Object occurrence queries	RetrieveIntObj (intervalA): *object_list*	Show all objects present in the last 5 min in the clip	• RetrieveIntObj (Int_A): *objects* • Int_A($x - 5$, x). [x: Temporal length of video]
Activity type occurence queries	RetrieveIntAct (intervalA): *activity_list*	Retrieve activities performed in the first 20 min	• RetrieveIntAct (Int_A): *activities* • -Int_A (0, 20)
Event occurrence queries	RetrieveIntEvt (intervalA): *events_list*	Find all events performed in the last 10 min	• RetrieveIntEvt(Int_A): *events* • Int_A($x - 10$, x). [x: Temporal length of video]
Fuzzy spatial relationship queries	RetrieveObj_ObjRel (rel,threshold): *frame_list*	Find all frames in which Al Gore is at the left of the piano with the threshold value of 0.7	• RetrieveObj_ObjRel (LEFT, 0.7): *frames* • LEFT (Obj_A, Obj_B) • Obj_A (Al Gore, NULL, NULL) • Obj_B (piano, NULL, NULL)
Object interval queries	RetrieveIntervalofObj (objA): *interval_list*	When is Mel Gibson seen?	• RetrieveIntervalofObj (Obj_A): *intervals* • Obj_A (Mel Gibson, NULL, NULL)
Activity interval queries	RetrieveIntervalofAct (actA): *interval_list*	Retrieve intervals where somebody runs	• RetrieveIntervalofAct (Act_A): *intervals* • Act_A (run)
Event interval queries	RetrieveIntervalofEvnt (evtA): *interval_list*	Find all intervals where the cat is running	• RetrieveIntervalofEvnt(Evnt_A): *intervals* • Act_A (run) • Evnt_A (Act_A, Obj_A, NULL) • Obj_A (cat, NULL, NULL)
Regional(frame) queries	RetrieveObjReg (objA, region): *frame_list*	Show all frames where Bill is seen at the upper-left of the screen	• RetrieveObjReg (Obj_A, Reg_A): *frames* • Obj_A (ball) • Reg_A($x/2, 0, x, y$) [if coordinates of the frame's rectangle is considered as 0, 0, x, y]
Regional(interval) queries	RetrieveObjInt (objA, intervalA): *region_list*	Find the regions where the ball is seen during the last 10 min	• RetrieveObjInt (Obj_A, Int_A): *regions* • Obj_A (ball, NULL, NULL) • Int_A (Int_A($x - 10$, x) [x: Temporal length of video]
Trajectory queries	TrajectoryReg(objA, start_region, end_region): *frame_sequence*	Show the trajectory of a ball that moves from the left to the center	• TrajectoryReg (Obj_A, Reg_A, Reg_B): *frames* • Obj_A (ball, NULL, NULL) • Reg_A (0, 0, $x/2$, y) • Reg_B($x/4$, $y/4$, $3x/4$, $3y/4$) [if coordinates of the frame's rectangle is considered as 0, 0, x, y]

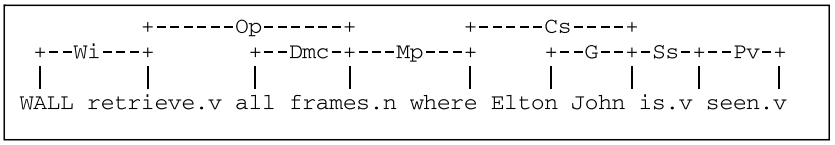


Fig. 3. The output of link parser for a sample object query.

C	connects subordinating conjunctions
G	connects proper nouns together in series
S	connects subject-nouns to finite verbs
P	connects forms of the verb "be" to various words
Wi	connects imperatives to the *wall* (beginning of the sentence)

The lowercase letters next to the representation of link types are the attributes of the links. For instance p means "plural" in Op, mc means "plural countable" in Dmc, s means "singular" in Cs, and v means "passive" in Pv.

4.3. Information extraction module

After a query is parsed with the link parser, the information extraction module forms the semantic representation of the query from the output of the parser. A similar technique is also used in crime scene reconstruction [11] which has been adopted from information extraction methodology used in SOCIS [31]. In [11], crime photos are indexed with spatial relations and scene descriptions by using the information extraction in an application domain. In our system, the information extraction module depends on a rule-based extraction defined on the linkages of a link parser. The aim is to extract word groups in the parsed query in order to map them to the semantic representations defined for *objects*, *activities, intervals, regions* and *spatial relations*. After extracting the basic terms, the full representation of the query is formed by determining the query type and the return value. Objects are nouns and their attributes are adjectives; activities are verbs, regions and spatial relations can be nouns or adjectives. So, the link types and the order of the links determine what it is to be extracted.

For each word group that can be extracted, one or more rules are implemented in a knowledge base. Once the query is parsed, special link types are scanned. Whenever a special linkage path is found, the rules written for finding out the structure (like object, query type, event, etc.) are applied to the path. For example, the following rule is one of the rules that are used to extract an activity:

- *Control the Cs link.*
- *If an Ss link follows this link and if the right-end of Cs is any word like "somebody, anybody, someone, etc.". Ss link's right word is the activity.*
- *If there is a following Os link, then the right-end of Os is a part of the activity (ex: playing football)*

For each query, the query type is first extracted from the parsed query. Then, the parts of the semantic representation are extracted. For instance, consider the query in Fig. 4. In this query, the word `interval` is the

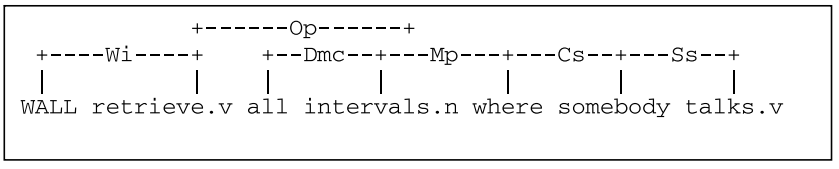


Fig. 4. The output of link parser for a sample interval query.

key word to determine the return value of the query's semantic representation, and it indicates that the query is an interval query. In order to determine the type of this interval query (i.e. *object interval query*, *activity interval query*, or *event interval query*), the right-end of `Mp` link is analyzed. For instance, when the rule above is applied, it is determined that there is an activity in the query. Therefore, this is an *activity interval query*, and the activity is *talk*. Thus, the following semantic representation is found by the information extraction module after all relevant rules are applied:

- RetrieveIntervalofAct (Act_A): *intervals*.
- Act_A (talk).

Let us consider the query in Fig. 3 as another example. `Op` linkage helps us to determine the return type of the query as *frames*. Then all atomic representations are searched tracing the linkage paths. When the `Cs` link is traced, an object, *Elton John*, is found. The tracing process is finished when no more atomic representations can be found. So depending on the keyword *frames* and only one object representation, the query type is decided to be an *Elementary Object Query*. Thus, we get the following semantic representation for the query.

- RetrieveObj (Obj_A): *frames*.
- Obj_A (Elton John, NULL, NULL).

Certain parts of the parsed query may not directly map to a part of the semantic representation. For instance, a numerical value can be entered either as a number or as a word phrase in a query. However in the data model it needs to be a numerical value. Therefore, a numerical value expressed as a word phrase should be converted into a number. This difficulty also arises in the extraction of regions. The regions are preferred to be described as areas relative to the screen like *left, center, upper-left*, etc. To map this data to the video data model, these areas should be converted into two-dimensional coordinates. Thus, the regions are represented as rectangles. So, the screen is assumed to be divided into five regions as *upper-right*, *upper-left*, *lower-left*, *lower-right* and *center*. The area in the query is matched with these regions. For example, for the word phrase "*right*" in the query, the coordinates of upper-right and lower-right are evaluated. A similar problem also occurs in interval queries. When the user phrases "the *last 10 min*", the beginning time must be evaluated to map with the video data. Therefore, the extraction algorithm is also responsible for these conversions. Some examples of these conversions are given in the last column of Table 1.

5. Ontology-based querying

In our video data model, all objects in the video database are annotated with nouns, and all activities are annotated with verbs. When a natural language query is converted into its semantic representation, the semantic representation of the query will contain at least one object or one activity. Similarly, each query object is represented with a noun and each query activity is represented with a verb. In order to find the video frames in which the query object appears, the noun representing the query object must match with one of the nouns representing the objects in the video frames. When the query object exactly matches a video object, we call this as an exact match.

When there does not exist any exact match between the object in the semantic representation and any of the objects in the video database, the system cannot return any results if it employs only an exact match method. This problem occurs when the user enters more generalized or more specific words in the query compared to the words used to represent the objects in the video database. For example, the user may query frames involving a 'car' but the database may include only the object 'sedan'. Although 'sedan' is also a 'car', the system cannot return any frames since the object 'car' does not exactly match with the object 'sedan'.

When there is no exact match between the object in the query and the objects in the video database, it may be desirable to return approximate results. A conceptual ontology can be used in order to return approximate results. An ontology is a kind of knowledge base that involves concepts and their definitions to be used for the semantics of the application domain [34,35]. A conceptual ontology can be used to

measure the similarity between two words representing the objects. For example, 'car' and 'sedan' are semantically similar words.

Although there are different types of ontologies, we have chosen WordNet [40] ontology, since it is the most general ontology used for semantic similarity of nouns and verbs. When user queries are processed, the most similar concepts are returned to the user by evaluating semantic similarities between objects in the video database and the object in the query using the WordNet ontology. Thus our system is able to return not only the exact matches but also approximate matches by finding semantically similar objects with the help of WordNet.

5.1. WordNet ontology

WordNet which is developed at the Princeton University is a free semantic English dictionary that represents words and concepts as a network. It organizes the semantic relations between words. The minimum set of related concepts is a 'synonym set' or 'synset'. This set contains the definitions of the word sense, an example sentence and all the word forms that can refer to the same concept. For instance, the concept 'person' has a synset of {person, individual, someone, somebody, mortal, human, soul}. All these words can represent the concept 'person'.

The WordNet has an is-a hierarchy model that can be viewed as a tree having one root. A parent node is a more general term than its children. For example, 'car' is parent of 'sedan' in the WordNet hierarchy. Although most of the nodes have a single parent, some of the nodes in WordNet can have more than one parent. For this reason, WordNet is not exactly a tree [9]. In WordNet, there are nearly 75.000 concepts defined in tree-like structures where nodes are linked by relations.

We use only noun and verb hierarchies in WordNet to measure similarity between objects and similarity between activities, respectively. In addition to noun and verb concepts, hierarchies for adjectives and adverbs are also included in WordNet. We have used version 2.0 of WordNet all throughout this work.

5.2. Measuring semantic similarity between words

The aim is to use the knowledge of domain-independent semantic concepts to get better and closer results for the query. The main issue in semantic similarity is getting more accurate results between two words: the annotation word for the stored video object and the word used in the query [3]. Here semantic similarity is measured between 'words'. That is no word sense disambiguation (WSD) method is used to find the sense of the query words and video object annotation words.

The user does not enter any information about senses of the words during the annotation of videos. Therefore, all senses of both the query word and the video object should be evaluated for semantic similarity. There have been many methods for evaluating the conceptual similarity, which can be divided into three groups:

- *Distance-Based Similarity*: The methods in this group depend on counting the edges in a tree or graph based ontology [6]. Finding the shortest path is important, but when the edges are not weighted, like in WordNet, other metrics, such as the density of the graph, link type and the relation among the siblings, should also be considered.
- *Information Based Similarity*: These methods use corpus in addition to the ontology in order to get statistical values [34]. Information content is a kind of measure showing the relatedness of a concept to the domain. If the information content is high, it means the concept is more specific to the domain. For example, *school bag* has higher information content while *bag* has lower information content. Implementing these methods is more difficult than evaluating path lengths.
- *Gloss Based Similarity*: Gloss is the definition of a concept. Gloss based similarity methods depend on WordNet to find the overlapping definitions of concepts and concepts to which they are related [32]. It has the advantage that similarity between different part of speech concepts can be compared. However, gloss definitions are too short to be compared with other glosses.

In processing a query, ontology processing should constitute a small amount of the workload. Therefore, the aim is to select the fastest and relatively the most accurate method. After certain experiments with different

conceptual similarity techniques, we selected a version of Wu and Palmer's method [41] to measure the similarity between query objects and objects in the video database. Wu and Palmer's method is a distance-based similarity method. The object in the query is compared with the objects in the video database to measure their similarities. The most similar objects (the objects whose similarity values are above a certain cut-off value) are selected by our conceptual similarity method in order to be used in the semantic representation of the query.

Our conceptual similarity algorithm can find the semantic similarity degree between a query word and a stored video object word using WordNet. Since both the query word and the video object word can have many word senses, we find the similarity values for all sense pairs. The sense pair with the highest similarity value is taken as the video object's similarity to the query word. This operation is done between the query word and every video object word. Similarity values are sorted in descending order and the resulting set of objects is returned according to a cut-off value. The similarity value between a sense of the query word and a sense of a video object word is between 0 and 1, and that similarity value depends on the distance between those two senses in WordNet hierarchy. If the distance between the senses is small, the similarity value will be higher (close to 1). On the other hand, the similarity value will be low when the distance is more.

5.3. Expanding semantic representations with ontology

The information extraction module creates semantic representations of queries by using only the words appearing in queries, which are the words used for exact matches. These words are the parameters of the semantic representations, and they represent objects or activities. In order to get approximate results, the semantic representations are expanded with new words that are semantically similar to the words appearing in queries. Our conceptual similarity algorithm finds the words that are semantically similar to a query word by using the WordNet ontology.

Whenever the query includes a word representing an object, or an activity, our conceptual similarity algorithm is invoked for that word, in order to get similar words representing objects or activities in the video database. The words representing the objects and activities are stored in a result set. There may be more than one element in the result set, therefore a new semantic representation is built for each element in the set. The following example illustrates the idea:

Query: *Retrieve all frames where a car is seen.*
Semantic representation before expansion:
RetrieveObj(ObjA): frames.
ObjA(car,NULL,NULL).
Semantic representations after expansion with the ontology:
RetrieveObj(ObjA): frames.
RetrieveObj(ObjB): frames.
RetrieveObj(ObjC): frames.
ObjA(car, NULL, NULL).
ObjB(jeep, NULL, NULL).
ObjC(sedan, NULL, NULL).

In this example, we assume that our similarity algorithm finds the similarity set {*car*, *jeep*, *sedan*} for 'car'. Each word in a similarity set represents an object available in the video database, and it is semantically similar to the query word according to the WordNet ontology. The objects *car*, *jeep* and *sedan* are the objects in the video database. *Car* is retrieved since it is an exact match. *Jeep* and *sedan* are retrieved by the similarity algorithm as the most similar objects to *car*. For each element in the result set, a new semantic representation is formed. Since the similarity set is an ordered set, the order of the semantic representations is the same as the order in this similarity set. This means that *jeep* is semantically closer to *car* than *sedan* according to our similarity algorithm.

Semantic representations are expanded for not only objects but also activities. A similarity set for a verb that represents an activity is also created by the similarity algorithm, and the set is used in the expansion of the semantic representation.

Events include an activity and one or more objects. When the query includes an event with more than one object, the number of formed semantic representations increases. An ontology search is performed for all objects and the activity. According to the established semantic similarities, a result set is obtained for the activity and the object(s). The elements in the result sets are ranked depending on their semantic similarity values. Then these elements are combined to form tuples to obtain only one result set. The similarity values of the elements are multiplied so that their product represents the semantic similarity value of the tuple. The tuples are ranked depending on their similarity values in the final result set. For each tuple in the last result set, a semantic representation for the event is formed in a similar way done for the objects. As an example, consider the following query:

Query: *Retrieve all frames where John is driving a car.*
Semantic representation before expansion:
RetrieveEvnt(EvntA): frames.
EvntA(ActA, ObjA, ObjB).
ActA(drive).
ObjA(John, NULL, NULL).
ObjB(car, NULL, NULL).

Let us abstract the representation as *RetrieveEvnt(drive, John, car): frames*. Some of the similarities that are found for the words in the query by our conceptual similarity method are given as the following similarity sets:

- Similarity set of *John* is {*John*}.
- Similarity set of *car* is {*car, jeep, sedan*}.
- Similarity set of *drive* is {*drive, operate*}.

From these sets, we can find six different semantic representations. These semantic representations are ordered with respect to their computed similarity values. The similarity value of a semantic representation depends on the similarity values of the words used in the semantic representation. Thus, we get the following six abstract semantic representations by using our conceptual similarity algorithm and the WordNet ontology.

Semantic representations (abstracts) after expansion with ontology:
RetrieveEvnt(drive, John, car): frames.
RetrieveEvnt(drive, John, jeep): frames.
RetrieveEvnt(drive, John, sedan): frames.
RetrieveEvnt(operate, John, car): frames.
RetrieveEvnt(operate, John, jeep): frames.
RetrieveEvnt(operate, John, sedan): frames.

Semantic representations are constructed in order to map the query to the query processing module of the video database. The video database system has necessary functions to execute each query. These functions require the same parameters as the ones in the semantic representations. Since there is a certain semantic representation for each query type, the representations can be mapped to the functions directly. Thus the video database and the natural language query interface are independent systems that communicate through the semantic representations.

Whenever a query is posed to the natural language interface, it returns not only the semantic representation of the query but also the type of the query to the video database. Since the parameters of the functions and the representations are the same, the system calls the appropriate functions to execute the query.

6. Evaluation

In order to evaluate the effectiveness of our ontology-based querying method, we created a test domain. This test domain consists of videos of a TV serial and videos obtained from a street camera. We especially

put videos from different domains in order to show that our method is domain-independent. Video objects are annotated with words by our annotation tool [22]. The total length of videos in the test domain was 280 min, and there were 710 distinct video objects that were annotated with nouns.

We prepared a set of queries and made a list of all words that were used in these queries. The total number of query words was 300. Some of these query words were the same as the words used to describe the objects in the database. In other words, they would correspond to exact matches. The 83% of the query words did not appear in the database. A human expert decided the correct frames for each query. The human expert marked not only the exact matches but also the video frames that could match semantically with the query word. For instance, for the query word 'vehicle', the human expert marked all video frames containing objects described by words such as 'car', 'bus', 'sedan', 'plane', 'ship' as correct answers. The human expert also considered the intended meanings of the query words in marking the correct answers. For instance, if the intended meaning of the query word 'bat' is baseball bat, all video frames containing a baseball bat are marked as correct answers but not the video frames containing the animal 'bat'.

Then we executed the prepared queries to see which video frames would be retrieved by the system. For each query, the returned answer set was compared with the correct answers prepared by the human expert, in order to evaluate the success of our querying mechanism. The accuracy was measured by using the well-known metric F-measure which was evaluated by the following formulas:

$$\text{precision} = \frac{\text{number of correct answers in the answer set}}{\text{number of all answers in the answer set}}$$

$$\text{recall} = \frac{\text{number of correct answers in the answer set}}{\text{number of all possible correct answers}}$$

$$F\text{-measure} = \frac{2 * \text{ precision } * \text{ recall}}{\text{precision} + \text{recall}}$$

Each query was expanded by using the words that are similar to the words appearing in the query. The amount of the expansion depends on the cut-off value used by our conceptual similarity algorithm. We got different expansions for different cut-off values and we got different answers for these different expansions. The test results for the test domain are given in Fig. 5. For the test domain, we got the best accuracy result (F-measure) when the cut-off value was 0.80. Our accuracy results are as follows when the cut-off value is 0.80:

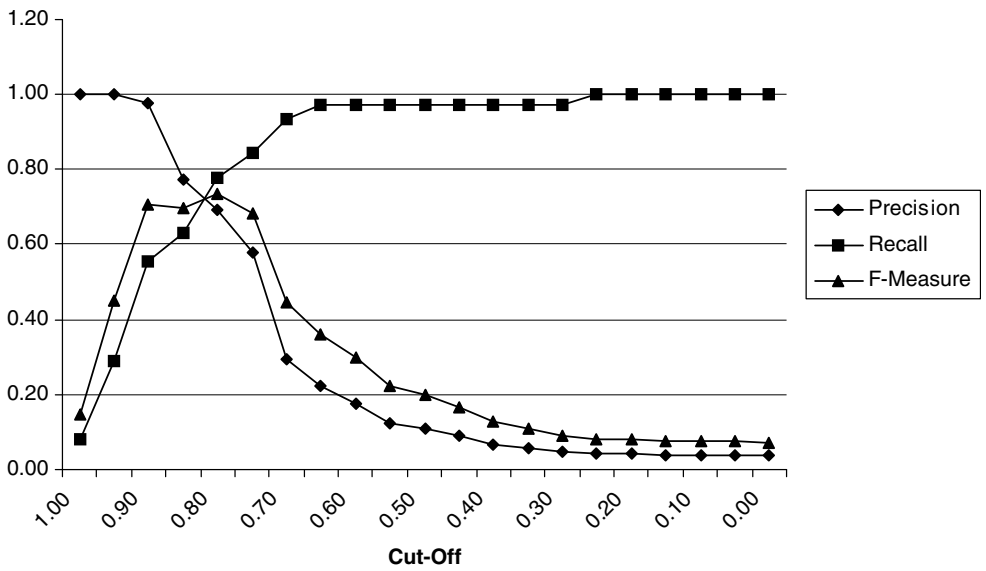


Fig. 5. Test results for the test domain (non-sensed).

F-measure 0.73
Precision 0.69
Recall 0.78

These results indicate that 69% of the results in the answer set are correct, and 78% of all possible correct answers appear in the answer set when the cut-off value is 0.80. These are the results with respect to the best *F*-measure value (0.73). Of course, the precision can increase when we increase the cut-off value, but the recall drops in this case. If the cut-off value comes close to 1.0, the precision becomes 1.0 too while the recall drops to its minimum (8%). The recall reaches to its maximum when the cut-off value comes close to 0, but the precision drops to its minimum (4%).

A noun can have multiple senses (meanings). Our similarity algorithm assumes that the query word is related to a video object word, if a sense of the query word is similar to a sense of the video object word. The query word can be related to an unintended sense of the video object word. In this case, they will be treated as similar words, and the video frame containing that video object word will be selected into the answer set. For example, the word 'mammal' is related with the animal sense of the word 'bat', and it is unrelated with the baseball bat sense of the word 'bat'. When the video frames containing a mammal are searched, the video frame containing a baseball bat can also appear as an incorrect result in the answer set, just because the video frame containing a baseball bat is annotated with the word 'bat' only.

When a video object is annotated with a word, the sense of the word is actually known by the annotator. If the annotator records the intended sense of the word together with the word, the precision and the recall will increase. We wanted to know how much improvement we would get in our results if the video objects were annotated with words together with their intended senses. We repeated the experiment with a test domain in which words were recorded together with their intended senses during annotation. When we executed test queries with this re-annotated test domain, the results were dramatically improved. We got the following results in this experiment when the cut-off value was 0.80:

F-measure 0.91
Precision 0.88
Recall 0.95

This experiment indicates that an extra effort in the annotation dramatically improves the result. The results of this experiment are given in Fig. 6.

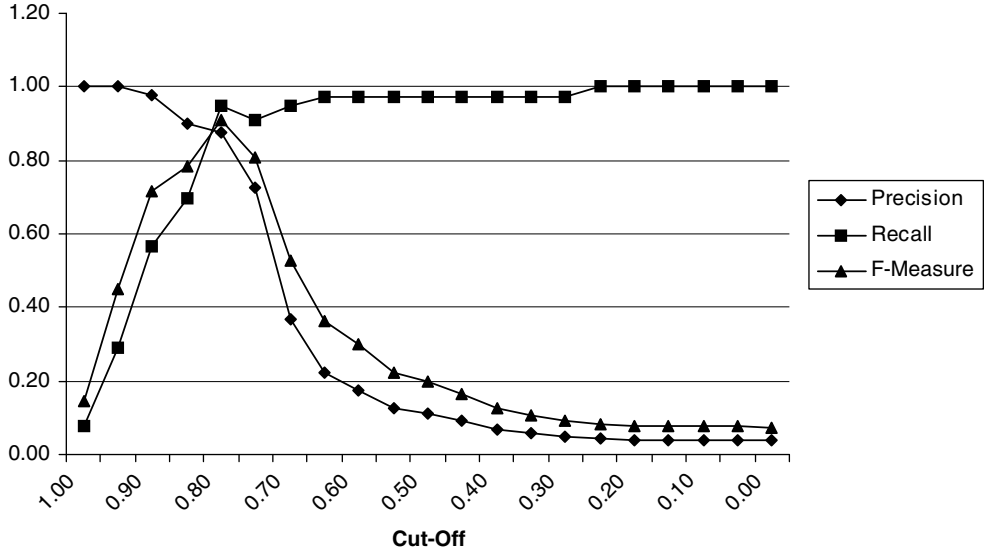


Fig. 6. Test results for the test domain (sensed).

In the literature, there are not too many systems that use a natural language interface with their accuracy results reported in their papers. The system described in [43] reports that their precision value for their tests is 0.372, and their recall value is 0.626. Of course, since their test domain and our test domain are not same, it may not be fair to compare the quantitative results of two systems.

7. Conclusion

The system described in this paper uses a natural language querying interface to retrieve information from a video database which supports content-based spatio-temporal querying. In order to implement the natural language interface, a light-parsing algorithm is used to parse queries and an information extraction algorithm is used to find the semantic representations of the queries. The core part of the extraction step is the detection of objects, events, activities and spatio-temporal relations. The semantic representation is constructed as the result of parsing the sentence with the link rules in a knowledge base. The semantic representation is used to map the query to the functions of the query processing module of the video database system. A conceptual ontology search is implemented as part of the natural language interface. Using the ontological structure, WordNet, the system retrieves the most similar objects or activities to the words given in the query. An edge-based method is combined with corpus-based techniques in order to get higher accuracy from the system. The semantic representations enriched with the ontology are sent to the video database system to call the appropriate query function.

In the current semantic representation of objects, an object can have only limited number of simple attributes. As a future extension, we are planning to add more complex attributes to describe the objects. Adding more complex attributes means that we have to deal with more complex noun phrases in the queries. The information extraction module will then be more complex for objects; however the querying capability of the system will have been increased. When the video database is expanded to handle compound and conjunctive query types, the extraction rules will be expanded to handle more complex queries.

Ontology related experiments show us that if the user annotates the video with not only plain words but also their senses in the WordNet, the accuracy rate of the natural language processing would increase. In a future study, we plan to expand the annotations in video database with senses attached to the words describing the entities. This extension can increase the annotation cost of videos, but it will increase the accuracy of the results.

Acknowledgements

This work is partially supported by The Scientific and Technical Council of Turkey Grant "TUBITAK EEEAG-107E234".

References

[1] B. Acharya, A.K. Majumdar, J. Mukherjee, Video model for dynamic objects, Information Sciences 176 (17) (2006) 2567–2602.
[2] S. Adali, K.S. Candan, S. Chen, K. Erol, V.S. Subrahmanian, The advanced video information system: data structures and query processing, Multimedia Systems 4 (1996) 172–186.
[3] T. Andreasen, H. Bulskov, R. Knappe, On ontology-based querying, in: H. Stuckenschmidt (Eds.), Proceedings of the 18th International Joint Conference on Artificial Intelligence, Ontologies and Distributed Systems (IJCAI 2003), Acapulco, Mexico, 2003, pp. 53–59.
[4] I. Androutsopoulos, G. Ritchie, P. Thanisch, MASQUE/SQL – An efficient and portable natural language query interface for relational databases, in: Proceedings of the Sixth International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Edinburgh, 1993, pp. 327–330.
[5] I. Androutsopoulos, G. Ritchie, P. Thanisch, Natural language interfaces to databases – an introduction, Journal of Natural Language Engineering 1 (1) (1995) 29–81.
[6] A. Budanitsky, G. Hirst, Semantic distance in WordNet: an experiment, application-oriented evaluation of five measures, in: Proceedings of NAACL 2001 – WordNet and Other Lexical Resources Workshop, Pittsburgh, 2001, pp. 29–34.
[7] S.F. Chang, W. Chen, H.J. Meng, H. Sundaram, D. Zhong, VideoQ: an automated content based video search system using visual cue, in: Proceedings of ACM International Conference on Multimedia'97, Seattle, WA, November 9–13, 1997, pp. 313–324.
[8] C. Decleir, M.S. Hacid, J. Kouloumdjian, Modeling and querying video databases, in: Proceedings of Conference EUROMICRO, Multimedia and Communication Track, Vastras, Sweden, 1998, pp. 492–498.

[9] A. Devitt, C. Vogel, The topology of WordNet: some metrics, in: P. Sojka, K. Pala, P. Smrz, C. Fellbaum, P. Vossen (Eds.), Proceedings of GWC 2004, Masaryk University, Brno, 2003, pp. 106–111.
[10] M.E. Donderler, E. Şaykol, U. Arslan, O. Ulusoy, U. Gudukbay, BilVideo: design and implementations of a video database management system, Multimedia Tools and Applications 27 (1) (2005) 79–104.
[11] F. Durupinar, U. Kahramankaptan, I. Cicekli, Intelligent indexing, querying and reconstruction of crime scene photographs, in: Proceedings of TAINN2004, Izmir, Turkey, 2004, pp. 297–306.
[12] G. Erozel, Natural language interface on a video data Model, MS Thesis, Department of Computer Engineering, METU, Ankara, 2005.
[13] G. Erozel, N.K. Cicekli, I. Cicekli, Natural language interface on a video data model, in: Proceedings of IASTED International Conference on Databases and Applications (DBA2005), Innsbruck, Austria, 2005, pp. 198–203.
[14] T.R. Gayatri, S. Raman, Natural language interface to video database, Natural Language Engineering 7 (1) (2001) 1–27.
[15] M.C. Hacid, C. Decleir, J. Kouloumdjian, A database approach for modeling and querying video data, IEEE Transactions on Knowledge and Data Engineering 12 (5) (2000) 729–750.
[16] R. Hjelsvold, R. Midtstraum, Modeling and querying video data, in: Proceedings of the 20th VLDB Conference, Santiago, Chile, 1994, pp. 686–694.
[17] R. Hjelsvold, S. Lagorgen, R. Midtstraum, O. Sandsta, Integrated video archive tools, in: Proceedings of ACM International Conference on Multimedia'95, San Francisco, CA, November 5–9, 1995, pp. 283–293.
[18] Informedia, Carnegie Mellon University, Available from: <http://www.informedia.cs.cmu.edu>.
[19] B. Katz, J. Lin, C. Stauffer, E. Grimson, Answering questions about moving objects in surveillance videos, in: Proceedings of AAAI Symposium on New Directions in Question Answering, Palo Alto, California, 2003.
[20] B. Katz, S. Felshin, D. Yuret, A. Ibrahim, J. Lin, G. Marton, A.J. McFarland, B. Temelkuran, Omnibase: Uniform access to heterogeneous data for question answering, in: Proceedings of the 7th International Workshop on Applications of Natural Language to Information Systems (NLDB 2002), 2002.
[21] R. Knappe, H. Bulskov, T. Andreasen, Perspectives on ontology-based querying, in: H. Stuckenschmidt (Eds.), Proceedings of the 18th International Joint Conference on Artificial Intelligence, Ontologies and Distributed Systems (IJCAI 2003), Acapulco, Mexico, 2003.
[22] M. Koprulu, N.K. Cicekli, A. Yazici, Spatio-temporal querying in video databases, Information Sciences 160 (2004) 131–152.
[23] O. Küçüktunç, U. Güdükbay, O. Ulusoy, A natural language-based interface for querying a video database, IEEE Multimedia – Multimedia at Work 14 (1) (2007) 83–89.
[24] T.C.T. Kuo, A.L.P. Chen, Content-based query processing for video databases, IEEE Transactions on Multimedia 2 (1) (2000) 1–13.
[25] H. Lee, A.F. Smeaton, J. Furner, User interface issues for browsing digital video, in: Proceedings of the 21st BCS IRSG Colloquium on IR, Glasgow, 1999.
[26] J. Li, M. Özsu, D. Szafron, V. Oria, Multimedia Extensions to Database Query Languages, Technical Report TR-97-01, Department of Computing Science, The University of Alberta, Alberta, Canada, 1997.
[27] W. Li, S. Gauch, J. Gauch, K. Pua, VISION: a digital video library, in: Proceedings of ACM International Conference on Digital Libraries (DL'96), Bethesda, MD, 1996, pp. 19–27.
[28] LinkParser, Available from <http://www.link.cs.cmu.edu/link>.
[29] V. Lum, D.A. Keim, K. Changkim, Intelligent natural language processing for media data query, in: Proceedings of International Golden West Conference on Intelligent Systems, Reno, Nevada, 1992.
[30] E. Oomoto, K. Tanaka, OVID: design and implementation of a video object database system, IEEE Transaction on Knowledge and Data Engineering 5 (4) (1993) 629–643.
[31] K. Pastra, H. Saggion, Y. Wilkis, Extracting relational facts for indexing and retrieval of crime-scene photographs, Knowledge-Based Systems 16 (5–6) (2002) 313–320.
[32] T. Pedersen, S. Banerjee, S. Pathwardan, Maximizing semantic relatedness to perform word sense disambiguation, University of Minnesota Supercomputing Institute, Research Report UMSI 2005/25, March, 2005.
[33] A.L. Ramshaw, M. Marcus, Text chunking using transformation-based learning, in: Proceedings of the ACL Third Workshop on Very Large Corpora, 1995, pp. 82–94.
[34] P. Resnik, Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language, Journal Artificial Intelligence Research 11 (1999) 95–130.
[35] M.A. Rodriguez, M.J. Egenhofer, Determining semantic similarity among entity classes from different ontologies, IEEE Transactions on Knowledge and Data Engineering 15 (2003) 442–465.
[36] D. Sleator, D. Temperley, Parsing English with a link grammar, in: Proceedings of the Third International Workshop on Parsing Technologies, 1993.
[37] D. Swanberg, C.F. Shu, R. Jain, Knowledge guided parsing in video databases, in: W. Niblack (Ed.), Proceedings of SPIE, Storage and Retrieval for Image and Video Databases, vol. 1908, San Jose, California, 1993, pp. 13–24.
[38] T. Vanrullen, P. Blache, An evaluation of different shallow parsing techniques, in: Proceedings of LREC-2002, 2002.
[39] W.A. Woods, R.M. Kaplan, B.N. Webber, The lunar sciences natural language information system: Final report, BBN Report 2378, Bolt Beranek and Newman Inc., Cambridge, Massachussetts, 1972.
[40] WordNet 2.1, Available from: <http://wordnet.princeton.edu/online/>, 2005.
[41] Z. Wu, M. Palmer, Verb Semantics and lexical selection, in: Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, Las Cruces, New Mexico, 1994, pp. 133–138.

[42] H. Yang, C. Lekha, Y. Zhao, S. Neo, T. Chua, VideoQA: question answering in news video, in: Proceedings of the 11th ACM MM, Berkeley, USA, 2003, pp. 632–641.
[43] D. Zhang, J.F. Nunamaker, A natural language approach to content-based video indexing and retrieval for interactive e-learning, IEEE Transaction Multimedia 6 (3) (2004) 450–458.
[44] H.J. Zhang, C.Y. Low, S.W. Smoliar, J.H. Wu, Video parsing, retrieval and browsing: an integrated and content-based solution, in: Proceedings of ACM International Conference on Multimedia'95, San Francisco, CA, November 7–9, 1995, pp. 503–512.

Journal: COMJNL

Article id: bxr001

The following queries have arisen while collating the corrections. Please check and advise us on the below queries.

1.	Please check and reply for q6.	

doi:10.1093/comjnl/bxr001

A Content-Boosted Collaborative Filtering Approach for Movie Recommendation Based on Local and Global Similarity and Missing Data Prediction

GÖZDE ÖZBAL*, HİLAL KARAMAN AND FERDA N. ALPASLAN

Department of Computer Engineering, Middle East Technical University, 06531 Ankara, Turkey
**Corresponding author: gozbalde@gmail.com*

Most traditional recommender systems lack accuracy in the case where data used in the recommendation process is sparse. This study addresses the sparsity problem and aims to get rid of it by means of a content-boosted collaborative filtering approach applied to a web-based movie recommendation system. The main motivation is to investigate whether further success can be obtained by combining 'local and global user similarity' and 'effective missing data prediction' approaches, which were previously introduced and proved to be successful separately. The present work improves these approaches by taking the content information of the movies into account during the item similarity calculations. The comparison of the proposed approach with the original methods was carried out using mean absolute error, and more accurate predictions were achieved.

Keywords: recommender systems; user modelling; collaborative filtering; sparsity; Pearson correlation coefficient; Floyd–Warshall algorithm

Received 14 July 2010; revised 22 October 2010
Handling editor: Franco Zambonelli

1. INTRODUCTION

Recently, the Web has been expanding both in the size of the information space and in the number of the users of that space. As a result, the task of locating relevant information and navigating that space in order to make choices of good items has been becoming more and more difficult. This information overload has created a great interest in automated filtering, refinement and organized presentation of relevant information to the users. Such automated methods are used to locate and retrieve information with respect to a user's individual preferences. Also, efforts to rank and sort information based on user preferences have generated interest over the past few years. One key area of research to achieve this goal is targeted around recommender systems (RSs). In today's world, many systems and approaches make it possible for the users to be guided by the recommendations people provide about new items such as news, web pages, articles, books, music and movies.

Many of the recommendation strategies used today rely on the modelling of intrinsic attributes about each item (e.g. the keywords for a document or the genre of a movie) so that the items can be categorized, and the level of interest that a user has can be expressed in terms of these attributes. This knowledge is usually gathered over time, by monitoring and logging various user interactions with the system so that the knowledge base is updated dynamically. An RS usually combines the values (ratings) of the elements of every dimension according to some evaluation scheme before obtaining a recommendation value (rating) for an item.

Collaborative filtering (CF) is a recommendation method that automatically predicts the interest of an active user by collecting rating information from the similar users or items. The underlying assumption of CF is that the active user will prefer the items preferred by similar users [1]. This approach is based on the idea that tastes of people are not randomly

distributed, and there exist general trends and patterns within the tastes of a person and between groups of people. However, when the users in the system have rated just a few items in the collection, the probability of finding similar users will be reduced. Our study addresses this limitation, which is often called the sparsity problem, by means of a content-boosted CF approach applied to the task of movie recommendation. Our main motivation is to investigate whether further success can be obtained for handling the sparsity problem by combining the previously proven methods, namely, 'local and global user similarity' (LU&GU) [2] and 'effective missing data prediction' (EMDP) [1]. LU&GU aims to overcome the difficulty of making a prediction under sparse user data, whereas EMDP determines whether to predict the missing data by using the information of user, items or both. Additionally, an enhanced Pearson correlation coefficient (PCC) algorithm, where a significance weighting factor has been added to overcome the potential decrease of accuracy during user/item similarity computation, is used. These approaches were enhanced by taking the content information of the movies into consideration during the item similarity calculations. The new updated formulas are given in Section 4. We applied these methods to a movie recommendation system named ReMovender.

The remainder of this paper is organized as follows. First, we introduce the necessary background in Section 2 and related work in Section 3, respectively. Then, the prediction mechanism of the system is presented in Section 4. We describe the basics of the movie recommendation system, ReMovender, in Section 5. Section 6 is devoted to the evaluation of our approach. Finally, we report the concluding remarks and state the further research directions in Section 7.

2. BACKGROUND

There are many systems designed to cope with the various problems of RSs using different approaches. CF has been the most popular one among them for a long time due to its success in many cases. However, because of the sparsity problem, CF requires a considerable amount of rating information in order to be successful. Besides, CF suffers from the new item problem where it is not possible to make a rating prediction for the items that have not been rated by any user yet.

In order to provide recommendations using CF, the system first collects and maintains information about the user. This information includes specific interest of users in certain items and it is stored in separate profiles. Once all the user profiles have been collected, the active user's similarities with the remainder of the users are calculated. This calculation is system specific and it depends on the algorithm used. Then, the group of the users that are most similar to the active user is selected, and their ratings are combined to produce predictions. Predictions of ratings may typically lead to the presentation of a ranked or a top-n list of the most relevant items.

Most popular prediction methods that are widely adopted in commercial CF systems are memory-based approaches. Memory-based CF methods base rating predictions on the entire collection of previously rated items by users. This information is stored in the form of a user-item matrix. However, memory-based collaborative techniques work correctly only if a reasonable amount of reliable data about user preferences is available. Traditional CF methods may not achieve success when the user-item matrix data are sparse. Thus, two separate methods, LU&GU [2] and EMDP [1], have been used to attack the data sparsity problem with a stronger approach. Secondly, the positive impact of using content information in a CF approach is addressed by developing a content-boosted CF prediction technique.

In [1], it is stated that sparsity of the user-item matrix leads to inaccurate recommendations. In order to handle this problem, an enhanced PCC algorithm, which adds a parameter to overcome the potential decrease of accuracy during user/item similarity computation, is used. In addition, an EMDP algorithm, in which both user and item information is taken into consideration, is proposed. In this algorithm, some similarity thresholds for both users and items are set so that the prediction algorithm decides whether to make a prediction or not. This research also addresses how to predict the missing data by employing a combination of user and item information.

One important aspect to be considered in the CF method is the way similarity between the profiles of users is computed. There are several possible algorithms for this computation. The PCC is a method that is time-efficient and can achieve higher accuracy than other similarity computation methods. In user-based CF, PCC is employed to define the similarity between two users based on the items that they rated in common. On the other hand, the basic idea in similarity computation between two items i and j by using PCC is to first isolate the users who have rated both of these items and then apply a similarity computation technique to determine the similarity between the items. When PCC is used for the similarity calculations of both user and item, the possible similarity values range from -1 to $+1$ including 0. The larger the similarity value of two items is, the more similar the related items are.

In our approach, user ratings for an item are predicted by using the content of that item and the rating information of similar users and items. PCC is used for the calculation of user/item similarity. However, PCC overestimates the similarity of users who have rated a few items identically and who do not have similar overall preferences. A correlation significance weighting factor is added to devalue the similarity weights based on a small number of co-rated items.

Throughout the study in [2], the concept of local and global similarity based on surprisal-based vector similarity, and an application of the concept of maximin distance in graph theory are presented. Besides, a user-based CF framework based on these concepts is explained.

Surprisal-based vector similarity expresses the relationship between any two users based on the quantities of information (called surprisal) contained in their ratings. The intuition behind this method is that less common ratings for a specific item provide more discriminative information than the most common ones. As for the global similarity, it defines two users as similar if they can be connected through their locally similar neighbours. The approaches that make use of both user and item-based algorithms can employ the approach followed by [2] to replace the traditional user-based approach so that they can achieve a higher performance. In fact, we base our method of attacking the data sparsity problem on this assertion, and use a combination of EMDP, and LU&GU concepts in our prediction technique.

3. RELATED WORK

There are many systems designed to cope with the various problems of RSs using different approaches. CF has been the most popular approach used for RSs, due to its success in many cases. However, CF requires a considerable amount of rating information in order to be successful. Besides, CF suffers from the new item problem where it is not possible to make a rating prediction for the items that have not been rated by any user yet. For this reason, some researchers tend to combine the existing methods and generate hybrid methods to overcome these problems. The main idea behind hybrid recommendation techniques is that 'a combination of algorithms can provide more accurate recommendations than a single algorithm and disadvantages of one algorithm can be overcome by other algorithms'.

There is a survey of the landscape of actual and possible hybrid recommenders in [3] in which a novel hybrid RS, named EntreeC, is introduced. EntreeC is a system that combines knowledge-based recommendation and CF to recommend restaurants to users. It shows that semantic ratings obtained from the knowledge-based part of the system enhance the effectiveness of CF.

An overview of RSs is presented in [4], which classifies the recommendation methods into three main categories: content-based, collaborative and hybrid recommendation approaches. This paper also describes various limitations of current recommendation methods and discusses possible extensions that can improve recommendation capabilities to cover a broader range of applications. These extensions include, among others, an improvement of understanding of users and items, incorporation of the contextual information into the recommendation process, support for multi-criteria ratings and a provision of more flexible recommendations.

The approach used throughout [5] describes a framework that combines missing data scores with content-based recommendations in order to produce a hybrid recommendation system. In the first stage, personalized user agents produce recommendations for items with a content-based method. Then, a second agent models the likelihood that the user already finds this item interesting. This model of the missing data is combined with the personalized content agents to form a model of stacked agents using CF. Improved results over a baseline content model are obtained with the help of the proposed approach. Besides, since the system combines content-based methods with CF ones, it attacks the cold-start problem in an effective way and outperforms approaches that rely on pure CF.

The study in [6] presents a hybrid music recommendation method that solves the problems of both CF and content-based recommendation. The proposed method integrates both rating and content data by using a Bayesian network called an aspect model. Unobservable user preferences are directly represented by introducing latent variables that are statistically estimated.

The evaluation results show that this method outperforms the two conventional recommendation methods in terms of recommendation accuracy and artist variety. High recommendation accuracy is achieved by the reliable modelling of user preferences and the integration of rating and content data. Besides, the proposed system attacks the 'new item problem' by recommending reasonable pieces even if they have no ratings.

The movie recommendation system, called MoRe, uses a hybrid approach based on content-based and CF techniques [7]. The system uses switching and substitute techniques by monitoring certain parameters that trigger either a CB or CF prediction. Besides, an empirical comparison of the hybrid approach to the base methods of CF and CB is provided.

The system collects user ratings concerning movies on a one-to-five scale through the graphical user interface. In order to handle the new user problem, the user is asked to provide several ratings after registration. In this manner, the prediction process can be initiated by the system. The two versions of the hybrid algorithm are differentiated by the parameter that controls the switch from the CF to the CB method. A web crawler is used in order to seek the movie description features from the website of an internet movie database. The system creates the set of most similar movies for all available movies at an off-line phase in order to speed up run-time predictions. The size of the neighbour set is determined by the system administrator.

In order to make movie recommendations, CF uses the ratings matrix, whereas the content-based predictor uses mainly the movie data files. As for the hybrid methods, they make use of both CB and CF engines. Although the system is able to produce recommendations based on more than one method, only one method, which is specified by the administrator, is applied at any given time.

Users of this system receive the recommendations in a ranked list of movies where the prediction appears to the user in a 'five-star' scale, while users provide their feedback directly on the recommended movies.

Melville *et al*. [8] present a framework for combining content and collaboration. Personalized suggestions through CF are maintained by using a content-based predictor to enhance

existing user data. Data sparsity and first-rater problems, which are the drawbacks of the CF systems, are overcome by exploiting content information of the items already rated. Basically, content-based predictions are used to convert a sparse user ratings matrix into a full ratings matrix, and then CF is used to provide the recommendations. The content information of each movie is collected from IMDb by using a simple crawler and the content information of each movie is represented as set of slots (features), each of which is simply a bag of words. The features that are used for this system are movie title, director, cast, genre, plot, summary, plot keywords, user comments, external reviews, newsgroup reviews and awards. The conducted experiments show that content-boosted CF performs better than a pure content-based predictor, pure collaborative filter and naïve hybrid approach. A naïve Bayesian text-classifier, which is used as a content-based predictor, is considered to be not ideal, since it disregards the fact that classes represent ratings on a linear scale. This problem is planned to be overcome by using a learning algorithm that can directly produce numerical predictions, such as logistic regression and locally weighted regression. In addition, the CF component is expected to improve by using a Clustered Pearson Predictor.

The idea of improving predictions by integrating the content information to the CF techniques constitutes the basis of our work. We developed the system ReMovender by using a hybrid approach that takes the content information of the movies into consideration during the item similarity calculations.

4. CONTENT-BOOSTED CF APPROACH

4.1. System components

The overall design of ReMovender can be seen in Fig. 1. The system functionality is maintained by three main components.

(1) *Information extractor*: This component is used to store the necessary metadata about all movies in the movie database IMDb [9] to the local database of ReMovender by means of information extraction techniques.

(2) *User interface*: With the help of this component, the interaction of all the users including the system administrator with the system is maintained. The administrator can update the movie database and calculate similarities and predictions off-line via the user interface. In addition, by tracking the behaviour of the user throughout the system like rating movies, making comments, etc., the necessary updates are made in the knowledge base of the system.

(3) *Recommender*: This component makes the appropriate recommendations to the user by making use of movie content and user profile (rating) information with the help of some prediction techniques.

4.2. User modelling

The input to the CF system is a matrix of users' ratings on a set of items, where each row represents ratings of a single user and each column represents ratings for a single item [10]. Given a CF recommendation system consisting of M users and N items, there exist an $M \times N$ user-item matrix R (Fig. 2). Each entry, $r_{m,n} = x$, of this matrix represents the rating that user m gives to item n, where $x \in \{1, 2, \ldots, r_{\max}\}$ [2]. This user-item matrix can be decomposed into row vectors as follows:

$R = [u_1, u_2, \ldots, u_{\mathrm{M}}]^T$, $u_{\mathrm{M}} = [r_{m,1}, r_{m,2}, \ldots, r_{m,\mathrm{N}}]^T$ where $m = 1, 2, \ldots, M$. Here, the row vector u_{M} represents the ratings of user m for all of N items. This row vector can be considered as our user modelling technique.

4.3. Features and dimensions

In order to make the content information available, we extract all necessary metadata about the movies from the movie database IMDb [9] by using a Python package called IMDbPY [11]. Each movie is represented by a set of features where each feature belongs to a dimension. In Table 1, the set of these dimensions with their type, domain and distance measures are

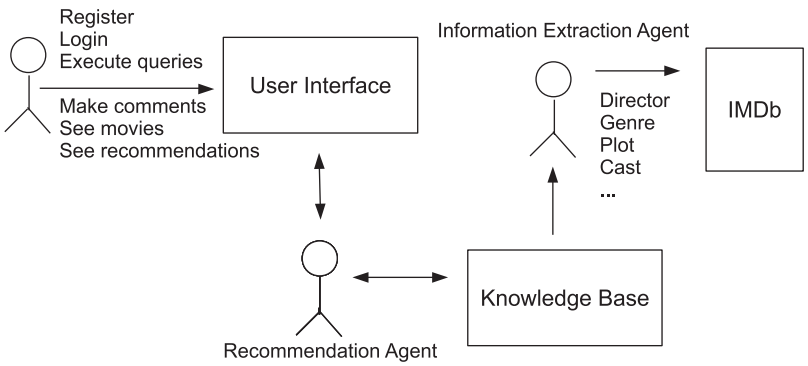


FIGURE 1. Overall design of the system.

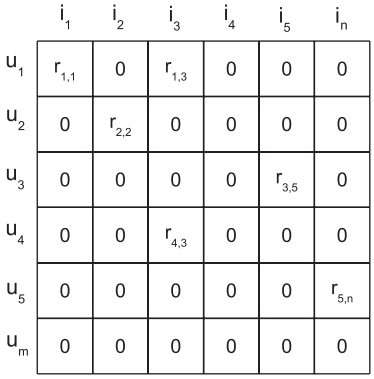


FIGURE 2. An example of user-item matrix.

TABLE 1. Dimensions of movie features.

Dimension	Type	Domain	Distance measure
Rating	Integer	[1,5]	
Production year	Integer	1913, 1986, etc.	
Run time	Integer	30, 60, 90, etc.	
Type	String	Movie, TV, etc.	$T_1 = T_2 ? 1 : 0$
Country	String	France, Italy, etc.	$\frac{\lvert C_1 \cap C_2 \rvert}{\lvert C_1 \rvert}$
Cast	String list	Natalie Portman, Mel Gibson, etc.	$\frac{\lvert C_1 \cap C_2 \rvert}{\lvert C_1 \rvert}$
Genre	String list	Comedy, etc.	$\frac{\lvert G_1 \cap G_2 \rvert}{\lvert G_1 \rvert}$
Language	String list	English, etc.	$\frac{\lvert L_1 \cap L_2 \rvert}{\lvert L_1 \rvert}$
Company	String list	Warner Bros, etc.	$\frac{\lvert C_1 \cap C_2 \rvert}{\lvert C_1 \rvert}$
Writer	String list	Vivian Newton, Kim Watson, etc.	$W_1 = W_2 ? 1 : 0$
Keyword	String list	Murder, love, etc.	$\frac{\lvert K_1 \cap K_2 \rvert}{\lvert K_1 \rvert}$
Plot	String list		

shown. Distance measures are available only for the ones used in the prediction mechanism.

We use two different methods for the distance measure calculation. The first one is valid for features with string type. If the two values are equal, the distance is calculated as 1, else as 0. The second method is applied to the features with string list type and the result is equal to the cardinality of the intersection of the two lists divided by the cardinality of the first list. Although the divisor for the related calculation in [12] is specified as the cardinality of the list with maximum cardinality, we do not use this approach in order to preserve equality. To illustrate, let us suppose that the cast of movie A consists of Kate Winslet, Jack Nicholson and Demi Moore, the cast of movie B consists of Kate Winslet, Demi Moore and Brad Pitt and lastly the cast of movie C consists of Kate Winslet, Demi Moore, Angeline Jolie and Richard Gere. While calculating the similarity of movie A to the other movies in the system, the similarity between A and B should not be calculated as greater than the one between B and C just because the cast of the movie C is more crowded. The similarity calculation mechanism in our approach considers these two similarities as equal since the number of common actors or actresses is the same.

4.4. Prediction and recommendation mechanism

User-based CF predicts the missing data by using the ratings of similar users, whereas item-based CF makes a prediction by using the ratings of similar items. We used a combination of these two approaches to avoid the possibility of ignoring some valuable information that will make the prediction more accurate. The algorithms behind our approach and their interrelationships are given in Fig. 3.

4.4.1. Significance weighting in the PCC

The PCC method was used for user and item similarity calculations. The PCC takes the factor of the differences in user rating styles into account. However, as stated in [13], the PCC overestimates the similarities of users who happen to have rated a few items identically, but may not have similar overall preferences. Thus, we adopt the solution of [1], which proposes a correlation significance weighting factor in order to devalue the similarity weights that are based on a small number of co-rated items.

Accordingly, the new user similarity calculation, where $I_a \cap I_u$ is defined as the number of items rated in common by user a and user u, is defined as follows:

$$\text{CollabSim}(a,u) = \frac{\text{Min}(\lvert I_\alpha \cap I_u \rvert \gamma)}{\gamma} \text{Sim}(a,u). \quad (1)$$

And the updated formula for the item similarity calculation, where $U_i \cap U_j$ is defined as the number of users who rated both

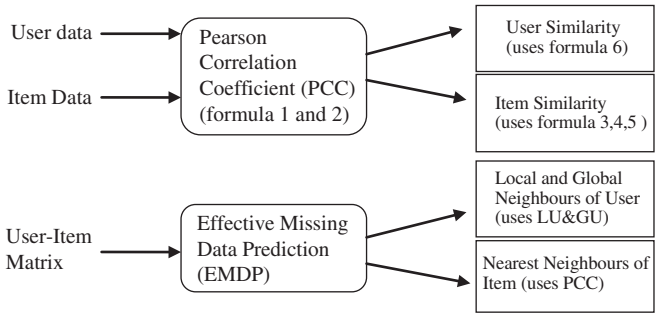


FIGURE 3. Modules of the system.

item i and item j, is as follows:

$$\text{CollabSim}(i, j) = \frac{\text{Min}(|U_i \cap U_j|\delta)}{\delta}\text{Sim}(i, j). \quad (2)$$

$\text{Sim}(a, u)$ represents the similarities of two users, a and u, and is calculated by using a local and global similarity approach as defined in Section 4.4.4. $\text{Sim}(i, j)$ represents the similarities of two items, i and j, and is calculated by using the formulas in Section 4.4.2.

4.4.2. Content-boosted item similarity calculation

Traditional CF approaches do not take the content information of the two items into account while calculating the similarity of these two items. The PCC value of two items, i and j, is given below:

$$\text{CollabSim}(i, j) = \frac{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \text{avg}(r_i))(r_{u,j} - \text{avg}(r_j))}{\sqrt{\sum_{u \in U(i) \cap U(j)} (r_{u,i} - \text{avg}(r_i))^2 \sum_{u \in U(i) \cap U(j)} (r_{u,j} - \text{avg}(r_j))^2}}, \quad (3)$$

where u belongs to the subset of users who rated both of the items, $r_{u,i}$ is the rate user u gave to item i and $\text{avg}(r_i)$ represents the average rating of item i. This algorithm can work without any problem for a very dense user-item matrix. However, there might be very crucial problems while dealing with sparse data. In order for the PCC to be able to calculate the similarity between two items, there must be a subset of users who rated both items, i and j. However, the PCC may not find such a subset, which prevents the algorithm from finding the neighbours of the items. To overcome this difficulty, we also use the content information of the items while calculating the similarity.

It is asserted in [12] that human judgment of similarity between two items often gives different weights to different attributes. For example, while choosing a movie to watch, the genre of a movie can be more important than the writer. Thus, users base their preferences on some latent criteria, which is a weighted linear combination of the differences in individual attributes. Accordingly, [12] defines the similarity between items I_i and I_j as

$$\text{ContentSim}(I_i, I_j) = w_1 f(A_{1i}, A_{1j}) + w_2 f(A_{2i}, A_{2j}) + \cdots + w_n f(A_{ni}, A_{nj}), \quad (4)$$

where w_n is defined as the weight given to the difference between the items in value of attribute A_n, and the distance between the attributes is given by $f(A_{ni}, A_{nj})$. The distance measures defined in Table 1 are used for the attribute distance calculations. These measures provide the f functions to return a value in range [0,1]. As for the feature weights, we exploit the mean values that [12] estimate from a social network graph of items. The list of these weights can be found in Table 2. This estimation is based on the presumption that feature weights are almost universal for different sets of users and movies. To test this presumption, different sets of regression equations have been considered and they have been solved for the weights by Debnath *et al.* [12].

TABLE 2. Feature weight values.

Feature	Mean
Type	0.18
Writer	0.36
Genre	0.04
Keyword	0.03
Cast	0.01
Country	0.07
Language	0.09
Company	0.21

As a conclusion, the formula that we use for the overall item similarity calculation is

$$\text{OverallItemSim}(i, j) = (1 - \beta)\,\text{CollabSim}(i, j) + \beta\,\text{ContentSim}(i, j), \quad (5)$$

where β determines the extent to which the item similarity relies on CF methods or content similarity.

4.4.3. Local neighbour selection

The process of selecting similar users and items has a great impact on the overall prediction mechanism. However, commonly used algorithms generate many dissimilar users. If selected neighbours are not very similar with the current user, the prediction mechanism has a risk of calculating inaccurate values. In order to overcome this problem, we use the thresholds introduced in [1] by modifying the related algorithm: if similarity between the neighbour and the current user is bigger than a similarity threshold η which is a predetermined value, then this neighbour is added to the potential neighbour list which is sorted according to the similarity values. The real neighbours of the user are determined as the minimum of N, the size of the potential neighbour list. Item similarity calculations are made similarly.

4.4.4. Local/global user similarity

Local and global similarity concepts were first introduced in [2] in order to address the sparsity problem. Although [2] makes use of surprisal-based-vector space similarity for local user similarity calculations, we adopted the PCC in our approach.

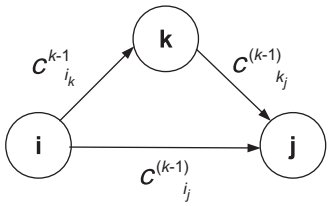


FIGURE 4. The Floyd–Warshall algorithm graph.

The formula for user similarity calculation with the PCC is

$$\mathrm{Sim}(a,u) = \frac{\sum_{i\in I(a)\cap I(u)}(r_{a,i} - \mathrm{avg}(r_a))(r_{u,i} - \mathrm{avg}(r_u))}{\sqrt{\sum_{i\in I(a)\cap I(u)}(r_{a,i} - \mathrm{avg}(r_a))^2}\sqrt{\sum_{i\in I(a)\cap I(u)}(r_{u,i} - \mathrm{avg}(r_u))^2}}, \quad (6)$$

where i belongs to the subset of items rated by both users, a and u; $r_{a,i}$ is the rate user a gave to item i and $\mathrm{avg}(r_a)$ represents the average rating of user a. When there exist no sufficient amount of items that user a and user u both rated, this formula may not return realistic similarity values. But with the help of global user similarity, more neighbours of a user can be found when he/she has few or no immediate neighbours using local user similarity with the PCC. In another sense, global user similarity prevents the underestimation of the similarity of users who have not rated common items. The global similarity between two users is evaluated only if the local similarity between them is below the determined threshold value.

In order to fulfil the previously mentioned goal, we adopt the approach of [2] so that first a user graph is constructed using the users as nodes and the local similarity values as the weight of edges (Fig. 4). The negative local similarity values are set to 0. Then, the maximin distance of two users in the graph is calculated as the global user similarity value between them. We update and exploit the Floyd–Warshall algorithm [14] in order to effectively compute all-pairs of maximin distances. Given an RS consisting of M users, an $M \times M$ user–user matrix R can be constructed so that each entry of this matrix represents the local similarity as the weight of edges. The maximin distance c^k_{ij} between two nodes i and j can be defined with intermediate vertices belonging to the set $\{1, 2, \ldots, k\}$ as follows:

$$c^k_{i.j} = \max^k\{c^{k-1}_{i.j}, \quad \min(c^{k-1}_{i.k}, c^{k-1}_{k.j})\}. \quad (7)$$

4.4.5. Effective missing data prediction

The EMDP algorithm addresses the data sparsity problem by evaluating each missing data with the help of available information. If the evaluation achieves confidence, the predicted rating is stored in the entry of the new matrix. Otherwise, no prediction takes place and the value of the missing data remains zero.

To illustrate our prediction approach by using EMDP, let us assume that user a's rating for item i will be predicted. Then, the main steps for our rating estimation process can be summarized as follows:

(1) The user-item rating matrix is initialized according to the user ratings for the movies. The entries that correspond to the ratings of a movie that has not been rated by the related user are set to 0.

(2) Local and global nearest user neighbours having the similarity above the specified threshold are calculated. If the number of both local and global neighbours is equal to 0, no user-based prediction is made; else, the following formula for the user-based prediction is used. Here $nn_{\mathrm{L}}(u_a)$ denotes the local neighbours of u_a; $nn_{\mathrm{G}}(u_a)$ denotes the global neighbours of u_a; sim'_L, which is calculated with the help of formula (1), denotes the local similarity with significance weighting between u_k and u_a and sim_G represents the global similarity between them.

$$\mathrm{ub\_}r_{a,i} = (1-\alpha)\frac{\sum_{u_k\in nn_{\mathrm{L}}}(u_a)\,\mathrm{sim}'_{\mathrm{L}}(u_k,u_a)\,r_{k,i}}{\sum_{u_k\in nn_{\mathrm{L}}}(u_a)\,\mathrm{sim}'_{\mathrm{L}}(u_k,u_a)} + \alpha\frac{\sum_{u_k\in nn_{\mathrm{G}}}(u_a)\,\mathrm{sim}'_{\mathrm{G}}(u_k,u_a)\,r_{k,i}}{\sum_{u_k\in nn_{\mathrm{G}}}(u_a)\,\mathrm{sim}'_{\mathrm{G}}(u_k,u_a)}. \quad (8)$$

(3) The nearest neighbours of the item are calculated. If the number of neighbours is equal to 0, no item-based prediction is conducted; else, the prediction is made according to the below formula. Here $\mathrm{CollabSim}(i_k, i)$, which is calculated by formula (2), denotes the item similarity between i_k and i. $nn(i)$ denotes the nearest neighbours of item i. Lastly, $\mathrm{avg}(i)$ denotes the average rating of item i.

$$\mathrm{ibr}_{a,i} = \mathrm{avg}(i) + \frac{\sum_{i_k\in nn\,(i)}\mathrm{CollabSim}(i_k,i)(r_{a,i_k} - \mathrm{avg}(i_k))}{\sum_{i_k\in nn\,(i)}\mathrm{CollabSim}(i_k,i)}. \quad (9)$$

(4) This step applies a different procedure according to whether the prediction mechanism is in the training or the testing phase of the evaluation procedure. The users taken into consideration in the training phase are called training users and the ones in the test phase are referred to as active users.

If the number of both user and item neighbours is 0, the predicted value changes according to the below conditions.

(1) If the rating to be predicted belongs to a training user, the return value is 0.

$$\mathrm{ubib\_}r_{a,i} = 0. \quad (10)$$

(2) Else if the number of both user and item neighbours is 0, and the rating to be predicted belongs to an active user, the formula for the predicted value is below, where $\text{avg}(r_a)$ denotes the average rating of user a, and $\text{avg}(r_i)$ denotes the average rating of item i.

$$\text{ubib\_}r_{a,i} = \lambda \text{avg}(r_a) + (1 - \lambda)\,\text{avg}(r_i). \quad (11)$$

Else if the number of user neighbours is 0, the overall predicted value is equal to the result of the item-based prediction.

$$\text{ubib\_}r_{a,i} = \text{ib\_}r_{a,i}. \quad (12)$$

Else if the number of item neighbours is 0, the overall predicted value is equal to the result of the user-based prediction.

$$\text{ubib\_}r_{a,i} = \text{ub\_}r_{a,i}. \quad (13)$$

Else the overall prediction depends both on the result of the user and item-based prediction, and the resulting formula is as follows:

$$\text{ubib\_}r_{a,i} = \lambda\,\text{ub\_}r_{a,i} + (1 - \lambda)\,\text{ib\_}r. \quad (14)$$

During the EMDP prediction for training users, the obtained results are stored in another matrix in order to make this step fair for all entries. Although the predicted ratings can have a value <1 or >5, no rounding procedure is used in this step.

4.4.6. Recommendation

In order to recommend a set of movies to a user, all the missing values in the related row of the user-item matrix are predicted according to the procedure explained above. Then, the movies of which ratings are predicted are sorted in a decreasing rating order so that the recommendations can be presented to the user in that order.

5. REMOVENDER: A CONTENT-BOOSTED CF MOVIE RECOMMENDER

ReMovender is a web-based movie recommendation system where people can freely navigate through, make comments on and give a rating to the movies. Besides, the users of this system are able to search specific movies, and make discussions about movies with the other users. In the meantime, the system tracks the actions of users and tries to learn their movie tastes in order to make some movie recommendations to them. In addition to these features, all users in the system have the chance of finding the similar users who share the same kind of movie preferences.

The users of ReMovender are capable of viewing the details of a specific movie. These details include the information about the movie's genre, language, countries, companies, writers, keywords, run time, cast, plot and ratings on IMDb, which is the current biggest movie database in the world. The related interface for displaying the movie details can be seen in Fig. 5.

The users are also able to rate any movie with the help of this interface. The profile of a user in the system is created automatically from the ratings this user has given to the movies that he/she has watched before. Each time a user gives a rating for a movie, the profile of the related user is updated accordingly so that the predictions in the future can be more successful and satisfactory.

Each time a user logs into the system, some movie recommendations are provided to the user as a list which has been created by various prediction techniques of ReMovender. The predicted ratings of the user for the movies in this list are in the decreasing order, so that the most appealing ones occur at the top. The related screen can be seen in Fig. 6.

The main page of ReMovender can be seen in Fig. 7.

ReMovender also has an interface designed for system administrators to enable them to set some parameters that are used throughout the prediction phase. With the help of this interface, the administrator can also update the movie database, the recommendations for the users in the system and the information about similar users and movies. These processes are completed off-line in order to decrease the response time of the system for providing the user with the necessary recommendations, and similarity information. The related interface is shown in Figs 8 and 9.

6. EVALUATION

The evaluation of web-based recommendation systems is somewhat different from the evaluation of other systems in many aspects. The most fundamental distinction is that the core of this problem is to model and maximize the user satisfaction while making recommendations, which is not only difficult to

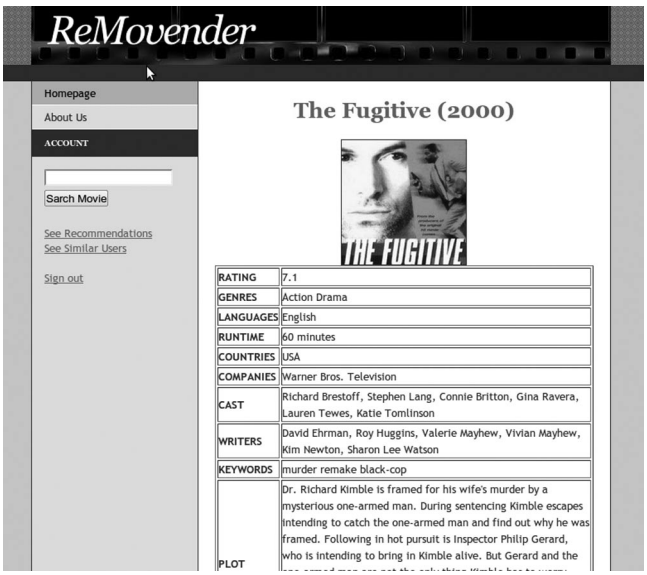


FIGURE 5. Movie page.

FIGURE 6. Recommendation page.

FIGURE 7. Main page.

measure but also changing over time. Below we give the details of the evaluation process applied to ReMovender.

6.1. Data set

The experimental evaluation of ReMovender was conducted using the MovieLens [15] data set maintained by the GroupLens Research group at University of Minnesota. Among the three available data sets, the one containing 100 000 ratings on a scale of 1–5 for 1682 movies by 943 users, where each user has rated at least 20 movies, was preferred in order to make the evaluation

FIGURE 8. Administration page part 1.

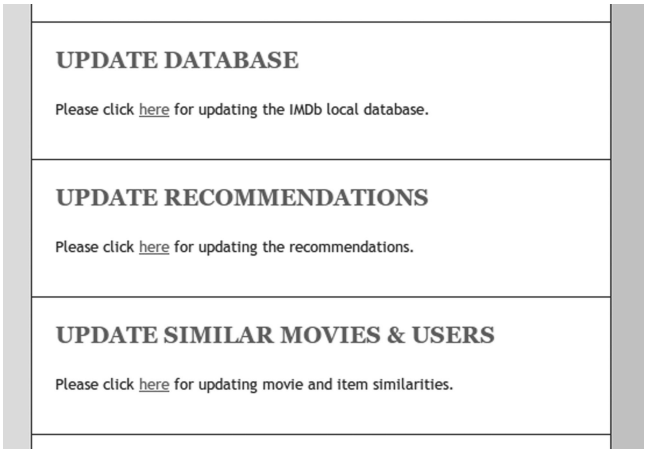


FIGURE 9. Administration page part 2.

results comparable with the results in [1,2] that used the same data set. The density of the user-item matrix created from the MovieLens data set is given by:

$$\frac{100\,000}{943 \times 1682} = 6.30\%,$$

which can be considered sparse enough for the evaluation of the system.

In order to make the contents of the movies in the data set available for the content-boosted CF prediction approach of ReMovender, the title and year of each movie in the MovieLens data set were used for retrieving the related IMDb id from the local copy of the IMDb database. However, ∼400 movies could not be correlated due to the language, year 'and'/'&' and capital letter inconsistencies. Besides, it was observed that the titles of some movies in the MovieLens data set use the a.k.a. (also known as) titles in IMDb. All of these inconsistencies

were corrected manually so that all of the movie ids in the MovieLens data set could be correlated against the IMDb ids, which provided the system to obtain all the content required during the content-boosted prediction.

In order to evaluate the prediction mechanism of ReMovender, a cross-validation method was used. Among various cross validation methods, the holdout method was preferred. Following this method, the data set was separated into two sets, called the training set and the test set. Thus, after a subset of 500 users was extracted randomly from the data set, 300, 200 and 100 of them were selected as the training users and the remaining 200, 300 and 400 were selected as the test users, respectively. The respective sets were named as MovieLens300, MovieLens200 and MovieLens100. The number of ratings provided by the users varied from 5 to 10 and 20, which were named as Given5, Given10 and Given20, respectively. This resulted in a total of nine configurations which represented different item sparsity and user sparsity. The most important reason for adopting this protocol during the experimental set-up of ReMovender is to compare ReMovender with the other systems in the literature that used the same protocol.

6.2. Metrics

Mean absolute error (MAE) metrics were used to measure the prediction quality of the proposed approach and to compare the results with the results of other collaboration filtering methods for which the same metric was used. Although many papers on RSs evaluate their results with methodologies based on root mean square error, we preferred to use the MAE in order to compare our results with the results in the studies our work is based on. However, as stated in [16], different evaluation methodologies lead to totally contrasting conclusions about the quality of recommendations.

MAE is computed by first summing the absolute errors of the N corresponding ratings–prediction pairs and then averaging the sum. And it can be more formally defined as

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |r_i - r_i'|}{N}, \tag{15}$$

where r_i denotes the actual rating that the related user gave to item i, r_i' denotes the rating predicted by our approach and N denotes the number of tested ratings. As can be observed, a larger MAE indicates a lower accuracy.

6.3. Comparison

In order to test the performance of our prediction approach, the MAE values obtained for the nine configurations explained above were compared with the state-of-the-art algorithms on MovieLens database (Table 3).

The parameters or thresholds that were used throughout the prediction process were set as $\lambda = 0.6$, $\gamma = 30$, $\delta = 25$, $\eta = \theta = 0.6$, *numberOfNeighbours* $= 35$ and $\alpha = 0.5$ just like the experimental set-up of [2] to obtain comparable results. Other than that, β was set to 0.5 for evaluating our content-based CF (CBCF) approach. We compared our two separate prediction techniques including the one that uses a pure CF approach without using content information (CFReM), and the other one that exploits content information (CBCFReM) with state-of-the-art algorithms including a user-based approach using the PCC (UPCC) [17], item-based approach using the PCC (IPCC) [18], similarity fusion (SF) [19], EMDP [1], and LU&GU [2] as shown in Table 3. It can be easily observed from this table that either by using content information or not, our prediction approach significantly improves the recommendation quality and outperforms the other competitive algorithms in various configurations. As explained previously, the EMDP algorithm in [1] is a combination of a user-based and item-based predictor, whereas the LU&GU approach in [2] is an improvement of user-based algorithms. When EMDP employed LU&GU to replace traditional user-based approaches, a better performance was achieved. Another conclusion that can be drawn is that using content information in item similarity calculations with our CF approach improves the recommendation accuracy for all configurations.

TABLE 3. MAE comparison with state-of-the-art algorithms on MovieLens.

Training users	Methods	Given5	Given10	Given20
100	CBCFReM	0.7889	0.7653	0.7541
	CFReM	0.7893	0.7665	0.7553
	LU&GU	0.791	0.7681	0.7565
	EMDP	0.7896	0.7668	0.7806
	SF	0.8446	0.7807	0.7717
	UPCC	0.8377	0.8044	0.7943
	IPCC	0.9639	0.8922	0.8577
200	CBCFReM	0.7816	0.7628	0.7533
	CFReM	0.7884	0.7637	0.7588
	LU&GU	0.7937	0.7733	0.7719
	EMDP	0.7997	0.7953	0.7908
	SF	0.8507	0.8012	0.7862
	UPCC	0.8185	0.8067	0.796
	IPCC	0.955	0.9135	0.871
300	CBCFReM	0.7637	0.7562	0.7384
	CFReM	0.7653	0.7616	0.7394
	LU&GU	0.7718	0.7704	0.7444
	EMDP	0.7925	0.7951	0.7552
	SF	0.8062	0.7971	0.7527
	UPCC	0.8055	0.7910	0.7805
	IPCC	0.9862	0.9266	0.8573

6.4. Impact of parameter β

We introduced the parameter β to determine the extent to which the item similarity relies on collaborative similarity or content similarity. With $\beta = 0$, the similarity depends completely on collaborative similarity, whereas it depends completely on content similarity when $\beta = 1$.

We conducted several experiments on all configurations to determine the sensitivity of β, in which the value of β was varied from 0 to 1. The results of these experiments are shown in Figs 10–12, respectively.

During the item-based prediction of the rating for a specific item, the ratings of the other users in the system for that item are not taken into consideration directly. These ratings only have a contribution on the calculation of the average rating of that item. As a design issue, while making user-based predictions, the users who have not rated that item are not considered as similar to the current user, whereas while making item-based predictions, the items who have not been rated by the user are not considered as similar to the current item. Due to the second statement, in order to be capable of using the content information of the items that are similar to the item for which rating will be predicted, the user should have rated these items. Thus, the number of ratings given by a user has importance for our overall prediction mechanism. For these reasons, a decrease in the MAE was observed for all of the configurations when the number of the ratings of the user was increased. The experimental results also show that more accurate and realistic predictions can be obtained when the value of β is around 0.5, because in this way, the prediction can both exploit collaborative and content-based similarity in certain and sensible amounts, which shows that CF and CB approaches both have an important and indispensable role in rating prediction.

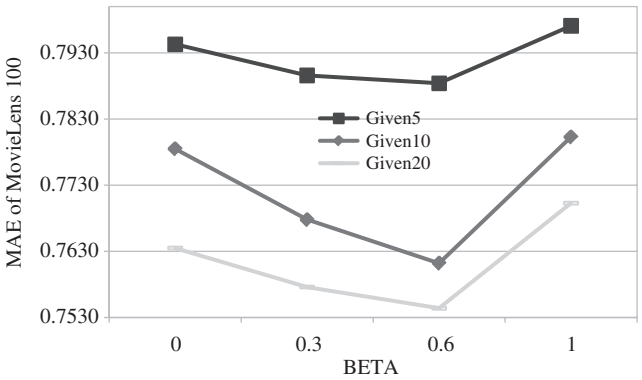


FIGURE 10. Impact of β on MAE (on MovieLens100).

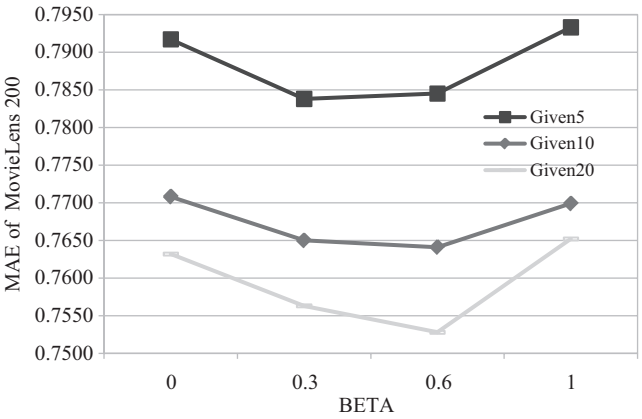


FIGURE 11. Impact of β on MAE (on MovieLens200).

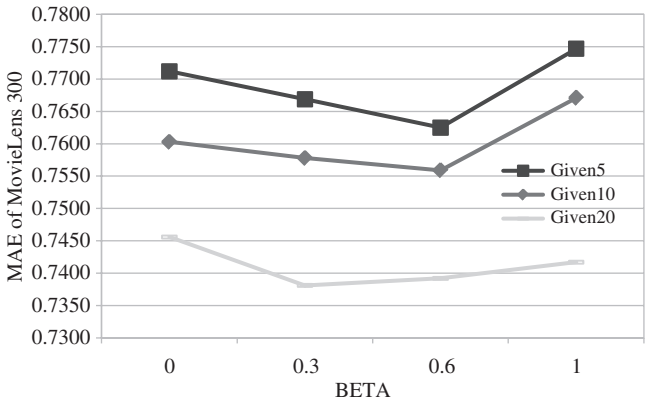


FIGURE 12. Impact of β on MAE (on MovieLens300).

The differences here show that our hybrid approach outperforms the state-of-the-art algorithms in all experiments we conducted. We obtained better solutions than LU&GU and EMDP alone when we used EMDP along with LU&GU. Employing content information in our hybrid approach produced the best results from our experiments. The same parameter values, thresholds and evaluation metrics were used with the state-of-the-art algorithms in order to make the consistent comparisons.

7. CONCLUSION

In this paper, we presented a hybrid approach for RSs which uses a content-boosted CF approach combining the LU&GU and EMDP techniques in order to handle the sparsity problem effectively. We also applied this approach to a movie RS, namely ReMovender, in which the content information of the movies, which are obtained from IMDb, is exploited by the proposed approach during the item similarity calculations.

Empirical analysis shows that our proposed prediction algorithm outperforms other state-of-the-art CF approaches in various configurations. When local and global user (LU&GU) similarity is used by employing the EMDP technique to replace traditional user-based approaches, a better performance is achieved. Moreover, using content information during

the item similarity calculations significantly improves the recommendation quality of the CF approach.

As a future work, we plan to make the system publicly available for daily use as an up-to-date movie recommendation system so that further evaluation about the performance of the prediction mechanism can be possible. Besides, we plan to extend our method by adding content information in a more natural way. Furthermore, different methods using the relationship between user and item information can be developed since the results of this research show that combining these two kinds of information generates better performance. People's preferences change over time. These mind changes can be put into consideration by devising new recommendation algorithms taking time into consideration. Finally, further experiments can be conducted to evaluate the success of the system when similarity calculations are done by using surprisal-based vector similarity introduced by [2] instead of the PCC.

FUNDING

This work is partially supported by The Scientific and Technical Council of Turkey Grant TUBITAK EEEAG-107E234.

REFERENCES

[1] Ma, H., King, I. and Lyu, M.R. (2007) Effective Missing Data Prediction for Collaborative Filtering. *Proc. 30th Annual Int. ACM SIGIR*, Amsterdam, The Netherlands, July 23–27, pp. 39–46. ACM Press, NewYork, NY.

[2] Luo, H., Niu, C., Shen, R. and Ullrich, C. (2008) A collaborative filtering framework based on both local user similarity and global user similarity. Machine Learning **72**(3), 231–245.

[3] Burke, R. (2002) Hybrid recommender systems: survey and experiments. *User Model. User-Adapt. Interact.*, **12**, 331–370.

[4] Adomavicius, G. and Tuzhilin, A. (2005) Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.*, **17**, 734–749.

[5] Adams, J.M., Bennett, P.N. and Tomasic, A. (2007) Combining Personalized Agents to Improve Content Based Recommendations. Technical Report CMU-LTI-07-015.

[6] Yoshii, K., Goto, M., Komatani, K., Ogata, T. and Okuno, H.G. (2006) Hybrid Collaborative and Content-Based Movie Recommendation Using Probabilistic Model with Latent User Preferences. *Proc. 7th Int. Conf. Music Information Retrieval (ISMIR)*, Victoria, Alberta, Canada, October 8–12, pp. 296–301.

[7] Lekakos, G. and Caravelas, P. (2008) A hybrid approach for movie recommendation. *Multimedia Tools Appl.*, **36**, 55–70.

[8] Melville, P., Mooney, R.J. and Nagarajan, R. (2002) Content-Boosted Collaborative Filtering for Improved Recommendation. *Proc. National Conf. Artificial Intelligence*, Edmonton, Canada, July 28–August 1, pp.187–192. American Association for Artificial Intelligence (AAAI) Press, Menlo Park, CA.

[9] The Internet Movie Database (IMDb). http://www.imdb.com.

[10] Kirmemis, O. and Birturk, A. (2008) A Content-Based User Model Generation and Optimization Approach for Movie Recommendation. *Proc. 6th Workshop on Intelligent Techniques for Web Personalization & Recommender Systems,* Chicago, IL, USA, July 13–17. American Association for Artificial Intelligence (AAAI) Press, CA, USA.

[11] IMDbPY. http://imdbpy.sourceforge.net/.

[12] Debnath, S., Ganguly, N. and Mitra, P. (2008) Feature Weighting in Content Based Recommendation System Using Social Network Analysis. *Proc. 17th Int. Conf. World Wide Web (WWW'08)*, Beijing, China, April 21–25, pp. 1041–1042. ACM Press, NewYork, NY.

[13] McLaughlin, M.R. and Herlocker, J.L. (2004) A Collaborative Filtering Algorithm and Evaluation Metric that Accurately Model the User Experience. *27th Annual Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, Sheffield, UK, July 25–29, pp. 329–336. ACM Press, NewYork, NY.

[14] Floyd, R.W. (1962) Algorithm 97: shortest path. *Commun. ACM*, **5**, 345–348.

[15] MovieLens. www.movielens.umn.edu.

[16] Campochiaro, E., Casatta, R., Cremonesi, P. and Turrin, R. (2009) Do Metrics Make Recommender Algorithms? *Proc. Int. Conf. Advanced Information Networking and Applications (AINA)*, Bradford, UK, May 26–29, pp. 648–653. IEEE Computer Society Press, Washington, DC, USA.

[17] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. and Riedl, J. (1994) GroupLens: An Open Architecture for Collaborative Filtering of Netnews. *Proc. ACM Conf. Computer Supported Cooperative Work*, Chapel Hill, NC, USA, October 22–26, pp. 175–186. ACM Press, NewYork, NY.

[18] Sarwar, B., Karypis, G., Konstan, J. and Riedl, J. (2001) Item-Based Collaborative Filtering Recommendation Algorithms. *Proc. 10th Int. Conf. World Wide Web (WWW'01)*, Hong Kong, May 1–5, pp. 285–295. WWW10 Ltd., Kowloon, Hong Kong.

[19] Wang, J., de Vries, A.P. and Reinders, M.J. (2006) Unifying User-Based and Item-Based Collaborative Filtering Approaches by Similarity Fusion. *Proc. 29th Annual Int. ACM SIGIR Conf. Research and Development in Information Retrieval*, Seatttle, WA, USA, August 6–11, pp. 501–508. ACM Press, NewYork, NY.

Ontological Multimedia Information Management System

Hilal TARAKCI[1], Nihan Kesim CICEKLI[2]
[1]*MilSOFT Software Technologies, METU, Technopolis, Ankara, 06531, Turkey*
Email: hilaltarakci@yahoo.com
[2]*Department of Computer Engineering, METU, Ankara, 06531, Turkey*
Tel: +90 312 2105582, Fax: + 90 312 2105544, Email: nihan@ceng.metu.edu.tr

Abstract: In order to manage the content of multimedia data, the content must be annotated. Although any user-defined annotation is acceptable, it is better if many systems use the same annotation format. MPEG-7 is a widely accepted standard for multimedia content annotation. In MPEG-7, semantically identical metadata can be represented in multiple ways due to lack of precise semantics in its XML-based syntax. This unfortunately prevents metadata interoperability. To overcome this, MPEG-7 standard is translated into an ontology. In our work, we use an MPEG-7 ontology on top and wrap the given user-defined ontologies with MPEG-7 ontology, thus building MPEG-7 based ontologies automatically. Our proposed system is an ontological multimedia information management framework due to its modular architecture, ease of integrating with user-defined ontologies naturally and automatic harmonization of MPEG-7 ontology and domain-specific ontologies.

Keywords: semantic querying of video content, multimedia content annotation, mpeg-7, mpeg-7 ontology, mpeg-7 based ontology

1. Introduction

Nowadays, using computer technology is the most common way to socialize. People share their special or common multimedia data on youtube, facebook and similar web sites. Besides, everybody has a personal digital library of photos, videos etc. and has experienced the annoyance of looking for a specific video scene inside a huge amount of data without the help of an intelligent multimedia data management system.

Many projects have been developed for the purpose of managing multimedia data with respect to its content. Among these, we can list AceMedia [1], K-Space[2], BilVideo [3], Informedia [4], VideoQ[5]. In this paper, we are concerned with semantic annotation of multimedia data, especially videos. We will summarize the state of the art and then present the difference of our proposed system. In order to manage huge amount of multimedia data, the content must be annotated. The way in which the content is annotated depends on the annotation environment of the multimedia information management system. Although any user-defined annotation is acceptable, it is better if many systems use the same annotation format. In other words, standardizing the metadata of the content is much better than each system using its own defined annotation format. The widely accepted content annotation standard is Multimedia Content Description Standard known as MPEG-7[6]. MPEG-7 is an ISO/IEC standard and developed by MPEG (Moving Picture Experts Group). Furthermore, MPEG-7 uses XML as the language of choice for the textual representation of content description. In MPEG-7, semantically identical metadata can be represented in many different ways due to lack of precise semantics in XML-based syntax. This unfortunately prevents metadata interoperability. In order to overcome the interoperability issues, efforts

have been spent to translate MPEG-7 standard into an ontology and to enable its integration with other ontologies through appropriate frameworks, thus enhancing interoperability. There exist four OWL/RDF proposals of MPEG-7. These are Jane Hunter's MPEG-7/ABC ontology[7], Tsinaraki's MPEG-7/Tsinaraki ontology[8], Garcia and Celma's Rhizomik model[9] and Arndt's COMM[10]. In this paper, basics of MPEG-7 ontologies are mentioned and our choice of MPEG-7 ontology is presented. In our work, we use an MPEG-7 ontology on top and wrap the given user-defined ontologies with MPEG-7 ontology, thus building MPEG-7 based ontologies automatically. Prior to wrapping the user-defined ontology, we let the user to select concepts that are going to be used in annotation. On the annotation and querying interface, we let the user to annotate or query the wrapped concepts with their attributes. Our proposed system is an ontological multimedia information management framework due to its modular architecture, ease of integrating with user-defined ontologies naturally and automatic harmonization of MPEG-7 ontology and domain-specific ontologies which does not include automated or semi-automated annotation but enables integration of any such module. In the paper, these concepts are explained and some user interface screenshots are presented to give a flavour of the usage of the system. The ontological multimedia information framework can be easily used in specific domains naturally. Moreover, the system can easily be modified according to domain-specific requirements due to its modular architecture.

Since our proposed system is based on MPEG-7 ontology, when a mapping between MPEG-7 and another multimedia content description is available, the system can be easily expanded to welcome new annotations. A domain-specific integration of the framework can be easily produced.

The rest of the paper is organized as follows. A brief summary of related projects is given in Section 2, emphasizing the difference between our framework and the existing work. Section 3 reviews MPEG-7 standard and MPEG-7 based ontologies. Our ontological video model is presented in Section 4. Section 5 summarizes the implementation of the proposed multimedia information management system. Section 6 concludes the paper with some comments about future work.

2. Related Work

People want to search and find the digital content according to its semantics. In order to achieve this, there should be knowledge about the content. This knowledge comes from the metadata of the content. Metadata can be on different levels of abstraction. On the lowest syntactical level there are basic visual features of content like shape, size, texture, color and movement of a camera or an object in a scene. On a higher level these physical features are interpreted to derive semantic information. This includes taxonomies (e.g. genre), organizational information (e.g. scenes for supporting indexing) and basic descriptions (e.g. identification of objects involved in a scene, roles, etc.). Another type of semantic information is the description of the content as annotations in natural language. As the abstraction level of the metadata increases, the management and querying power of the system increases.

There exist many projects that have been developed on the management of multimedia data with respect to its content [1,2,3,4,5].

AceMedia[1] aims to automate annotation process at all levels and ease content creation, search, access, consumption and re-use.

K-Space [2] focuses on narrowing the semantic gap between content descriptors which may be computed automatically and the richness and subjectivity of semantics in high-level human interpretations of audiovisual media.

BilVideo provides an integrated support for queries on spatio-temporal, semantic and low-level features (color, shape, and texture) on video data [3]. BilVideo is an application-

independent system. In other words, the system can easily be tailored for the specific requirements of such applications with the help of the definition of external predicates supported by the system's query language without much effort.

The aim of the Informedia project is to achieve machine understanding of video and film media, in terms of search, retrieval, visualization and summarization[4]. Informedia provides full-content search and retrieval of TV and radio news and documentary broadcasts.

VideoQ[5] is a Web based video search system, where the user queries the system using animated sketches that is defined as a sketch where the user can assign motion to any part of the scene. VideoQ adopts client-server architecture. The client is a java applet which is loaded to a web browser. The user sketches a query scene as a collection of objects with different attributes including motion, spatio-temporal ordering, shape, color and texture.

Another research activity focuses on creating a framework for the automatic annotation of videos in soccer domain and the semantic retrieval of soccer videos based on high-level concepts[11]. The Multimedia Ontologies Annotator is the framework that allows users to import basic ontology schemas, generate the multimedia ontology and annotate videos according to the given ontology. Besides, the system performs complex queries in order to retrieve videos containing specific visual concepts and high-level linguistic concepts.

In our project, we do not focus on automating or semi-automating annotation process itself as most of the above projects do. We focus on providing the user with an ontological video management system framework enabling him to make use of his ontologies in video annotation and querying without any extra knowledge. Amongst the mentioned projects, our proposed system is closest to the Multimedia Ontologies Annotator[11]. Moreover, we aim to import existing MPEG-7 xml files to support backward compatibility to existing systems. Our system is a general framework that delegates the annotation and querying power from the system to the user's hands by allowing him to configure the system with his user-defined ontologies, in opposition to most of the above systems.

3. MPEG-7 Standard and MPEG-7 Ontology

It is preferable that all systems use the same format for annotating the multimedia content. In other words, standardizing the metadata of the content is much better than each system using its own annotation format. To visualize the benefit of multimedia data content standardization, assume you annotated your video in a system that is using the standard format. Then, you can query your annotated video in any system that is using the same standard in annotation, since the systems are talking the same language. In real world, there is such a standard named as MPEG-7[6].

MPEG-7 is an ISO/IEC standard for descriptions of multimedia content. It can be classified into the group of standardised description schemes, however in contrast with other standardised description schemes, it has not been developed in a restricted application domain but it has been intended to be applicable to a wide range of application domains.

The goal of the MPEG-7 standard is to allow interoperable searching, indexing, filtering, and access of audiovisual (AV) content by enabling interoperability among devices and applications that deal with AV content description. MPEG-7 specifies the description of features related to the AV content as well as information related to the management of AV content. The scope of the standard is to define the representation of the description, that is, the syntax and the semantics of the structures used to create MPEG-7 descriptions. The MPEG-7 does this by attaching complex semantics to the content.

In MPEG-7, semantically identical metadata can be represented in multiple ways due to lack of precise semantics in XML-based syntax. This unfortunately prevents metadata

interoperability. To overcome these interoperability issues, efforts have been spent to translate MPEG-7 standard into an ontology and to enable its integration with other ontologies through appropriate frameworks, thus enhancing interoperability. There exist four OWL/RDF proposals of MPEG-7[13]. In the approach proposed by Jane Hunter [7],ABC ontology is used as the core ontology and it provides attachment points for integrating MPEG-7 and domain specific ontologies. Technically, the `mpeg7:MultimediaContent` class is defined as a subclass of the `abc:Manifestation` class, while the corresponding domain ontologies are assumed to be appropriately attached to corresponding ABC classes. Complexity of Hunter's MPEG-7 ontology is OWL-Full.

In Tsinaraki's MPEG-7 ontology[8], the semantic part of MPEG-7 is translated into an ontology that serves as the core ontology for the attachment of domain specific ontologies, in order to achieve MPEG-7 compliant domain specific annotations. Its complexity is OWL-DL.

Garcia and Celma's Rhizomik model[9] is fully automatic translation of the whole standard. Therefore, it is not limited to description schema and has an OWL-DL complexity.

Core Ontology for Multimedia, which is abbreviated as COMM[10] is re-engineering of MPEG-7 using DOLCE design patterns. COMM is an OWL DL ontology.

A brief comparison for existing four MPEG-7 ontologies is given in Table 1 [13].

	Hunter	**DS-MIRF**	**Rhizomik**	**COMM**
Foundations	ABC	None	None	DOLCE
Complexity	OWL-Full	OWL-DL	OWL-DL	OWL-DL
Coverage	MDS+Visual	MDS+CS	All	MDS+Visual
Applications	Digital Libraries	Digital Libraries	Digital Right	MM Analysis

Table 1: Comparison of four MPEG-7 ontologies

In our proposed system, we want to accept existing MPEG-7 annotated multimedia metadata xml files, and convert them to the system supported MPEG7 based ontology. Therefore, we chose to use an MPEG-7 ontology that fully covers the existing MPEG-7 standard, which directs us to use Rhizomik Model.

4. Ontological Video Model

In order to achieve efficient querying and retrieval of multimedia data, the data should be stored in the multimedia database in such a way that queries could be answered in a reasonable time. This can be carried out by using a powerful video modelling technique. In Advanced Video Information System (AVIS) [14], video is divided into frame sequences to which activities, events and objects are associated. These associations are modelled by using special data structures. These data structures are frame segment tree, which is abbreviated as FST, and arrays that contain activities, events and objects. The model has been extended in order to support spatio-temporal query types[16]. In our current system we focus on providing the necessary infrastructure for a general framework which makes use of ontologies in annotation and spatio-temporal querying of videos.

MPEG-7 ontologies try to find an elegant way to integrate MPEG-7 ontology with domain-specific ontologies. Our goal on the other hand, is to automate the integration of MPEG-7 ontology and domain-specific ontologies in an acceptable way. In order to automate the integration of MPEG-7 and user-define ontologies, we propose a video model

ontology specification inspired by AVIS. The resulting video specification is the glue between MPEG-7 ontology and other user-defined ontologies. The specification is composed of the following rules:

1. There is a concept class which will be the superclass of the classes in the user-defined ontology that are to be used in annotation and querying.
2. There is a temporal holder class which is the subclass of the mpeg7:VideoSegmentType in Rhizomik MPEG-7 ontology.
3. There is a hasAppearanceOf property whose domain is video segment class and range is video concept class.
4. There is a isAppearedOn property which is inverse property of hasApperanceOf property.

The third and fourth items of the specification are inspired by AVIS FST tree. In our system, we implemented this specification and use it to stick MPEG-7 ontology and domain-specific ontologies together.

5. Proposed System: Ontological Multimedia Information Management System

There exist two approaches in binding MPEG-7 and domain-specific ontologies. In the first approach, which is Hunter's MPEG-7/ABC approach, there is a core ontology that provides attachment points for MPEG-7 ontology and other domain-specific ontologies. In the second approach, which is MPEG-7/Tsinaraki approach, MPEG-7 is the core ontology and it provides attachment points for other ontologies to bind.

In our work, we first examined Hunter's MPEG-7 ontology, however we had technical problems due to its incompleteness and therefore we head for another MPEG-7 ontology. We decided to use Rhizaomik MPEG-7 ontology because of its one-to-one correspondence with MPEG-7 schema. In addition, we adopted Tsinaraki's approach which uses MPEG-7 as the core ontology. However, we do not leave the users with the complex details of binding their domain-specific ontologies to attachment points provided by MPEG-7 ontology. Instead, we define a standard way of integrating MPEG-7 ontology and other user-defined ontologies and automate the harmonization process.

As stated above, in our system we use Rhizomik MPEG-7 ontology on top and wrap the given user-defined ontologies with MPEG-7 ontology, thus building MPEG-7 based ontologies automatically. Prior to wrapping the user-defined ontology by MPEG-7 ontology, we let the user select concepts that are going to be used in annotation and querying. The concepts that are going to be used in annotation and querying are wrapped by MPEG-7 ontology. The user interface for this process is shown in Figure 1.

The screenshot in Figure 1 is taken from Ontology Management module of our system. In this module, the user enters his ontology to the system by specifying a model name for the imported ontology and by giving the system the path of the file of the imported ontology. When the user clicks IMPORT button, the given model is imported by the system. Afterwards, the system loads the nontrivial concepts in Concept Selection pane, allowing the user to select which concepts in this ontology are subject to be used in annotation and querying. When the user clicks EMBED button, the system integrates the ontology, actually the selected concepts, with the MPEG-7 ontology.

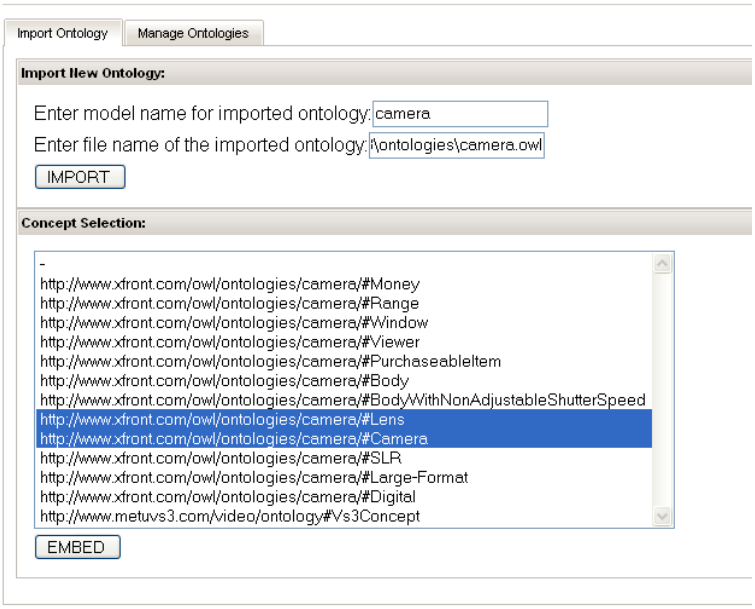


Figure 1: Screenshot for binding MPEG-7 and user-defined ontology

The screenshot in Figure 2 is taken from New Video Metadata page of the system. In this page, the user annotates his videos by using the concepts from his ontology that is entered to the system via Ontology Management page. In this page, the user can specify the interval to annotate, via JMF applet by clicking “Click to see JMF Applet” link in Video Properties pane. Moreover, the user can automatically select region of an object by using JMF applet. Meanwhile, region selection is meaningful for spatio-temporal queries and it is provided by the system infrastructure in order to support spatio-temporal querying. The user selects the model in Model Selection pane to load concepts that are going to be used in annotation. The embedded concepts of the selected model are loaded in the listbox in Concept Selection pane. Then the user selects a concept. Following concept selection, the individuals belonging to that concept are loaded to individual combo-box and attributes for the selected concept are loaded in listbox in Attribute Selection pane. The user can annotate the retrieved attributes in Attribute Selection pane.

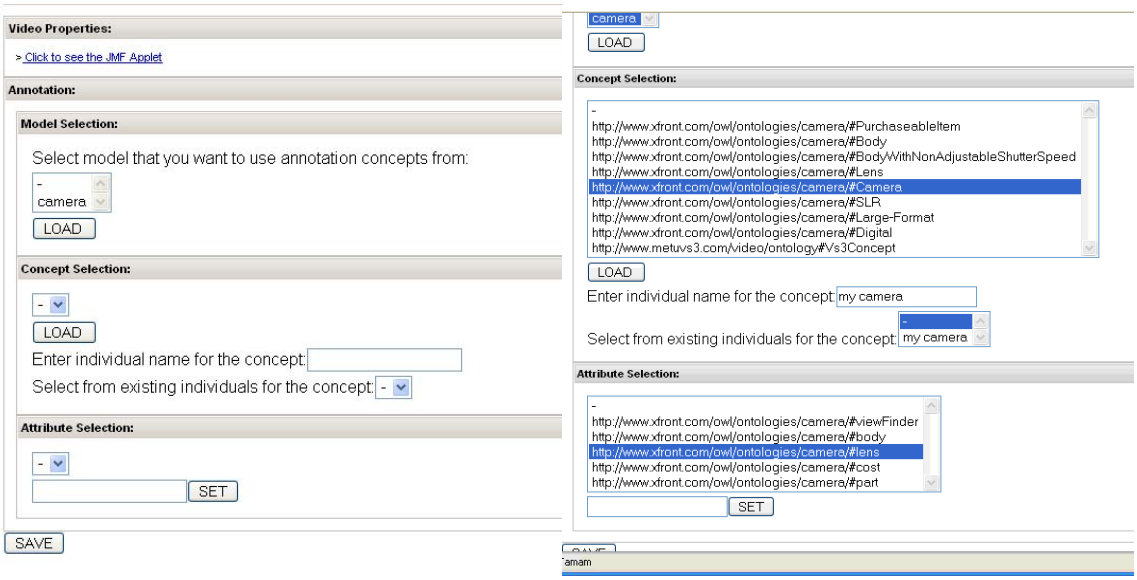


Figure 2: Annotation screenshot: before and after loading concepts

In the implementation, we have used JSF and facelets in front side with myfaces and richfaces components; Java programming language in server side; Jena as ontology API and MySQL as database.

6. Conclusion and Future Work

The ontological multimedia information framework can be easily used in some specific domain naturally. For example, a user who wants to use the system in soccer domain, can achieve this just by feeding the program with a soccer-domain ontology. Moreover, the system can easily be modified according to domain-specific requirements due to its modular architecture. For example, if the user wants to specialize the interface according to a specific domain, it is possible since the implementation is as modular as possible.

In our proposed work, we focused on binding MPEG-7 ontology and domain-specific ontologies in a standard and automated way, the general annotation and querying features, importing existing MPEG-7 xml files into system. We do not consider automating the annotation process itself. As a future work, automatic or semi-automatic annotation feature may be added to the system. The architecture of the system is designed to allow this improvement. Furthermore, a domain-specific multimedia content management system can be put on top of the proposed system.

Our proposed system is an ontological multimedia information management framework with a modular architecture, ease of integrating with user-defined ontologies naturally and automatic harmonization of MPEG-7 ontology and domain-specific ontologies. The system makes use of existing user-defined ontologies in multimedia content annotation. Therefore, the system delegates the querying power of the multimedia management system from the system to the user-defined ontology.

The system opens a door for many future works due to its general structure. One of the aims of this work is to give a starting point to developers who want to develop a multimedia information management system in a specific domain. With this system in their hand, they do not have to worry about MPEG-7 details, ontology API details and interface details

Acknowledgements

This work is partially supported by The Scientific and Technical Council of Turkey Grant ‘‘TUBITAK EEEAG-107E234”.

References

1. S. Bloehdorn, K. Petridis, C. Saathoff, N. Simou, V.Tzouvaras, Y. Avrithis, S. Handschuh, Y. Kompatsiaris, S. Staab and M. G. Strintzis, "Semantic Annotation of Images and Videos for Multimedia", in Proc. of 2nd European Semantic Web Conference, (ESWC '05),Heraklion, Greece, May 29 - June 1,2005.
2. E. Spyrou, G. Koumoulos, Y. Avrithis et al., "K-Space at TRECVid 2006", 4th TRECVID Workshop, Gaithersburg, USA, November 2006.
3. E. Şaykol, Web-based user interface for query specification in a video database system, M.S. thesis, Dept. of Computer Engineering, Bilkent University, Ankara, Turkey, Sept. 2001.
4. M. Christel, T. Kanade, M. Mauldin, R. Reddy, M. Sirbu, S. Stevens, and H. Wactlar, "Informedia Digital Video Library," Communications of the ACM, Vol. 38, No. 4, 1995, pp. 57 - 58.
5. Shih-Fu Chang ; Chen, W. ; Sundaram, H. VideoQ: a fully automated video retrieval system using motion sketches [1998-01-01]
6. http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm
7. J. Hunter, "Adding Multimedia to the Semantic Web - Building an MPEG-7 Ontology", International Semantic Web Working Symposium (SWWS), Stanford, July 30 - August 1, 2001

8. C. Tsinaraki, P. Polydoros and S. Christodoulakis. Interoperability support for Ontology-based Video Retrieval Applications. In Proc. of 3rd International Conference on Image and Video Retrieval (CIVR 2004), Dublin, Ireland, 21-23 July 2004.
9. R. Garcia and O. Celma. Semantic Integration and Retrieval of Multimedia Metadata . In Proc. of the 5th International Workshop on Knowledge Markup and Semantic Annotation (SemAnnot 2005), Galway, Ireland, 7 November 2005.
10. Richard Arndt, Raphaël Troncy, Steffen Staab, Lynda Hardman, Miroslav Vacura: COMM: Designing a Well-Founded Multimedia Ontology for the Web. ISWC/ASWC 2007: 30-43
11. http://www.ercim.org/publication/Ercim_News/enw66/del_bimbo.html
12. http://www.w3.org/2005/Incubator/mmsem/XGR-mpeg7/
13. MPEG-7 based Multimedia Ontologies: Interoperability Support or Interoperability Issue?,Troncy R., Celma O., Little S., García R., Tsinaraki C.,Multimedia Annotation and Retrieval enabled by Shared Ontologies Workshop, MARESO'07.
14. S. Adali, K.S. Candan, S. Chen, K. Erol, V.S. Subrahmanian, The advanced video information system: data structures and query processing, Multimedia Systems 4 (1996) 172–186.
15. Arslan, U., Donderler, M.E., Saykol, E., Ulusoy, Ö., Gudukbay, U., A Semi-Automatic Semantic Annotation Tool for Video Databases, In SOFSEM 2002, Workshop on Multimedia Semantics, Milovy, Czech Republic, November 2002.
16. Koprulu M., Çiçekli, N.K., Yazici, A., Spatio-temporal Querying in Video Databases, Information Sciences 160, 2004, Elsevier Science, pp. 131-152, 2004.,

An Ontology-Based Retrieval System Using Semantic Indexing

Soner Kara [#1], Özgür Alan [#2], Orkunt Sabuncu [#3], Samet Akpınar [*4], Nihan K. Çiçekli [*5], Ferda N. Alpaslan [*6]

[#] *Orbim Corp.*
METU Technopolis, Ankara, Turkey
[1] soner.kara@orbim.com.tr
[2] alan@ceng.metu.edu.tr
[3] orkunt@ceng.metu.edu.tr

[*] *Dept. of Computer Engineering*
METU, Ankara, Turkey
[4] samet@ceng.metu.edu.tr
[5] nihan@ceng.metu.edu.tr
[6] alpaslan@ceng.metu.edu.tr

***Abstract*— In this paper, we present an ontology-based information extraction and retrieval system and its application to soccer domain. In general, we deal with three issues in semantic search, namely, usability, scalability and retrieval performance. We propose a keyword-based semantic retrieval approach. The performance of the system is improved considerably using domain-specific information extraction, inference and rules. Scalability is achieved by adapting a semantic indexing approach. We implement the system using the state-of-the-art technologies in Semantic Web and evaluate the performance against traditional systems. Further detailed evaluation is provided to observe the performance gain due to domain-specific information extraction and inference.**

I. Introduction

The huge increase in the amount and complexity of reachable information in the World Wide Web caused an excessive demand for tools and techniques that can handle data semantically. Current practice in information retrieval mostly relies on keyword-based search over full-text data, which is modeled with bag-of-words. However, such a model misses the actual semantic information in text. To deal with this issue, ontologies are proposed [1] for knowledge representation, which are nowadays the backbone of semantic web applications. Both the information extraction and retrieval processes can benefit from such metadata, which gives semantics to plain text.

Having obtained the semantic knowledge and represented them via ontologies, the next step is querying the semantic data, also known as semantic search. There are several query languages designed for semantic querying. Currently, SPARQL is the state-of-the-art query language for Semantic Web. Unfortunately, these formal query languages are not meant to be used by the end-users. Formulating a query using such languages requires the knowledge of the domain ontology as well as the syntax of the language. Therefore, Semantic Web community works on simplifying the process of query formulating for the end-user. Current studies on semantic query interfaces are carried in four categories, namely, keyword-based, form-based, view-based and natural language-based systems as reviewed in [2]. Out of these, keyword-based query interfaces are the most user-friendly ones and people are already used to use such interfaces thanks to Google.

Combining the usability of keyword-based interfaces with the power of semantic technologies is one of the most challenging areas in semantic searching. According to our vision of Semantic Web, all the efforts towards increasing retrieval performance while preserving user-friendliness will eventually come to the point of improving semantic searching with keyword-based interfaces. This is a challenging task as it requires complex queries to be answered with only a few keywords. Furthermore, it should allow the inferred knowledge to be retrieved easily and provide a ranking mechanism to reflect semantics and ontological importance.

In this paper, we present a complete ontology-based framework for extraction and retrieval of semantic information in limited domains. We applied the framework in soccer domain and observed the improvements over classical keyword-based approaches. The system consists of an automated information extraction module, an ontology populator module, an inference module, and a keyword-based semantic query interface. Our main concern, in this study, is achieving a high retrieval performance while preserving user-friendliness. We show that our system achieves very high precision and recall values even for the very complex queries a user can ask in soccer domain. Furthermore, we evaluate and report the effects of information extraction and inference on query performance.

The rest of the paper is organized as follows: A brief discussion about the related work is given in Section II. In Section III, we give the details of the components of the system, namely IE, ontology population, inference and searching. In Section IV, we report the experiments and their results. Section V concludes the paper with some remarks for future work.

978-1-4244-6523-1/10/$26.00 © 2010 IEEE

II. Related Work

Classical or traditional keyword-based information retrieval approaches are based on the vector space model proposed by Salton et al. [3]. In this model, documents and queries are simply represented as a vector of term weights and retrieval is done according to the cosine similarity between these vectors. [4], [5], [6] and [7] are some of the important studies related to traditional searching. This approach does not require any extraction or annotation phase. Therefore, its easy to implement, however, the precision values are relatively low.

The first step towards semantic retrieval was using WordNet synonym sets (synsets) for word semantics [8], [9]. The main idea was expanding both the indices and queries with the semantics of the words to achieve better recall and precision. If used together with an effective word sense disambiguation (WSD) algorithm, this approach is shown to improve retrieval performance. On the other hand, a poor WSD will cause degradation in performance. Another drawback of this approach is the lack of complex semantics as it is limited with the relations defined in the WordNet.

With the introduction of semantic web technologies, knowledge representation has become more structured and sophisticated, which requires more advanced extraction and retrieval methods to be implemented. The general approach is storing the extracted data in RDF or OWL format, and querying with RDF query languages such as RDQL or SPARQL. Although this approach offers the ultimate precision and recall performance, it is far from useful since it requires a relatively complex query language.

To overcome the difficulties of learning a formal query language, a number of query interface methods are proposed [2]. As we stated earlier, our main focus is keyword-based interfaces. There are several approaches to implement keyword-based querying. To mention a few, SPARK [10] uses a probabilistic query ranking approach for constructing the best query represented by the keywords. Q2Semantic [11] tries to find the best sub-graph expressing the query in the RDF graph. SemSearch [12] uses a template-based approach for query construction. These approaches are not easily scaled to large knowledge-bases as they require traversing RDF graphs or querying the same knowledgebase several times for a single search.

A scalable alternative to query construction from keywords is semantic indexing. In this approach, semantic data in RDF knowledge-bases are indexed in a structured way and made directly available to use with keyword queries. [13], [14] and [15] adapt a similar approach. They index all extracted RDF triples together with the corresponding free text. Since they use very basic extraction methods, such a naive indexing seems feasible. However, complex semantics cannot be captured from the indices containing only subject-predicate-object triples as index terms. If a retrieval system should answer complex queries involving extracted and inferred knowledge, the index must be designed and enriched accordingly.

Our literature survey revealed that current studies on the keyword-based semantic searching are not mature enough: Either they are not scalable to large knowledgebases or they cannot capture all the semantics in the queries. Our aim is to fill this gap by implementing a keyword-based semantic retrieval system using the semantic indexing approach. In other words, we try to implement a system that performs at least as good as traditional approaches and improves the performance and usability of semantic querying. We tested our system in soccer domain to see the effectiveness of semantic searching over traditional approaches and observed a remarkable increase in precision and recall. Moreover we noted that our system can answer complex semantic queries, which is not possible with traditional methods. The study presented in this paper can be extended to other domains as well by modifying the current ontology and the information extraction module as described in [16].

III. Our Approach To Semantic Retrieval

Within the scope of this paper we have developed a fully fledged semantic application which a) contains all the aspects of SW from information extraction to information retrieval and b) uses all the cutting-edge technologies such as OWL-DL, inference, rules, RDF repositories and semantic indexing. The overall diagram of the framework can be seen in Fig. 1.

A. Ontology Design

We designed a central soccer ontology, which is utilized by every aspect of the system, especially in information extraction, inference and retrieval phases. Thus, the overall performance of the system is highly dependent on its quality. We followed an iterative development process in the ontology engineering phase. First, we started with a core ontology including basic concepts and a simple hierarchy. Then, we experimented with this ontology and fix the issues in reasoning and searching. These steps were repeated until we end up with a stable ontology containing 79 classes and 95 properties in soccer domain.

B. Information Extraction (IE)

Information extraction is one of the most important parts of ontology-based semantic web applications. It is the process of adding structured information to the knowledgebase by processing unstructured resources. In this phase, we use the data crawled from the Web sites such as UEFA[1] and SporX[2]. What we obtain as the output of web crawler is some basic information specific to a match (teams, players, goals, stadium, etc.) and natural language texts (narrations of that match). The basic information and the narrations are used as input to our information extraction (IE) module. The details of this module are reported in [16] by Tunaoglu et al. Basically, it is a template-based IE approach for specific domains.We can achieve 100% success rate in UEFA narrations thanks to the language used in the UEFA web-site, which is highly structured and error-free. Fig. 2 gives an idea about the

[1]http://www.uefa.com/
[2]http://www.sporx.com/

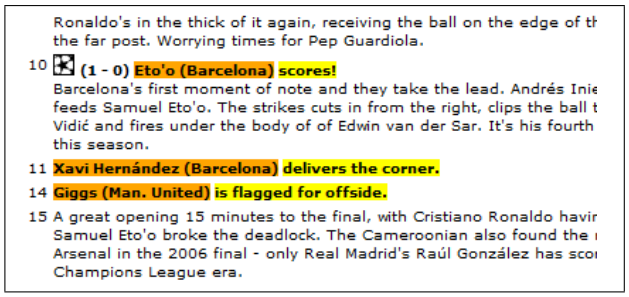


Fig. 2. Example extractions from UEFA narrations

information we can extract from the UEFA web-site. The integration of this module to the system is done in a loosely coupled fashion, so we can use it in semantic applications for any language or domain.

C. Ontology Population

Ontology population is the process of knowledge acquisition by transforming or mapping unstructured, semi-structured and structured data into ontology instances [17]. Our information extractor module [16] already does most of the labor by extracting structured information from unstructured text narrations. For example, from the narration "Keita commits a foul after challenging Belletti" we obtain a foul object, more specifically `FOUL(Keita, Belletti)`. Having the output of the IE module, the ontology population process now becomes creating an OWL individual for each object extracted during IE.

If the IE module cannot extract some attribute of an event, we still create an instance with empty properties. Thus, the recall performance for simple queries will not be affected even IE fails to extract some details of the event. Moreover, if no event is detected in a narration, an instance with the type `UnknownEvent` is created. Unknown events are not discarded because of the reasons described in Section III-E.1. Fig. 3 shows the process of ontology population starting from UEFA narrations ending with OWL instances.

Ontology population is not restricted with the events extracted from the IE module. As mentioned earlier, the crawled information also contains some basic information about the match including players, teams, referees, stadium etc. This information is also added to the ontology by creating an OWL individual for each of them if they do not already exist in the knowledgebase.

D. Inference and Rules

The formal specification of Web Ontology Language, OWL, is highly influenced by Description Logics (DLs)[3]. OWL-DL is designed to be computationally complete and decidable version of OWL, thus it benefits from a wide range of sound, complete and terminating DL reasoners. For our inference module, we use Pellet[4], an open-source DL-reasoner, which supports all the standard inference services such as consistency checking, concept satisfiability, classification and realization.

Consistency checking ensures that there is no contradictory assertion in the ontology. In order to benefit from this feature, we specify some property restrictions during the ontology development. There are two kinds of restrictions in OWL: value constraints and cardinality constraints. We use value constraints, for example, to state that only the goalkeepers (a subset of players) are allowed in the position of goalkeeping and using a cardinality constraint, we can say that only one goalkeeper is allowed in the game. These restrictions not only are useful in consistency checking but also allow new information to be inferred. For example, we could infer the type of an individual if it is the value of a property whose range is restricted to a certain class.

Using classification reasoning we obtain the whole class hierarchy according to class-subclass definitions in the ontology. Inferring new knowledge through classification is a domain independent process and its contribution to the knowledgebase is trivial. In order to infer more interesting information, we use Jena[5] rules.

To illustrate the power of Jena Rules, we give the example of inferring an Assist event. Using the Jena rule shown in Fig. 4, we are able to add a new Assist instance to our knowledgebase. The rule simply looks for two events, namely a Goal and a Pass, that happened in the same match in the same minute and the receiver of the pass is the same person with the scorer. If this is the case, then an Assist instance is created and added to the knowledgebase.

```
noValue(?pass rdf:type pre:Assist)
(?pass rdf:type pre:Pass)
(?pass pre:passingPlayer ?passer)
(?pass pre:passReceiver ?receiver)
(?pass pre:inMatch ?match)
(?pass pre:inMinute ?minute)
(?goal pre:inMatch ?match)
(?goal pre:inMinute ?minute)
(?goal pre:scorerPlayer ?receiver)
makeTemp(?tmp)

-> (?tmp rdf:type pre:Assist)
   (?tmp pre:inMatch ?match)
   (?tmp pre:inMinute ?minute)
   (?tmp pre:passingPlayer ?passer)
   (?tmp pre:passReceiver ?receiver)
```

Fig. 4. An Example for Jena Rules (Assist rule)

E. Semantic Indexing and Retrieval

For the retrieval part, we adapt a semantic indexing approach based on Lucene[6] indices. The idea is extending traditional full-text index with the extracted and inferred knowledge and modifying the ranking so that documents containing ontological information gets higher rates. The details of the index structure and ranking are given in Section III-E.1 and III-E.2 respectively.

[3]http://www.w3.org/TR/owl-guide/
[4]http://clarkparsia.com/pellet
[5]http://jena.sourceforge.net/
[6]http://lucene.apache.org/

TABLE I
INDEX STRUCTURE (SIMPLIFIED FOR BETTER UNDERSTANDING)

Field	Value
docNo	7
event	Foul
match	Chelsea_Barcelona_06_05_2009_20_45
team1	Chelsea
team2	Barcelona
date	2009-05-06
minute	43
subjectPlayer	Michael Ballack
subjectTeam	–
objectPlayer	Sergio Busquets
objectTeam	–
narration	Ballack gives away a free-kick following a challenge on Busquets

1) Index Structure: The structure of semantic index has utmost importance in the retrieval performance. We constructed a Lucene index such that each entry represents a soccer event. As we have mentioned in the previous sections, each event has its own properties associated with it, such as subjects and objects. That information is also included with each event. We also include full-text narrations associated with events to the index. This is especially important if the event type is unknown (an event which is not recognized by the information extractor). Adding full-text narrations to the index tolerates the incomplete event information, thus ensures at least the recall values of traditional full-text search. The index structure can be seen with an example entry in Table I.

2) Searching and Ranking: In traditional keyword search, indexed documents usually contain nothing but raw text associated with that document. Lucene can easily handle such indices and its default ranking gives usually good results. However, complex indices should be handled carefully. In order to take the advantages of our ontology-aided index structure, we slightly modified default querying and ranking mechanism of Lucene. First of all, we boosted the ranking of fields containing extracted and inferred information to stress the importance of them. Secondly, these fields are re-ranked according to their importance. For example, the "event" field is given the highest ranking. This approach prevents misleadings stemming from ambiguous words in full-text. For example, lets say a narration contains "Ronaldo misses a goal": Searching for a "goal" in a traditional search may return this document in the first place, which is a false positive. However, in ontology-aided index, the events whose type is `Goal` will have higher ranks. Since the type of the event above is a `Miss`, it will have a lower rank.

IV. EVALUATION

In order to evaluate the retrieval performance of our system, we have crawled 10 UEFA matches, containing a total of 1182 narrations. Out of these narrations, our IE module was able to extract 902 events. Using these data, we constructed 4 Lucene indices for detailed comparisons. First, we built a traditional full-text index, `TRAD`, using only the narrations of the UEFA matches. This index is used as the baseline for the performance of other methods. Then, we built 2 indices for ontology aided semantic search, namely `BASIC_EXT` and `FULL_EXT`, where the former contains only the basic information available in the UEFA crawl and the latter contains the extracted information in addition to the basic information. Finally, we built an index, `FULL_INF`, which is the expanded version of `FULL_EXT` with the inferred knowledge. All of the indices are evaluated with the queries shown in Table II.

The results can be seen in Table III. First of all, consider the first three queries. There is a considerable difference between `TRAD` and the other methods. The reason is that UEFA narrations use the phrase "P scores!" when the player P scores a goal. Since they omit the word "goal" in narrations, traditional index is not able to retrieve all the goals with the keyword query: "goal". However, the information extraction module can successfully recognize the goal and we can index it as a document with its eventType field filled as "goal". Thus, the improved index can answer both the queries "goal" and "scores" successfully. That is the reason why `BASIC_EXT` and the other indices have very high precision rates.

The improvement provided by the information extraction module can be seen clearly by looking at the difference between `BASIC_EXT` and `FULL_EXT` in 9th and 10th queries. The difference stems from the extracted events such as shoots and goalkeeper saves.

Improvements stemming from the inference can be observed by looking at the queries 4, 7 and 10. In these queries, `FULL_INF` index performs much better than other indices, because it contains additional information due to ontological inference and classification. For example, the 4th query exploits the inferred knowledge about the fact that red cards and yellow cards are also known as punishments. Similarly, the 10th query benefits from the inferred defence players through classification. Finally, the 7th query uses the knowledge obtained from the property hierarchies defined in the ontology. This means, the system can recognize the properties such as `actorOfMissedGoal`, `actorOfOffside`, and `actorOfRedCard` as `actorOfNegativeMove`. Moreover, in the 6th query, we can see the effect of Jena rules. Here, according to one of the rules we defined, we can infer the implicit knowledge of which goal is scored to which goalkeeper, even if that knowledge does not exist explicitly.

However, some queries can slightly suffer from the pollution caused by the information added to the index during the inference. This is mainly due to the fact that some of the fields of `FULL_INF` become very crowded by adding the inferred knowledge, thus slightly deteriorate the rankings. Query-8 illustrates this problem. It is a simple query with a single player name (ronaldo). However, the subjectPlayer field of `FULL_INF` contains some detailed player information in addition to his name. Therefore, the name of the player becomes less significant and the corresponding document is poorly ranked. This problem can be solved by extending the index structure with additional fields rather than accumulating all the information in a single field.

TABLE II
EVALUATION QUERIES

Q-1	Find all goals **(query: goal)**
Q-2	Find all goals scored by Barcelona **(query: barcelona goal)**
Q-3	Find all goals scored by Messi at Barcelona **(query: messi barcelona goal)**
Q-4	Find all punishments **(query: punishment)**
Q-5	Find all yellow cards received by Alex **(query: alex yellow card)**
Q-6	Find all goals scored to Casillas **(query: goal scored to casillas)**
Q-7	Find all negative moves of Henry **(query: henry negative moves)**
Q-8	Find all events involving Ronaldo **(query: ronaldo)**
Q-9	Find all saves done by the goalkeeper of Barcelona **(query: save goalkeeper barcelona)**
Q-10	Find all shoots delivered by defence players **(query: shoot defence players)**

TABLE III
EVALUATION RESULTS (MEAN AVERAGE PRECISION)

	TRAD		**BASIC_EXT**		**FULL_EXT**		**FULL_INF**	
Q-1	0.5/35	1.4%	35/35	100%	35/35	100%	35/35	100%
Q-2	0.4/7	5.7%	5.3/7	75.7%	5.3/35	75.7%	5.3/35	75.7%
Q-3	0.7/3	23.3%	3/3	100%	3/3	100%	3/3	100%
Q-4	0/43	0%	0/43	0%	0/43	0%	43/43	100%
Q-5	1.1/2	55%	2/2	100%	2/2	100%	2/2	100%
Q-6	0.1/9	1.1%	5.7/9	63.3%	5.6/9	62.2%	9/9	100%
Q-7	2.2/7	31.4%	1.9/7	27.1%	2.3/7	32.8%	6.3/7	90.0%
Q-8	7.9/11	71.8%	8.6/11	78.1%	8.5/11	77.2%	7.4/11	67.2%
Q-9	5.1/8	63.7%	4.5/8	56.2%	6.3/8	78.7%	7.5/8	93.7%
Q-10	0/83	0%	0/83	0%	21.9/83	26.4%	81.4/83	98.1%

V. CONCLUSIONS

We have presented a novel semantic retrieval framework and its application to soccer domain, which includes all the aspects of Semantic Web, namely, ontology development, information extraction, ontology population, inference, semantic rules, semantic indexing and retrieval. During the evaluation of the system, we observed that domain-specific information extraction greatly boosts the precision and recall values. Moreover, inference and rules further improve the performance and allow complex domain-specific queries to be handled successfully. Having observed the success in soccer domain, we presume that similar performance can be achieved in other domains as well by extending the ontology and IE module to adapt to the new domains. Indexing and retrieval with keyword interface can be easily adapted to any domain. We also plan to further improve semantic retrieval by adding RDF triples to inferred index without deteriorating the ranking and solve the issues mentioned in Section IV.

ACKNOWLEDGMENT

This work is partially supported by The Scientific and Technical Council of Turkey Grant TUBITAK EEEAG-107E234 and by TUBITAK TEYDEB-3080231.

REFERENCES

[1] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing," *Int. J. Hum.-Comput. Stud.*, vol. 43, no. 5-6, pp. 907–928, 1995.

[2] V. Uren, Y. Lei, V. Lopez, H. Liu, E. Motta, and M. Giordanino, "The usability of semantic search tools: A review," *Knowl. Eng. Rev.*, vol. 22, no. 4, pp. 361–377, 2007.

[3] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Commun. ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[4] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of Documentation*, vol. 28, pp. 11–21, 1972.

[5] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

[6] G. Salton, E. A. Fox, and H. Wu, "Extended boolean information retrieval," *Commun. ACM*, vol. 26, no. 11, pp. 1022–1036, 1983.

[7] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," in *Information Processing and Management*, 1988, pp. 513–523.

[8] J. Gonzalo, F. Verdejo, I. Chugur, and J. Cigarrin, "Indexing with wordnet synsets can improve text retrieval," 1998, pp. 38–44.

[9] R. Mihalcea and D. Moldovan, "Semantic indexing using wordnet senses," in *Proceedings of the ACL-2000 workshop on Recent advances in natural language processing and information retrieval*. Morristown, NJ, USA: Association for Computational Linguistics, 2000, pp. 35–45.

[10] Q. Zhou, C. Wang, M. Xiong, H. Wang, and Y. Yu, "Spark: Adapting keyword query to semantic search," in *Proceedings of the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference (ISWC/ASWC2007), Busan, South Korea*, ser. LNCS, vol. 4825. Berlin, Heidelberg: Springer Verlag, November 2007, pp. 687–700.

[11] H. Wang, K. Zhang, Q. Liu, T. Tran, and Y. Yu, "Q2semantic: A lightweight keyword interface to semantic search," in *ESWC*, 2008, pp. 584–598.

[12] Y. Lei, V. S. Uren, and E. Motta, "Semsearch: A search engine for the semantic web," in *EKAW*, 2006, pp. 238–245.

[13] U. Shah, T. Finin, A. Joshi, R. S. Cost, and J. Matfield, "Information retrieval on the semantic web," in *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*. New York, NY, USA: ACM, 2002, pp. 461–468.

[14] J. Davies and R. Weeks, "Quizrdf: Search technology for the semantic web," *Hawaii International Conference on System Sciences*, vol. 4, pp. 40 112+, 2004. [Online]. Available: http://dx.doi.org/10.1109/HICSS.2004.1265293

[15] I. Celino, E. D. Valle, D. Cerizza, and A. Turati, "Squiggle: An experience in model-driven development of real-world semantic search engines." in *ICWE*, ser. Lecture Notes in Computer Science, L. Baresi, P. Fraternali, and G.-J. Houben, Eds., vol. 4607. Springer, 2007, pp. 485–490. [Online]. Available: http://dblp.uni-trier.de/db/conf/icwe/icwe2007.html#CelinoVCT07

[16] D. Tunaoglu, O. Alan, O. Sabuncu, S. Akpinar, N. K. Cicekli, and F. N. Alpaslan, "Event extraction from turkish football web-casting texts using hand-crafted templates," in *In Proc. of Third IEEE Inter. Conf. on Semantic Computing (ICSC) (in press)*, 2009.

[17] (2008) The semantic web wiki. [Online]. Available: http://semanticweb.org/wiki/Category:Topic_ontology_population

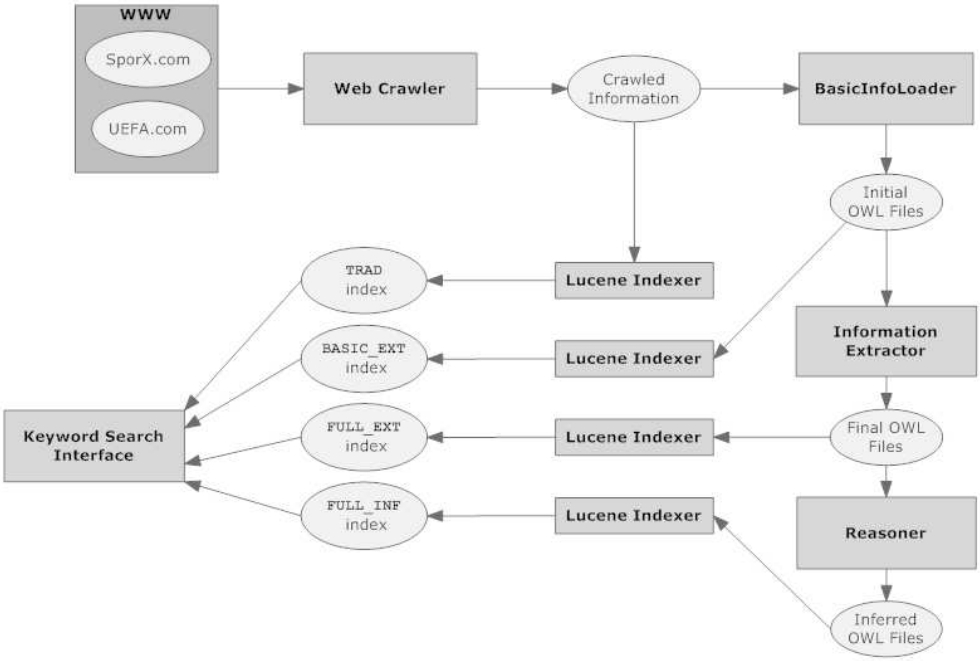


Fig. 1. Overall System Diagram

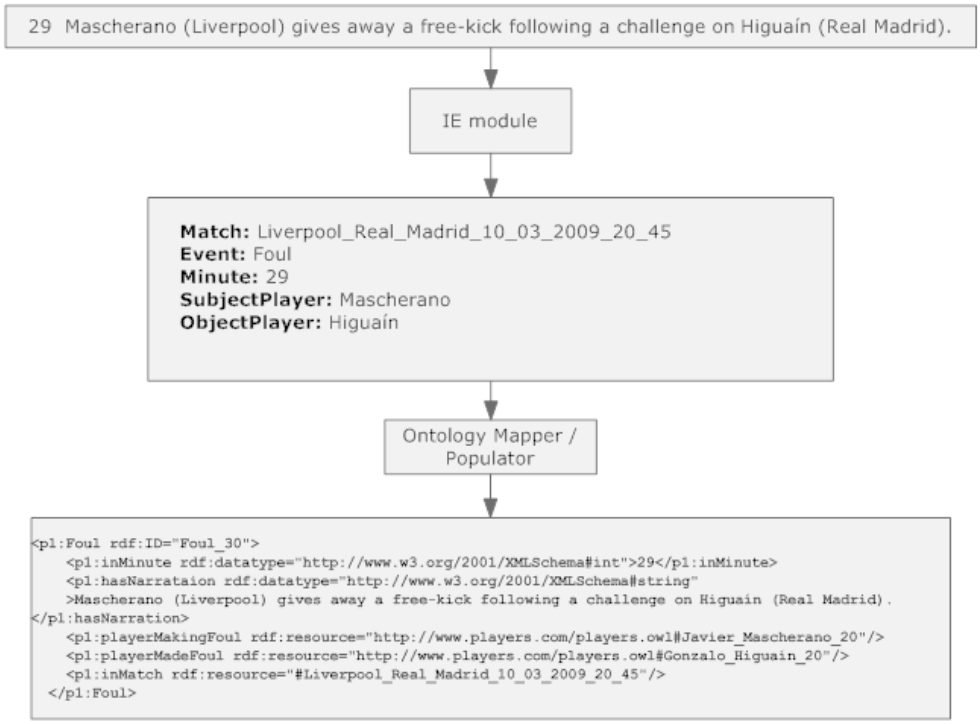


Fig. 3. Information Extraction and Ontology Population

EVENT BOUNDARY DETECTION USING AUDIO-VISUAL FEATURES AND WEB-CASTING TEXTS WITH IMPRECISE TIME INFORMATION

Mujdat Bayar, Özgür Alan, Samet Akpinar, Orkunt Sabuncu, Nihan K. Çiçekli, Ferda N. Alpaslan

Intelligent Systems Lab. Dept. of Computer Engineering METU, Ankara, Turkey
e139472@metu.edu.tr, alan@ceng.metu.edu.tr, samet@ceng.metu.edu.tr
orkunt@ceng.metu.edu.tr, nihan@ceng.metu.edu.tr, alpaslan@ceng.metu.edu.tr

ABSTRACT

We propose a method to detect events and event boundaries in soccer videos by using web-casting texts and audio-visual features. The events and their inaccurate time information given in web-casting texts need to be aligned with the visual content of the video. We overcome this issue by utilizing textual, visual and audio features. Existing methods assume that the time at which the event occurs is given precisely (in seconds). However, most web-casting texts presented by popular organizations such as uefa.com (the official site of Union of European Football Associations) provide the time information in minutes rather than seconds. We propose a robust method which is able to handle uncertainties in the time points of the events. As a result of our experiments, we claim that our method detects event boundaries satisfactorily for uncertain web-casting texts, and that the use of audio-visual features improves the performance of event boundary detection.

Keywords— Event Boundary Detection, Shot Detection and Classification, Multimedia Mining, four, five

1. INTRODUCTION

Creating searchable multimedia archives becomes an important requirement for different domains as a result of the increase in the amount of multimedia content. Especially, the widespread popularity of soccer broadcasts makes automatic annotation essential for querying the semantic content of soccer videos. The annotation provides the means for retrieving specific events in the soccer videos, such as goals, fouls, penalties, bookings etc. Events have duration, therefore they can not be defined by an exact time point; instead, a time interval is required. The problem of event boundary detection is detecting the boundaries of the periods in which events occur. The performance of this detection basically depends on the usage and fusion of different kinds of information sources such as web-casting text and audio-visual features.

A common approach to event boundary detection is to utilize visual features only [1, 2, 3]. Since videos contain a large amount of visual information, it is a very costly process to extract this information. Additionally, it is also hard to detect semantic events and event boundaries in this context. The methods using only visual features are suitable for sports whose videos have simple events and few camera views. For instance, this approach is applied to tennis videos in [1]. [4] presents a system that performs automatic annotation of soccer videos using visual/textual features directly extracted from video. However, these extracted features are insufficient to meet user expectations about richness of semantics. These approaches do not handle the issues such as event semantics like the goal scorer and how it is scored, exact event boundary detection and accurate annotation.

An alternative approach is to fuse the information extracted from textual and visual sources. Web-casting texts are textual resources that describe sports events minute-by-minute. Texts are used to extract high level (semantic) events and their timing in minutes, while visual sources are used to detect the boundaries of these events.

This approach is applied to the soccer domain in [5, 6]. [5] proposes a framework for the alignment of a web-casting text and the related video. First, shot sequences are extracted and then event boundaries are detected using Hidden Markov Models (HMM). The role of the web-casting text is to detect the types and times (minutes) of events. Similarly, [6] applies the same approach to live broadcasts with closed caption texts. In both of these studies, the textual sources include exact time points of events with an accuracy measured in seconds. Thus, the synchronization of text with video becomes easier. However, most textual sources do not include high precision information in terms of time.

We propose a new multi-modal method using web-casting texts and audio-visual features to detect events and event boundaries in soccer videos. The main issue of this method is to align the web-casting text with the visual content, and this is difficult because of inaccurate time information in the web-casting text. We overcome this issue by utilizing not only the textual and visual information, but also audio features. As mentioned above, existing methods assume that the time at which an event occurs is given precisely (in seconds). Therefore, they only focus on detecting time boundaries. However most web-casting texts presented by popular organizations such as uefa.com (the official site of Union of European Football Associations) provide the time information in minutes rather than seconds. We propose a robust method which is able to handle uncertainties in the time points of the events. As a result of our experiments,

978-1-4244-7493-6/10/$26.00

we claim that our method detects event boundaries satisfactorily for uncertain web-casting texts. Additionally, we showed that employing audio features improves the performance of the event boundary detection.

The rest of the paper is organized as follows. Feature extraction is explained in Section 2. Event and Event Boundary Detection is described in Section 3. In Section 4, experimental results are reported, and the paper is concluded with future work in Section 5.

2. FEATURE EXTRACTION

2.1. Video Analysis

In video analysis, the first process is $shot\ detection$ where hard cuts on the video are detected and shot boundaries are labeled. A shot is the smallest meaningful piece of video recording including only one camera view. When the camera view is changed, the existing shot ends and a new shot begins. The second step after shot detection is $shot\ classification$. Since a single shot includes a single camera view, we can classify the shots according to the camera views. In a soccer broadcast, the main camera views are 'far view', 'medium view', and 'close-up view' [5]. Far views are recorded by global cameras from which you can see nearly all the pitch, while close-up views are products of cameras focusing on players. The medium views have characteristics between the far views and close-up views, and these scenes include several players in action. Each camera view is illustrated in Figure 1.

There are many methods proposed for shot detection. All shot detection algorithms use differences between consecutive frames. During a shot, frames do not change rapidly, but the frame structure changes when the shot changes. Main shot detection approaches are Pixel-wise comparison [7, 8], Histogram Comparison [9, 10], Edge Tracking [11], Motion Vector [12] and combinations of these methods [13].

In our method, a color histogram is calculated in RGB color space. In order to obtain more generalized results, R, G, B values are discretized. Each R, G, and B value is down sampled to 3 bits from 8 bits. We then have 2^3x2^3x2^3 = 512 color samples. Each occurrence of a color is counted in a frame by traversing all the pixels. The next step is to calculate the histogram difference.

Let $F_i(r, g, b)$ be the number of pixels having color vector (r, g, b) in the ith frame of N pixels. D denotes our color histogram difference value, which is calculated by the following formula:

$$D = \frac{1}{N} \times \sum_{r=0}^{7} \sum_{g=0}^{7} \sum_{b=0}^{7} |F_i(r, g, b) - F_{i-1}(r, g, b)| \quad (1)$$

In the difference calculation, we choose consecutive frames with k-distance (i.e., every fifth frame where k is 5) in order to avoid false detections resulting from gradual transitions. When the color histogram difference value, D, is above a certain threshold for soccer videos, the current frame is meant to be the first frame of the new shot.

Fig. 1. Far view, Medium view and Close-up view [left to right]

In shot detection, the dominant color and edge pixel count are calculated in addition to the color histogram for each frame. After the shots are detected, the mean of these features is calculated for each shot, in order to be used in shot classification.

We classify the shots according to the camera views. Most of the soccer shot classification methods [14, 15, 11] use similar features and combine them. These features are mainly color distribution, edge pixels, and object segmentation. Since the color green is dominant in soccer videos, the distribution of colors is important in shot classification. For instance, green is everywhere in a far view; and the color of a player's kits covers half of the frame in a close-up view.

We propose a new approach for shot classification. It is not fully color independent, but ignores small field color differences. First, we classify shots as field or non-field. During shot detection, the total histogram, average color histogram difference, and the number of average edge pixels are calculated. If a new shot is detected, the previous shot's total histogram is evaluated. The dominant color in the shot is obtained using this histogram. The dominant color classifies a shot as field or non-field. If the grass color ratio is high, then it is field, otherwise it is non-field.

For field shots, we observed that far view shots have nearly zero color histogram difference values in contrast to medium and close-up views. Medium views and close-up views are distinguished by an edge pixel count with a threshold.

In non-field shots, it is impossible to have a far view shot because the far view camera always displays the entire play, which includes most of the field. Therefore, non-field shots are classified as medium view or close-up view only. We use an edge pixel count to distinguish medium views and close-up views, but we use a different threshold value from the used for field shots. These threshold values for edge pixel count are determined experimentally. Figure 2 depicts the overall method and gives a summary of the shot classification process.

2.2. Audio Analysis

The proposed method for audio analysis aims to determine the major events about which the speaker and the supporters get excited. For this purpose, we extract a sound amplitude value for each second of the soccer video. To calculate this value we use rms (root mean square) of the audio samples in a second and calculate this value for each second.

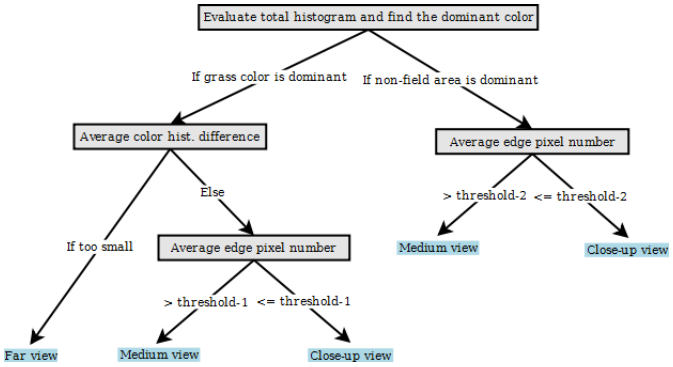


Fig. 2. Decision tree for shot classification

$$rms = \sqrt{\frac{\sum_{i=0}^{n-1} x_i^2}{n}} \quad (2)$$

Here x_i indicates the ith sample in the data-set and n indicates the number of samples in a second. After the shots are detected, an average sound value is calculated for each shot by averaging the samples for each second of the shot. Audio amplitude increases during exciting events such as goals and missed goals, so we use the shots that have raised sound amplitudes to detect exciting events and the boundaries of these events.

2.3. Text Analysis

Textual information describing the high level events in a video is an important source for assisting video semantic analysis. In soccer domain, textual information is mostly extracted from web-casting texts given in the form of match reports. Popular sports web sites (uefa.com, fifa.com, sporx.co-m etc.) and internet media provide adequate amount of soccer match reports. These match reports focus on events of soccer games and give detailed information such as time of the event, type of the event, actors of the event.

Web-casting texts are presented in different languages. We use English and Turkish match reports (sources are from uefa.com, sporx.com) in our work. The methods proposed in [16] are used for extracting the textual information in both languages. In [16] a domain specific information extraction approach is presented. Manually formed templates are used to extract information from unstructured text. This method has three steps. First, available web-casting texts are fetched by a web crawler to an intermediate file. Then, the named entities are tagged such as teams and players in the narrations for each match. Finally, two level lexical analyzer extracts events by analyzing the narrations for each event separately. As a result the type of the event, actors in the event and the time of the event are extracted from the web-casting text. We use these extracted data as input to our method. An example extraction summary can be seen in Figure 3.

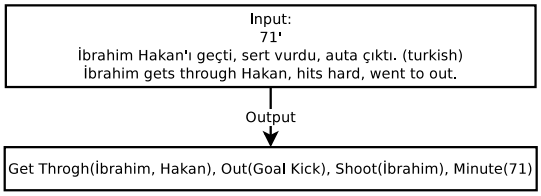


Fig. 3. Information Extraction Summary

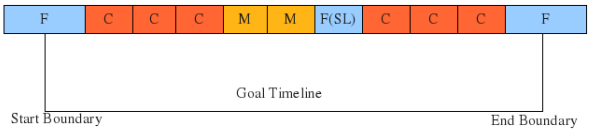


Fig. 4. An example shot sequence of a goal event. F: Far view, F(SL): Short Length Far view, C: Close-up view, M: Medium view

3. EVENT AND EVENT BOUNDARY DETECTION

The extracted data is used for detecting events and event boundaries. The data at hand includes:

1. Shots (start, end time, length).
2. Shot classes (Far view (F), Medium view (M), Close-up view(C)).
3. Sound amplitude averages (for each second of each shot).
4. Types of the events and approximate event time extracted from match reports on the web in minutes.

In soccer broadcasts, the video has a general structure. The match is recorded from a main camera (far view). The camera view is fixed until an event occurs. If an event occurs, the camera view changes and it is switched to medium view, close-up view, or replay mode. Replays are medium view shots or short length far view shots. When the event ends, the camera switches to the main camera view (far view) again. This structure enables us to look for events between two long far view shots. We are able to detect the events and their boundaries according to the shot sequences between two far view shots. Events do not have the same shot sequences, but they have similar camera switches as mentioned before. An example of a shot sequence for a goal event is given in Figure 4.

If we knew the exact time of the event, it would be easier to find the event and its boundaries. However, in our case, we do not have the exact moment of the event and web-casting text gives us inaccurate time. Match reports mark events in minutes, and on the average 2 or 3 events occur per minute. For example, the time points 14:11, 14:30, 14:45 are all labeled as the 15th minute, or sometimes they may even be labelled as the 16th minute if the video is not well synchronized. We have to determine the correct event among the possible events. For this purpose, the following rules are defined:

- Between two far view shots, only a single event can occur.
- Given the approximate time of the event, the search range for the event boundary starts three far view shots before

the event time and ends three far view shots after the event time.

These rules give us 5 possible time intervals to which an event could belong. The next step is to choose one of these time intervals correctly. The shot sequence of the search range looks like the following:

$$F_1....Int_1....F_2....Int_2....F_3....Int_r....F_4....Int_4....F_5....Int_5....F_6$$

Here F_i stands for far view shots. Int_i is the shot sequence (Interval) between far view shots F_i and $F_i + 1$. Int_r indicates the shot sequence (Interval) including the 'reference shot'. The reference shot is the shot which includes the related event time.

The search range can be narrowed down or widened up according to the reliability of the web-casting text. Currently we use five 'far-view to far-view' intervals (shot sequences). The one that looks like the event we are looking for is chosen to determine the event boundaries. After the search range is found, each interval is voted by the rules defined for the event type (there are different rules for different event types). The interval that gets the highest vote is chosen and the event boundary is determined using this interval (shot sequence). Let us give an example to make the process clear. First, the event type and event minute is extracted from the web-casting text. Let us assume that the event is 'a goal event' and the time is 'the 30th minute'. Then we find the shot (reference shot) which corresponds to the 30:00 of the video. After the reference shot is found, five 'far view to far view' intervals are determined by going backwards and forwards through the neighboring shot sequence. Each interval is voted by the rules defined specifically for the goal event and the interval with the highest vote is determined to be the interval containing the goal event. Finally, the winning interval is used to extract the boundaries of the goal event. Boundary extraction from the winning interval is a simple procedure. An event starts in the first far view shot, continues during the interval until the last far view shot. So we assume that the event boundary starts in 20 sec before the end of the first far view shot and ends with the beginning of the last far view shot of the interval.

A voting mechanism is defined for each event type. The features considered include the number of shots in the interval, the length of the interval, the number of close-up shots in the interval, and the total length of medium view shots. We also reward the interval including the reference shot. For exciting events (goals and missed goals), the interval with the highest sound amplitude is also rewarded. These rules differ according to the event type. The rules are determined by observation. Because of the broadcast standards and event specifications, each event has a particular shot count, class and duration on the average. For example, the rule for a goal event says: 1-)The interval having the goal event must contain more than 6 shots. 2-)Total close-up view shot duration must be longer than 15 seconds. 3-)The interval must include minimum 2 medium view or short length far view shots. 4-)The interval can not be shorter than 25 seconds and 5-)It must have high sound amplitude. All these

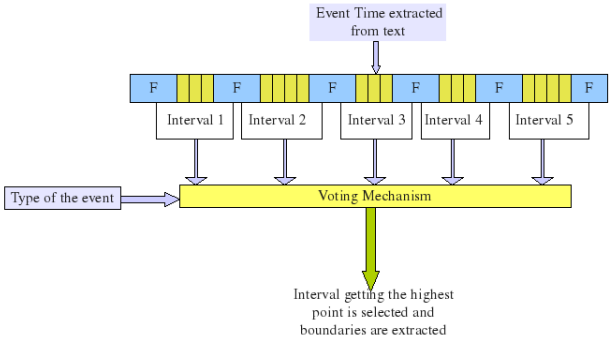


Fig. 5. Event Boundary Detection Mechanism.

items have different weights between 1 and 2 with respect to the importance. The interval getting the highest vote by these items is chosen as the goal event. Another example rule is for corner event: 1-)The interval having the corner event must contain shot count between 2 and 8. 2-)The duration of the interval need to be between 8 and 25 sec. 3-) Close-up views must last shorter than 15 sec. 4-)The corner event must include at least one medium view shot and total medium view shot duration need to be at most 12 sec. The rules for the other events are created in a similar way and some events use the same rules. A summary of the event boundary detection is shown in Figure 5.

4. EXPERIMENTAL RESULTS

We conduct our experiments on five soccer games. Web-casting texts for these games are obtained from "uefa.com" and "sporx.com" which is the most popular website for live match reports in Turkey. The results of the experiments on each step are shown in the following sections.

In the evaluation, 132 shot boundaries were labeled manually. Automatically detected shot boundaries are compared empirically with the ones labeled manually. Table 1 shows the results of automatic shot boundary detection. 123 out of 132 shot boundaries were detected correctly. The 'Missed' column represents the missed boundaries which were supposed to be detected. 9 boundaries were missed because the video had gradual transitions between shots whose color distributions were almost the same. Wrong detections are shown under the column 'False'. They were caused by the motion in close-up views.

Since we don't have the same data set, it would be misleading to compare results with the related works. However, [10] compared different shot detection algorithms. They used histogram differences method and they obtained results with different thresholds. They reached 90% detection rate on the average. With 89% presicion and 93% recall rates, we have satisfactory results compared to them. [5] used a commercial tool for shot detection. Unfortunately we were not able to compare our results.

First halves of 5 games were analyzed automatically by our shot detection tool and they were classified using the technique proposed in this paper. Shots were classified manually too. The

Table 1. Automatic Shot Boundary Detection Results

Total	Correct	Missed	False	Precision	Recall
132	123	9	15	89%	93%

Table 2. Shot Classification Accuracy for Three Types of Shots

Shot Class	Detection Rate
Far view	98%
Medium view	85%
Close-up view	89%

shot classification accuracy is evaluated by comparing the automatically classified shots with the ones classified manually. The results are given in Table 2. Far view shots have a high classification rate of 98%. Medium and close-up view shots have lower classification rates; they are more prone to be confused since it is hard to separate medium and close-up view shots, even by observation.

4.1. Event and Event Boundary Detection

Five soccer games which are different from the ones used for rule production are selected for the evaluation process. The events of the five soccer games are labeled manually to evaluate the proposed event and the event boundary detection method. The number of total labeled events is 98 as shown in Table 3. For each event, the beginning and ending moments of the event are marked. The results for each event are observed and compared with the labeled data. The comparison is done by using the evaluation method proposed as Boundary Detection Accuracy in [5] with the formula:

$$BDA = 1 - \frac{\alpha \left| t_{ms} - t_{ds} \right| + (1 - \alpha) \left| t_{me} - t_{de} \right|}{\max((t_{de} - t_{ds}), (t_{me} - t_{ms}))} \qquad (3)$$

where t_{ds} and t_{de} are automatically detected start and end event boundaries and t_{ms} and t_{me} are manually labeled start and end event boundaries. α is a weight which is set to 0.5.

The inputs to our tool are the approximate event time and the type of the event. The outputs are the starting and ending moments of the given event in seconds. Table 3 shows the results of the evaluation. These results have been achived by using audio feature in exciting events such as goal and missed goal. Although it is not correct to compare our results with [5] since the event time points given us is not precise as given in [5], our experiment results are comparable with them.

We also examined how audio features improve the detection of exciting events. Goals and missed goals are chosen as exciting events. The events are analyzed with and without sound amplitude feature. Table 3 has the results of goals and missed goals with the sound feature. The detection rate of goals and missed goals without the audio feature is given in Table 4. The results show that the audio features increase the detection accuracy of the event boundaries of exciting events.

Table 3. Event Boundary Detection Results

Event	Total	Accuracy
Goal	25	88%
Missed Goal	18	67%
Penalty	2	100%
Corner	40	48%
Red/Yellow Cards	13	62%

Table 4. Event Boundary Detection results for 2 major events without sound amplitude feature

Event	Total	Accuracy
Goal	25	80%
Missed Goal	18	39%

5. CONCLUSION AND FUTURE WORK

In this paper, we present a new multi-modal method for event and event boundary detection by aligning web-casting texts with videos. The event boundary detection technique is based on web-casting texts which are not precise. Web-casting texts describe the events minute-by-minute but these minutes are not the exact time points of the event. As a result, it is necessary to deal with the problem of synchronization in seconds. We aimed to solve this problem and increase the robustness of asynchronous web-casting texts. Similar approaches use extracted data from web-casting texts with the exact event time points and determine boundaries accordingly. When we consider inexact data, it is necessary to look for the events in a wider time period which causes poor event detection rates. The experimental results show that our method has satisfying rates for the event and event boundary detection. It enables us to detect the events and event boundaries that are asynchronous with the web-casting text data. Our experiments also demonstrate that audio-visual features become more important when the text sources are inaccurate. We show that the audio features have remarkable effects on event and event boundary detection in important and exciting events.

Although we have obtained satisfactory results, detailed audio-visual features are necessary to increase the detection rate. As a future project, we plan to add event specific audio-visual features such as time detection, referee detection, and yellow/red card detection. We expect to reach higher detection rates by adding new features on the base method we presented here.

6. ACKNOWLEDGEMENTS

This work is partially supported by The Scientic and Technical Council of Turkey Grant TUBITAK EEEAG-107E234 and by TUBITAK TEYDEB-3080231.

7. REFERENCES

[1] Di Zhong and Shih-Fu Chang, "Real-time view recognition and event detection for sports video," *Journal of Visual Communication and Image Representation*, vol. 15, no. 3, pp. 330–347, 2004.

[2] Ahmet Ekin and A. Murat Tekalp, "Automatic soccer video analysis and summarization," *IEEE Trans. on Image Processing*, vol. 12, pp. 796–807, 2003.

[3] Jian quan Ouyang, Jin tao Li, and Yong dong Zhang, "Replay scene based sports video abstraction," *Lecture Notes In Computer Science*, vol. 3614, pp. 689–697, 2005.

[4] Jrgen Assfalg, Marco Bertini, Carlo Colombo, Alberto Del Bimbo, and Walter Nunziati, "Semantic annotation of soccer videos: Automatic highlights identification," *Computer Vision and Image Understanding*, vol. 92, pp. 285–305, 2003.

[5] Changsheng Xu, Jinjun Wang, Hanqing Lu, and Yifan Zhang, "A novel framework for semantic annotation and personalized retrieval of sports video," *IEEE Transactions on Multimedia*, vol. 10, pp. 421–436, 2008.

[6] Changsheng Xu, Jinjun Wang, Kongwah Wan, Yiqun Li, and Lingyu Duan, "Live sports event detection based on broadcast video and web-casting text," in *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, 2006, pp. 221–230.

[7] Chengcui Zhang, Shu-Ching Chen, and Mei-Ling Shyu, "Pixso: a system for video shot detection," *Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on*, vol. 3, pp. 1320–1324 vol.3, Dec. 2003.

[8] Ming Luo, Daniel DeMenthon, and David Doermann, "Shot boundary detection using pixel-to-neighbour image differences in video," *TRECVID 2004 Workshop Notebook Papers*, 2004.

[9] R. Lienhart, "Comparison of automatic shot boundary detection algorithms," in *Storage and Retrieval for Image and Video Databases*, January 1999, number SPIE 3656, pp. 290–301.

[10] J. S. Boreczky and L. A. Rowe, "Comparison of video shot boundary detection techniques," in *Storage and Retrieval for Still Image and Video Databases IV*, Los Angeles, California, January 1996, number SPIE 2664.

[11] Jinjun Wang, Engsiong Chng, and Changsheng Xu, "Soccer replay detection using scene transition structure analysis," *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, vol. 2, pp. ii/433–ii/436 Vol. 2, March 2005.

[12] Yichuan Hu, Bo Han, Guijin Wang, and Xinggang Lin, "Enhanced shot change detection using motion features for soccer video analysis," *Multimedia and Expo, 2007 IEEE International Conference on*, pp. 1555–1558, July 2007.

[13] A. Jacobs, A. Miene, G. T. Ioannidis, and O. Herzog, "Automatic shot boundary detection combining color, edge, and motion features of adjacent frames," *TRECVID 2004 Workshop Notebook Papers*, pp. 197–207, 2004.

[14] Yi-Hua Zhou, Yuan-Da Cao, Long-Fei Zhang, and Hong-Xin Zhang, "An svm-based soccer video shot classification," *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, vol. 9, pp. 5398–5403 Vol. 9, Aug. 2005.

[15] Xiaofeng Tong, Qingshan Liu, and Hanqing Lu, "Shot classification in broadcast soccer video," *ELCVIA*, vol. 1, pp. 16–25, 2008.

[16] D. Tunaoglu, O. Alan, O. Sabuncu, S. Akpinar, N. Cicekli, and F. Alpaslan, "Event extraction from turkish football web-casting texts using hand-crafted templates," *Proc. of Third IEEE International Conference on Semantic Computing (ICSC '09)*, September 2009.

Event Extraction from Turkish Football Web-casting Texts Using Hand-crafted Templates

Doruk Tunaoğlu
Orbim Corp.
METU Technopolis
Ankara, Turkey
doruk@ceng.metu.edu.tr

Özgür Alan
Orbim Corp.
METU Technopolis
Ankara, Turkey
alan@ceng.metu.edu.tr

Orkunt Sabuncu
Orbim Corp.
METU Technopolis
Ankara, Turkey
orkunt@ceng.metu.edu.tr

Samet Akpınar
Intelligent Systems Lab.
Dept. of Computer Engineering
METU, Ankara, Turkey
samet@ceng.metu.edu.tr

Nihan K. Çiçekli
Intelligent Systems Lab.
Dept. of Computer Engineering
METU, Ankara, Turkey
nihan@ceng.metu.edu.tr

Ferda N. Alpaslan
Intelligent Systems Lab.
Dept. of Computer Engineering
METU, Ankara, Turkey
alpaslan@ceng.metu.edu.tr

Abstract

In this paper, we present a domain specific information extraction approach. We use manually formed templates to extract information from unstructured documents where grammatical and syntactical errors occur frequently. We applied our approach to primarily Turkish unstructured soccer web-casting texts. Compared to automated approaches we achieve high precision-recall rates (97% - 85%). In addition to that, unlike automated approaches we do not use part-of-speech taggers, parsers, phrase chunkers or that kind of a linguistic tool. As a result, our approach can be applied to any domain or any language without the necessity of successful linguistic tools. The drawback of our approach is the time spent on crafting the templates. We also propose the means to decrease that time.

1 Introduction

The amount of digitally available data grew drastically with the growth of the Internet in the last decade. As a result, manually searching and finding specific information from the Web became a difficult and time consuming process. These facts led researchers to study on Information Extraction (IE) whose goal is to automatically extract structured information hidden in texts. The main aim is to extract structured information by analyzing unstructured texts. This information can be used for many purposes including question answering, story tracking, summarizing, inferring facts, etc.

Sport games, especially football, attract many people which results in the necessity of building a search engine which can deal with complex queries. To achieve that goal, these games must be annotated manually or automatically by watching the video of the game or reading the match web-casting text (WCT) available in the Web. Our work focuses on extracting information from the WCT of football matches in Turkish. Although we study in Turkish football domain, our approach can be applied to other domains and other languages as well.

For IE, one needs to use some form of templates. The methods generating these templates in the literature differ in terms of how they form these templates. Basically, we use manually formed templates for extracting information from unstructured text automatically. In short, our approach gives good results (97% precision and 85% recall although there were a lot of grammatical and syntactical mistakes) but manually forming the templates takes considerable amount of time (approximately 1 day for each event

where there were 31 events defined in our ontology including shoot, goal, substitution, different types of pass, etc.)

Throughout the rest of the paper, we present the other available methods in the literature in Section 2, explain our method in detail in Section 3, show some test results in Section 4, make our conclusions and state the future work in Section 5.

2 Related Work

As stated above, methods in literature can be divided into two groups: Ones that automatically form the templates [4, 6, 10, 11] and the ones that use manually formed patterns. [3, 5, 7, 8, 9, 12, 13] Considering another feature we can make another distinction: domain independent methods [3, 4, 5, 11] and domain specific methods [6, 7, 8, 9, 10, 12, 13].

Domain independent methods [3, 4, 5, 11] apply templates on the parse trees of a sentence to extract information. For instance, [5] uses the manually formed rule "NP {',',} 'such as' NPList" to infer that there is an ISA relation between each entry in NPList and NP. If this rule is applied on the sentence "We have visited cities such as Paris, London and Barcelona", it is inferred that Paris, London, and Barcelona are cities. [4] applies the same approach but it forms its templates automatically by examining the relationsip between two entities in Wordnet and a sentence containing these two entities in Wikipedia. These methods use the redundant information available in the Web for confirming the extracted facts. [11] uses part-of-speech (POS) tagging and chunking and feeds the outputs to a perceptron algorithm. [3] states that it requires a great effort to shift to another domain if hand-crafted patterns are used directly on sentences instead of parse trees. On the other hand, domain independent methods cannot capture semantic relations since they use templates on the parse tree without considering the meaning of the nouns or verbs. Furthermore, they use redundant information available in the Internet, but our goal is to achieve high recall rates on unique data.

[6, 10] are domain specific approaches which automatically form their patterns. [6] computes generalizations based on longest common subsequences using tagged examples. [10] finds patterns by automatically examining sentences from football WCT in German using tools to find grammatical functions, phrase structures and POS tags. Both methods use statistical approaches for ambiguity resolution and pattern generalization. Automatic methods [4, 6, 10, 11] are superior compared to manual ones considering the effort spent on a domain. On the other hand, they suffer from low precision-recall rates: 45% - 25% for [10], 76% - 43% for [11] where [4, 6] did not mention about numerical results.

Our method is a domain dependent method which uses hand-crafted rules to extract information like the approaches [7, 8, 9, 12, 13]. [13] uses its templates directly on the sentences to extract information from sport news in Chinese. They try to identify the event, participants, date and the winner. [8] tries to identify relationships in biomedical text by using templates on the parse tree and some modification rules. [7] first identifies the events in a football WCT in English using keywords and then identifies the actors by using rules like "First player(a named entity) after the keyword which is preceded by 'from' is the subject". [9] examines the same football match from three different sources in three different languages using its patterns. It merges these results using a reasoner to increase precision-recall rates. [12] finds events by using keywords and then considers the actors mentioned in the same sentence as related to the event. This approach is weak because it does not find the roles of the actor.

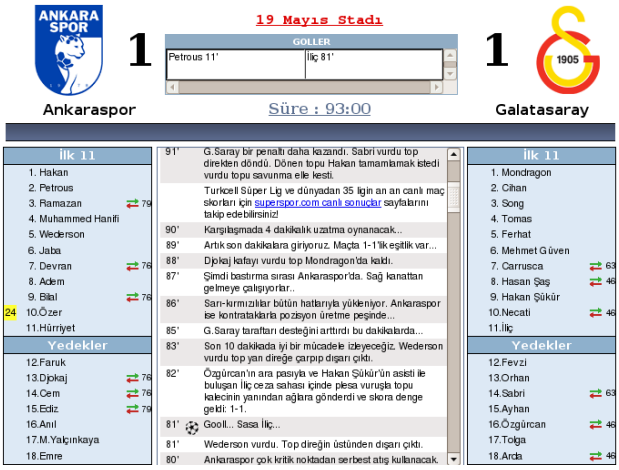


Figure 1. A web-casting text from Sporx

3 Approach

We apply our approach on football WCT in Turkish. Figure 1 shows such an example taken from Sporx [1]. IE is done according to an ontology we have created. This ontology consists of 31 events where each event has its own actors defined by their roles.

Our study consists of three steps: First, the web crawler fetches available WCTs resulting an intermediate file. Then, using the structured part of this intermediate file, named entities are tagged in the narrations for each match. Finally, two level lexical analyzer extracts events by analyzing the narrations for each event separately. Note that each narration consists of a minute and a corresponding unstructed text where grammatical and syntactical errors may occur. Most of our contribution is in the event extraction part which is explained widely in Section 3.3. Figure 2

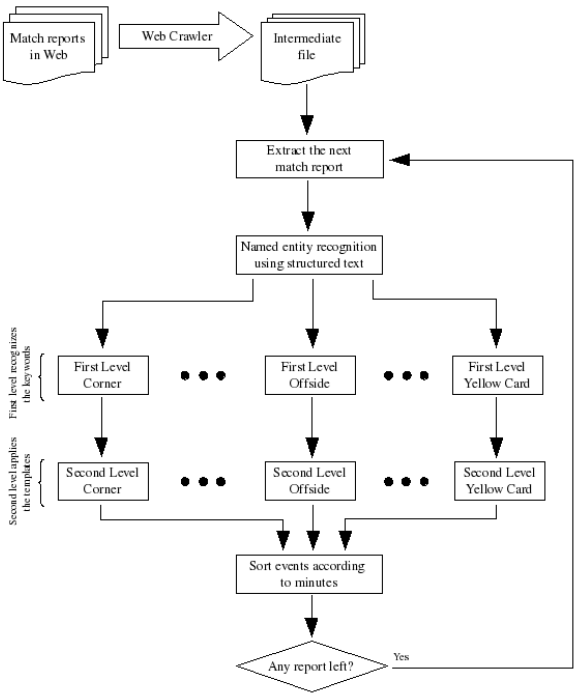


Figure 2. System architecture

shows the overall system architecture.

3.1 Web Crawler

Major sport-news portals across the web have its own standard interface for visualizing the content. Tag names, attribute names and values that are stored in HTML structure are used to download WCT in a structured way. Using an XML file that describes which information resides in which part of the HTML structure, WCT is mapped to a structured file where names of the teams, players, stadium is indicated separately. Since the web sites we used create WCT by automatic code generation, no error has occurred in this step.

3.2 Named Entity Recognition

For most of the WCTs, named entity recognition (NER) can be done easily because the names of the players, teams, stadium etc. are defined in a well-formed structure in the web page or in the match report. As a result, named entities can be extracted using simple regular expressions on the output of web crawler. For our case, in figure 1 you can see the team names in the upper left and right corners and the corresponding players in the columns just below the team names.

After extracting named entities, we tag names considering some rules and cases encountered. Turkish is an agglutinative language which means that most words are formed by joining morphemes together. In addition to that almost all of the affices are suffices. In other words all affices are appended to the end of the word. As a result, we require an empty space only in the beginning but not in the end of a word while tagging it so that we can analyze different forms of the same named entity (accusative, nominative, etc.).

Team names are tagged in the report in the following order considering different cases:

- Full name (Manchester United)
- Each word (Manchester, United)
- Tag for abbreviations (From M.ter to M.chester)

While tagging the names of the players, we consider a more complex approach because of the ambiguities that arose. For each step, names are sorted in descending order according to their length. Using a descending order according to length is important since a name can be a subset of another one (Musa and Musampa):

- Full name (David Beckham)
- Full name abbreviated by the first letter (D.Beckham)
- Each word (David, Beckham)

As a result of this process, there are tags in the report of the form $< team1 >$ instead of (Manchester, M.ter, etc.) and in the form $< team1player7 >$ instead of (David Beckham, D.Beckham, David) etc.

Although NER was an easy task for our study, there were some erroneous cases. These cases mostly occur because of the lack of world knowledge in the system or because of some interesting abbreviations used by the narrator. For instance, when the narrator says “Brezilyalı” (the Brazilian) or “siyah beyazlılar” (“the ones who are black and white”) to refer a player or a team, a human reader can infer which team or who the narrator is talking about. However, our system cannot. We did not perform an explicit analyses for NER but we did not omit errors that occur because of NER in the tests.

3.3 Two Level Lexical Analysis

Most of the IE approaches in English or German use linguistic pre-processing. POS tagging [3, 4, 5, 8, 9, 10, 11], parsing [3, 4, 5, 8, 9, 10, 11] and lemmatization [9, 10] are some of the used methods. We cannot apply their approaches to Turkish since Turkish parsers, POS taggers and other linguistic tools are not as successful as the tools in mostly studied languages like English or German. In order

to cope with this deficiency, we have developed the method of two level lexical analysis: First level recognizes the defined keywords/phrases and discards the rest. Second level extracts information from the output of the first by using the hand-crafted templates. This analysis is done separately for each event which ease the job of manually forming the templates and which allow parallel execution. Furthermore, this approach is fast and can be applied for live analysis. In an average personal computer (2GB ram, dual core 2.4 GHz CPU), an analysis of a match report for 18 events takes an average time of 1.2 seconds. Considering 600ms of this time is used for NER and 600ms for event recognition, it can be guessed that it will take 1.8 seconds when the analysis is done for all of the 31 events. As one can see the system can be used for live annotation if necessary.

Each event has its own keywords which carry information. First level of the analysis keeps these keywords while discarding the rest. For instance "Beckham has awfully fouled in the penalty area by Ballack" gives the output "Beckham FOULED BY Ballack" for first level of event foul. This approach helps to cope with the deficiency of not having a parser by keeping the words that are necessary to recognize the events and the actors.

Second level of the analysis applies the templates to extract information such as actors of events. For instance "PLYR FOULED BY PLYR" can infer foul(Ballack,Beckham) from the above example. Following this approach one can deal with anaphora and cataphora resolution but it is limited to the templates. For instance "PLYR TACKLES PLYR . SHOWING YELLOW" can be used on the sentence "Ballack tackles Beckham. Referee is showing a yellow card" to infer yellowCard(Ballack). Similarly, word sense disambiguation can be done by the templates. However, word sense disambiguation between named entities are not possible by using the templates: If two players share the same name, templates cannot disambiguate because world knowledge is required. For instance, if there are two players with the same name templates cannot decide which one of the players is mentioned but a human reader can infer from the context. Figure 3 shows the parallel structure of the analyses with an example. To clarify the approach better we give a small subset of the rules for both of the levels for event yellow card:

- A small subset of the first level for yellow card
  - kırmızı kart {} ("Red card" is dropped so that if the word "card" remains by itself then it is considered as yellow card)
  - sarı | sarı kart | kart {return "SARI";} (YELLOW | YELLOW CARD | CARD return "YELLOW")
  - sarıyı | sarı kartı | kartı {return "SARIYI";} (Yellow in the accusative form)
  - gördü | görüyor | görürken {return "GORDU"} (Different tenses of the verb see)
  - çıktı | çıkıyor | çıkarken {return "CIKTI"} (Different tenses of the verb to be shown)
  - FUTYLN {return FUTYLN} (Player name with no affices)
  - FUTA {return FUTA} (Player name plus dative case)
- A small subset of the second level for yellow card
  - FUTYLN (SARI| SARIYI) GORDU {return SARI(FUTYLN)} (Player yellow see. Arranged: Player sees yellow which is a phrase used in Turkish)
  - FUTA SARI CIKTI | FUTA SARIYI CIKARDI {return SARI(FUTA)} (Player is shown a yellow card or (referee) shows a yellow card to the player)

Main advantage of this approach is high precision and recall rates. Furthermore, a new event can be easily added to the system since it will not affect the success rates of the other events. And as stated before, this approach does not require any linguistic tools which makes it applicable to many languages where succesful linguistic tools are not developed yet. Main disadvantage is the time spent on forming the templates compared to the automated approaches. It takes 1 day on average for an event for a moderate user for the texts in Sporx [1]. On the other hand, since event extraction is done in a parallel fashion for each event, the job of crafting the templates can be divided in a group of people to decrease the total time spent. Another disadvantage is that the system cannot find events if no such example is seen when crafting the templates. These kind of errors occur when the narrator writes the report in an artistic or funny way which is not very common. For instance, when crafting the templates we encountered the following interesting errors:

- He could not have reached it even if he had taken a taxi. (The bad pass was not caught by the running player)
- He took the revenge in just 2 minutes. (A goal just after 2 minutes of the other team's goal)
- David sent Alex to the moon and back. (David fouled Alex)
- Mehmet conceived the last masterpiece of Alex. (Mehmet saved Alex's shoot)

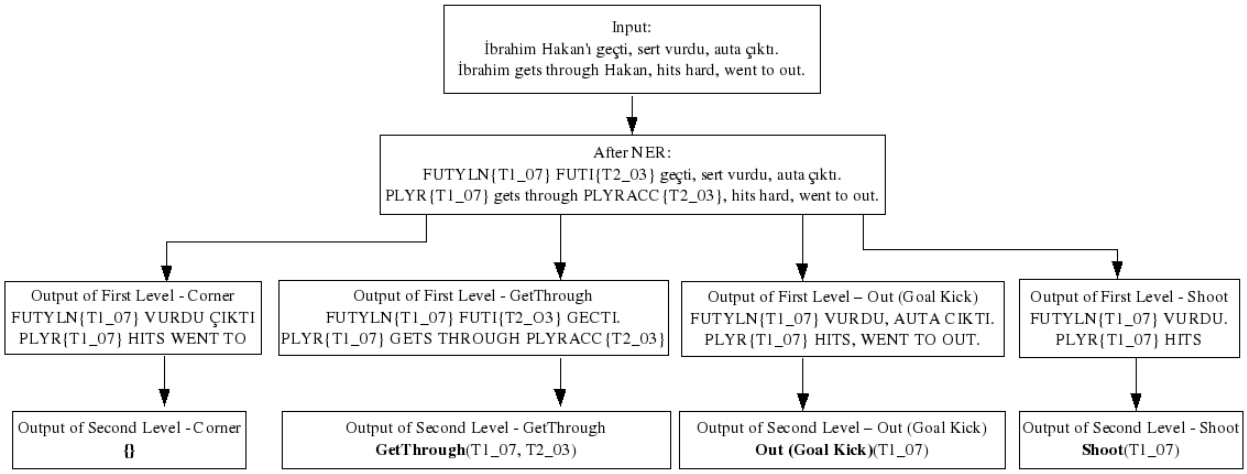


Figure 3. Two level lexical analysis example for different events

4 Test Results

First we tested our system on the WCTs taken from Sporx [1]. We used 70 matches to form the templates for 17 events and used another 20 matches for testing. We did not cover the whole ontology we have defined but we have selected the most important events according to us. Table 1 shows the test results. We calculated precision-recall rates since they are widely used for evaluation in many areas including IE. Precision is a measure which shows how much of the retrieved events are the searched event. Recall is a measure which shows how much of the searched events are retrieved. F score, which accounts precision and recall together, is the weighted harmonic mean of precision and recall. While calculating these scores we used a gold standard where a domain expert read the reports and tagged the events including the actors and their roles. Note that when crafting the templates from the 70 matches we did not use or see the tagged examples of the other 20 matches.

$$Precision = \frac{true\ positive}{true\ positive + false\ positive} \quad (1)$$

$$Recall = \frac{true\ positive}{true\ positive + false\ negative} \quad (2)$$

$$F_\beta = (1 + \beta^2) \times \frac{Precision \times Recall}{\beta^2 \times Precision + Recall} \quad (3)$$

As seen from Table 1 our system achieves a score of 90.4% for F_1 measure with 97% precision and 85% recall when all events are considered together. These rates increase further when we consider the F_1 measure for the primed cases where the event is found but some of the actors are missing. Some of the events (Penalty, Penalty-Kick, Throw-in) show very low success rates because there are not enough examples of them to generalize them. Furthermore, events with same number of examples may have very different success rates (Corner and Foul). This is related with in how many different ways these events are expressed and with the complexity of the phrases that explain the event. For instance, we use 4 templates for substitution and 12 templates for goal where substitiution has a F score 21% higher than goal.

Instead of using 70 matches one at a time to form the templates, we first used 30 of them and tested on the next 10 matches and improved according to the test results. We continued on this cycle until we reach 70 matches. Our aim was to see the evolution of the success rates of different events. Table 2 shows the results of these experiments. This table shows us which events are stabilized considering their F score and which events need more examination for stabilizing the results. By this way, one will not spend time on an event once its success rates are stabilized. Although, one can have a sense from Table 2, we leave defining a mathematical stop criterion as a future work.

To show that our approach can be applied to another languages, we built a system for analyzing the WCTs in English taken from Uefa [2]. Since these texts do not have any grammatical or syntactical error and events are described in a highly structured way we achieved 100% success rate for 13 events just with 4 hours of work using 5 matches for crafting the templates and 5 matches for testing.

Table 1. Results on the events extracted from 20 web-casting texts. tp' show the cases where the event is found but the some of the actors are missing. Precision' and recall' are calculated assuming that tp' = tp where as precision and recall is calculated as assuming that tp' = fn. (tp = true positive, fp = false positive, fn = false negative)

Event	tp	tp'	tp + tp'	fp	fn	precision'	recall'	F_1-score'	precision	recall	F_1-score
Corner	56	7	63	0	0	100	100	100	100	88.9	94.1
Corner-Kick	85	11	96	1	6	99	94.1	96.5	98.8	83.3	90.4
Injury	34	1	35	0	6	100	85.4	92.1	100	82.9	90.7
Foul	49	10	59	1	18	98.3	76.6	86.1	98	63.6	77.2
Free-kick	69	9	78	9	7	89.7	91.8	90.7	88.5	81.2	84.7
Goal	31	12	43	4	4	91.5	91.5	91.5	88.6	66	75.6
Offside	18	4	22	0	9	100	71	83	100	58	73.5
Out (Goal-Kick)	23	15	38	0	1	100	97.4	98.7	100	59	74.2
Out (Throw-in)	5	9	14	1	1	93.3	93.3	93.3	83.3	33.3	47.6
Penalty	1	1	2	0	1	100	66.7	80	100	33.3	50
Penalty-Kick	1	0	1	0	1	100	50	66.7	100	50	66.7
Red Card	2	0	2	0	0	100	100	100	100	100	100
Save	191	1	192	3	16	98.5	92.3	95.3	98.6	91.8	95
Shot	307	0	307	11	19	96.5	94.2	95.3	96.5	94.2	95.3
Substitution	110	0	110	4	4	96.5	96.5	96.5	96.5	96.5	96.5
Throw-in	1	1	2	0	5	100	28.6	44.4	100	14.3	25
Yellow Card	77	10	87	0	2	100	97.8	98.9	100	86.5	92.8
Total	1060	91	1151	34	100	97.1	92	94.5	96.9	84.7	90.4

Table 2. F_1 scores of events with respect to test cycles

	Cycle				
Event	1	2	3	4	5 (Test)
Corner	70.7	88.9	84.5	80	94.1
Corner-Kick	72.7	86	86.5	78.2	90.4
Injury	82.4	96.8	84	97.1	90.7
Foul	70.2	81.2	70.8	78.6	77.2
Free-kick	82.2	85.7	86.2	88.7	84.7
Goal	81.5	92	91.8	84	75.6
Offside	33.3	30.8	57.1	82.4	73.5
Out (Goal-Kick)	72	95.1	94.7	95.7	74.2
Out (Throw-in)	26.7	50	80	80	47.6
Penalty	NA	66.7	66.7	80	50
Penalty-Kick	NA	NA	NA	66.7	66.7
Red Card	NA	50	85.7	75	100
Save	95.1	93.6	97.9	95.4	95.0
Shot	93.6	93.9	95.6	93	95.3
Substitution	94.1	93.1	94.7	95.2	96.5
Throw-in	100	100	NA	75	25
Yellow Card	95.1	86.4	80	97.6	92.8
All Events	85.5	90	90.3	89.9	90.4

5 Conclusions & Future Work

In this paper we have presented an automatic IE method by using hand-crafted patterns. We have shown that high precision-recall rates can be achieved with this approach compared to the methods that automatically generate their patterns. In addition to that our approach can be applied to any language whereas automated ones require some kind of linguistic tool like POS tagger, parser, phrase chunker, etc. High success rates are obtained in an environment where grammatical and syntactical errors occur frequently and this is the realistic case for the Internet. The drawback of this approach is the time spent on a specific domain which increases with the size of the ontology and complexity of the sentences/phrases that describe the events. However, this time can be decreased by crafting the patterns in parallel because the analysis of one event does not affect other analyses.

As a future work, our first aim is to employ a reasoner to increase the success ratios. This reasoner can improve the performance of information extraction by using some logical rules: Completing the actors of events (Rule 4), named entity disambiguation (Rule 5) and inferring new events (Rule 6). Furthermore, as in [9] WCTs of the same match from different sources, if exists, can be used with a reasoner to increase success rates further.

$$Foul(X,Y) \wedge YellowCard() \Rightarrow YellowCard(X) \quad (4)$$

$$Subst.(X,Y|Z) \wedge differentTeam(X,Z) \Rightarrow Subst.(X,Y) \quad (5)$$

$$Pass(X,Y) \wedge Goal(Y) \Rightarrow Assist(X,Y) \quad (6)$$

Another future work consists of dealing with the syntactical errors. Edit distance with a threshold for the defined keywords and named entities can be used to correct some misspelled words. Since each event discards the words that it is not interested, errors only occur if there is a misspelling in the keywords. In addition to that, named entity recognition can be improved but the current simple system works well (2% error in NER).

Finally, we plan to study where to stop when crafting the templates. Number of examples needed to stabilize the success rates differ from event to event according to the complexity of the sentences/phrases that describe the event. Once a stopping criterion is met for an event, further analysis can be done on another event and this will decrease the total time spent on crafting the patterns.

6 Acknowledgments

This work is partially supported by The Scientific and Technical Council of Turkey Grant TUBITAK EEEAG-107E234“.

References

[1] Sporx, http://www.sporx.com.

[2] Uefa, http://www.uefa.com/.

[3] M. Banko and O. Etzioni. The tradeoffs between open and traditional relation extraction. In *Proceedings of ACL-08: HLT*, pages 28–36, Columbus, Ohio, June 2008. Association for Computational Linguistics.

[4] R. M. Casado, E. Alfonseca, and P. Castells. Automatic Extraction of Semantic Relationships for WordNet by Means of Pattern Learning from Wikipedia. In *10th International Conference on Applications of Natural Language to Information Systems, NLDB 2005*, pages 67–79. Springer, 2005.

[5] O. Etzioni, M. Cafarella, D. Downey, S. Kok, A.-M. Popescu, T. Shaked, S. Soderland, D. S. Weld, and A. Yates. Web-scale information extraction in knowitall: (preliminary results). In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 100–110, New York, NY, USA, 2004. ACM.

[6] R. J. Mooney and R. Bunescu. Mining knowledge from text using information extraction. *SIGKDD Explor. Newsl.*, 7(1):3–10, 2005.

[7] N. Nitta, N. Babaguchi, and T. Kitahashi. Extracting actors, actions and events from sports video - a fundamental approach to story tracking. *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, 4:718–721, 2000.

[8] C. Ramakrishnan, K. Kochut, and A. P. Sheth. A framework for schema-driven relationship discovery from unstructured text. In *International Semantic Web Conference*, pages 583–596, 2006.

[9] H. Saggion, H. Cunningham, K. Bontcheva, D. Maynard, O. Hamza, and Y. Wilks. Multimedia indexing through multi-source and multi-language information extraction: The mumis project. *Data Knowl. Eng.*, 48(2):247–264, 2004.

[10] A. Schutz and P. Buitelaar. Relext: A tool for relation extraction from text in ontology extension. In *The Semantic Web (ISWC 2005)*, pages 593–606, 2005.

[11] M. Surdeanu and M. Ciaramita. Robust information extraction with perceptrons. In *Proceedings of the NIST 2007 Automatic Content Extraction Workshop (ACE07)*, March 2007.

[12] C. Xu, J. Wang, K. Wan, Y. Li, and L. Duan. Live sports event detection based on broadcast video and web-casting text. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 221–230, New York, NY, USA, 2006. ACM.

[13] Y. Yang and L. Li. Research on sports game news information extraction. *Natural Language Processing and Knowledge Engineering, 2007. NLP-KE 2007. International Conference on*, pages 96–101, 30 2007-Sept. 1 2007.

A Hybrid Video Recommendation System Using a Graph-Based Algorithm

Gizem Öztürk Nihan Kesim Cicekli

Department of Computer Engineering,
Middle East Technical University, Ankara, Turkey
gizozturk@gmail.com, nihan@ceng.metu.edu.tr

Abstract. This paper proposes the design, development and evaluation of a hybrid video recommendation system. The proposed hybrid video recommendation system is based on a graph algorithm called Adsorption. Adsorption is a collaborative filtering algorithm in which relations between users are used to make recommendations. In this paper, Adsorption algorithm is enriched by content based filtering to provide better suggestions. Thus, collaborative recommendations are empowered considering item similarities. Therefore, the developed hybrid system combines both collaborative and content based approaches to produce more effective suggestions.

Keywords: Recommendation systems, collaborative filtering, content based filtering, information extraction

1 Introduction

Recommendation systems aim to overcome the difficulty of finding proper information. Available systems try to help users to find the most relevant data they want. There are recommendation systems in different domains. For instance Amazon.com recommends books in book domain; Last.fm helps users to find the songs that they want to listen, MovieLens tries to guide users to reach the movies they might like and Netflix which also aims to suggest relative matches to its customers provides various number of movies and TV shows.

Former research work was based on the idea of prediction of ratings only. In other words, the problem seems to guess the rating of unrated items by users. Recent research deals with more complex prediction approaches. Especially, with the improvement of information technologies, recommender systems make use of techniques such as information retrieval, user modeling and machine learning.

Recommender systems can be broadly divided into three categories according to the approach they used to make recommendations. These are content-based recommendation, collaborative recommendation and hybrid recommendation [2]. In content-based recommendation, items are suggested according to their similarity to the items the user selected before. In collaborative recommendation, items are suggested according to the similarity between users with similar habits. Hybrid systems combine these methods to obtain better performance.

Adsorption [3] is a collaborative filtering algorithm which is already applied to YouTube successfully. In YouTube, there are millions of videos available and users can state whether they like the video or not. Adsorption uses this rating information and tries to reach unrated videos using a graph-based algorithm. The newly reached videos are suggested to users as new recommendations.

Adsorption algorithm [3] is among the new generation graph-based collaborative filtering methods. However, this method is not used together with content-based recommendation before. In this paper, the results of Adsorption algorithm are improved by adding content-based filtering to obtain more accurate suggestions. The main contribution of this work [1] is improving the results of Adsorption algorithm by injecting content-based similarities between videos for the purpose of enhancing recommendations. In addition to videos in YouTube, Adsorption algorithm is also applied to movie domain in MovieLens.

The rest of the paper is organized as follows. Section 2 explains the main approach that is used for the development of the hybrid recommendation system. Section 3 describes experiments and evaluation approaches which are used to evaluate the system. Finally, Section 4 concludes the paper and discusses possible future work.

2 A Hybrid Video Recommendation System

The hybrid recommendation system that is developed in this paper is an application which aims to select appropriate videos or movies for users. The developed recommendation system can be used for both YouTube and MovieLens [1]. Recommendations are done according to both collaborative and contend based features. First, ratings are guessed according to collaborative relations. Then, content based (CB) features are injected to provide a hybrid system. The general system architecture is presented in Fig. 1

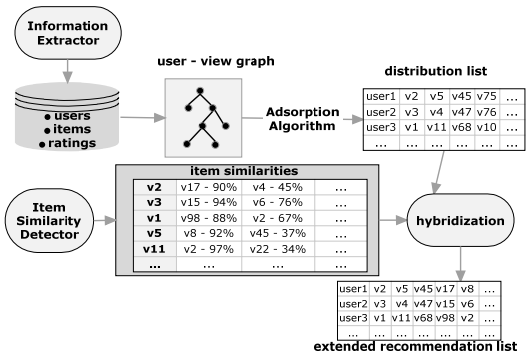


Fig. 1. General System Architecture

The distribution list consists of users and the list of videos for each user. The items in the video list are found to be related to that user and can be recommended accordingly. The item similarities table shows the similarity value of each pair of items.

2.1 Item and User Modeling

In the proposed hybrid recommendation system, first collaborative filtering (CF) is applied; then the content based approach is injected to the results. The input to the CF should be a graph. In this graph, users and items are represented as the nodes. At the beginning items and item ratings are structured together as item-rating pairs. Then, these objects are used in order to model users. Obtained user objects contain user names and a list of item-rating pairs. For each user a graph node is constructed. While examining the list of item-rating objects, a graph node is inserted for each distinct item. Weighted edges are added between nodes considering the ratings that are given to the items by the users. It should be noted that all user names and video IDs are unique in the system.

2.2 YouTube Information Extractor

YouTube does not provide a database that can be used as an evaluation data set. Instead an API [4] is provided in order to help developers to implement client applications. We used this Java API to retrieve the necessary data to construct our data set. The extracted data includes user information, such as user name, list of watched and rated videos, and given ratings. Periodically, the system checks for updates in user information and inserts new data accordingly. This enables the data to stay up-to-date.

The information extractor consists of three main modules which are video, user and rating fetcher. Since the list of YouTube users is not readily available via YouTube API, various videos are visited as a first step to collect user data. There are standard feeds such as top_rated, most_viewed, top_favorites, most_popular, which are provided by YouTube. The returned feeds are in xml format. Gathered feeds are examined and newly obtained video IDs are stored. The obtained xml files contain video information including a feed link for comments of the corresponding video. Comment feed is retrieved because it contains the list of users who share their opinions about the video.

The final step is getting ratings of users. In YouTube, each user has their events feed. If the users agree to share their activities, these feeds can be retrieved from the YouTube API. Activity feeds contain information such as rated videos, favourite videos and commented videos.

2.3 Recommender

The proposed recommender system uses both collaborative filtering and content based approaches in order to provide suggestions. Collaborative filtering forms the predictions for the movies and content based approach aims to improve the obtained results.

Pure Collaborative Filtering Approach. In this paper, a graph-based collaborative filtering algorithm, Adsorption [3], is used. It is a general framework in which there

are both labeled and unlabeled items and it can be used for classification and learning. The basis of the algorithm is giving labels to the unlabeled items using labeled items in the graph structure. The versions of Adsorption algorithm are 'Adsorption via Averaging', 'Adsorption via Random Walks' and 'Adsorption via Linear Systems'. According to the theorem given in [3] all three version of the Adsorption algorithms are equal. In this work 'Adsorption via Averaging' is used due to memory and time issues.

The main idea in 'Adsorption via Averaging' is forwarding labels from the labelled items to the neighbour items, and saving the received labels by neighbours. The important part of the algorithm is to make sure keeping important information while guaranteeing to converge with a reasonable number of label assignments. More formally it can be explained as the following [3].

A graph $G = (V, E, w)$ is given where V is the set of vertices, E denotes the set of edges, and $w : E \to R$ denotes a non-negative weight function on the edges. L denotes a set of labels. Assume each node v in a subset $V_L \subset V$ carries a probability distribution L_v on the label set L. V_L represents the set of labelled nodes.

At this point some pre-processing is necessary. For each vertex $v \in V_L$, a shadow vertex $\tilde{v}$ is created with exactly one outgoing neighbour v, which means $\tilde{v}$ and v are connected by an edge with a weight of 1.

The time complexity of the algorithm is $O(n^2)$ and the pseudo-code of the algorithm is as follows:

Input: $G = (V, E, w), L, V_L$.
repeat
  for each $v \in V \cup \tilde{V}$ do:
    let $L_v = \sum_u w(u, v) L_u$
  end-for
  Normalize L_v to have unit L_1 norm
until convergence
Output: Distributions $\{L_v \mid v \in V\}$

In order to apply the algorithm, the first step is to create the user-view graph. Considering effective usage of memory and processor, videos which have a rating lower than the decided threshold are pruned and not added to the graph. As it is aimed to find the preferences, this threshold value is set to 4. That is ratings greater or equal than 4 mean a certain choice of the user. After the pruning step, a shadow node is created for each user and video, which is the end of the graph construction part.

User-view graph helps reaching related videos using the connections between users and videos. Starting from a user, traverse is done to watched videos firstly and then to other people who watched those videos and so on. Each node of the graph is traversed one by one and its label distribution list is updated according to its neighbours. First, the label distribution list of the current node is cleared. Then, this list is reconstructed by traversing its neighbours and copying their label distribution lists. The edge weight

between the current node and its neighbour is also taken into account in this process. This copying process is continued with the neighbour of the neighbour of the current node and so on. While going deeper, the effect of labels reduces dramatically and time and memory constraints become crucial. For this reason, the system uses only the first 3 levels of the neighbour label distributions. That is after level 3, reached nodes are not very related with the original node so their do not have a concrete benefit.

The size of the label distribution list limits the labels which will be carried to the next iteration. It is accepted that after the upper bound of the label distribution list, remaining items has less importance so they are called poor labels. Therefore, after the label distribution list is formed, it is sorted and poor labels are deleted from the list. This process continues until the label distribution list of all nodes converges. To be more precise, whenever the label distribution list of all nodes remains same on an iteration, the algorithm terminates.

Injection of Content Based Methods to Collaborative Filtering. To increase the strength of recommendations it is decided to add content based filtering to the results obtained by collaborative filtering. The content based method that is used in this paper recommends videos/movies to the users that are similar to the ones obtained as a result of the Adsorption algorithm. The aim is to suggest different but also relevant items to the users. Content based filtering is added by using item similarities. Collaborative results are sorted by relevance and less relevant results are replaced with content based similarity results.

Item Similarities for videos in YouTube. In YouTube API there is a feed which retrieves the related videos to a specific one. When this feed is retrieved the list of related videos are gathered. If a related video is already in the recommendation list, another related item is added to recommendation list.

Item similarities for movies in MovieLens. The similarities between movies can be found according to their features such as year, actors, genre etc. However, in MovieLens database only basic information such as movie name, movie year, movie genre, movie IMDb URL, etc. is provided. So, it is required to gather more detailed movie information from IMDb which stores extra information about movies like movie kind, writer list, cast list, country, language, company and keywords.

In movie domain it is not reasonable to give the same importance to all attributes. To be more precise, writer, genre or country of the movie cannot have the same significance with each other for a movie to be preferred. Therefore, there is a need to decide the importance values of the features. This problem is studied in [6] and feature weighs for movies are determined experimentally. In addition, [6] defines similarity with the equation:

$$S(O_i, O_j) = w_1 f(A_{1i} \mid A_{1j}) + w_2 f(A_{2i} \mid A_{2j}) + ... + w_n f(A_{ni} \mid A_{nj})$$

According to the equation, S describes the similarity between objects O_i and O_j where w_n is the weight applied to the similarity between object attributes A_n. The difference is calculated by the function $f(A_{ni} \mid A_{nj})$. Table 1 shows the feature weight

values as determined in [6]. The related videos of a movie are found by using the values above and IMDb database.

As a result of Adsorption algorithm, a distribution list is obtained which is aimed to be used as the recommendation list itself. Half of bad results are deleted from the distribution list of user. As a result of calculating item similarities, new items are added to the recommendation list of the active user. Therefore, the recommendation list contains items from both collaborative filtering and content based filtering providing a hybrid recommendation to the user.

Table 1. Feature Weight Values

Feature	Mean
Type	0.18
Writer	0.36
Genre	0.04
Keyword	0.03
Cast	0.01
Country	0.07
Language	0.09
Company	0.21

3 Experiments and Evaluation

This section presents the experiments that were carried out in order to evaluate the performance of the system.

3.1 Datasets

Two different datasets are used in order to evaluate the proposed system. These are YouTube data set and MovieLens data set. YouTube dataset is formed by the help of the information extractor. The task of collecting data for our database continued nearly four months. Resulting dataset includes 177733 ratings, 117604 videos and 15090 users. As the values indicate, the YouTube dataset is very sparse. Selected MovieLens dataset has 100,000 ratings for 1682 movies by 943 users [7]. IMDb data is used and movie features are also taken into consideration while finding item similarities for MovieLens.

3.2 Evaluation Metrics

This paper focuses on evaluating the effectiveness of results. In order to evaluate effectiveness, precision and recall are among the most preferred metrics.

Precision is the ratio of the number of relevant items which are retrieved to the total number of retrieved items [9]. Recall is the ratio of the number of relevant items which are retrieved to the total number of relevant items [9]. F-measure is also a metric for evaluation which combines precision and recall. Actually F-measure is the harmonic mean of precision and recall. In this paper precision, recall and F-Measure values are calculated in order to evaluate the system performance. Evaluation is done

for a subset of existing data and using remaining part as training. There are various parameters that may be changed in order to examine results in different perspectives. These are $U, Y, \beta, \gamma, \delta$ and their explanations are given in the following.

Parameters U and Y. Different user groups are formed according to the number of ratings they gave to items. U denotes the user groups for MovieLens users and Y denotes user groups for YouTube users. Because of the high data sparsity of YouTube, only one group of users is formed. This group contains 20 users and average rating of the group is 70. On the other hand, in MovieLens dataset three types of user groups, $U1$, $U2$ and $U3$, are formed according to their average number of ratings. The details for both YouTube and MovieLens user-sets are shown in Table 2.

Table 2. Test User Groups

User Group	Average # of ratings
Y	70
$U1$	250
$U2$	150
$U3$	60

Parameter β. The parameter β denotes the depth value. It represents how deep the Adsorption algorithm goes in the user-view graph. 3 is selected for this parameter because of time constraints.

Parameter γ. It is the size of the label distribution list. Since increasing this parameter also increases the memory usage dramatically, an upper bound value of 40 is selected for its maximum value. On the other hand, there must be a sufficient number of recommendations in order to evaluate the recommendation system properly. So, lower boundary of this parameter is set to 20. Intermediate values are also considered to see the effect of this parameter on overall evaluation. Therefore, calculations are done for five different γ values. These are 20, 25, 30, 35 and 40.

Parameter δ. This is the threshold value of ratings. While traversing the videos that are rated by a user, related video is added as a video node only if its rating is equal to or higher than the value of δ. It is assumed that, users give ratings above 3 (in a 1 to 5 rating system) to videos they like. Because of this, 4 is selected for this parameter.

3.3 Experiments

Results of both pure CF and the hybrid system are presented in this section. As Adsorption is affected very much from sparsity, content based approach gives a chance to increase the quality of suggestions. Effects of this can be seen in results.

YouTube Experiments. In first experiment, the effectiveness of pure CF system is evaluated using YouTube data. For user-set Y calculations are done and the results are presented in Table 3. According to Table 3, while the values for precision are

increasing, recall values are decreasing. As it can be observed, especially recall values are very low. This happens because of data sparsity. It can also be deduced that recall is directly proportional to γ values whereas precision is inversely proportional to γ.

Table 3. YouTube Test Results with pure CF System

User Group Y					
γ-value	**20**	**25**	**30**	**35**	**40**
precision	0.255556	0.220311	0.173333	0.171984	0.171429
recall	0.046589	0.051023	0.057757	0.070678	0.077599
F-Measure	0.07881	0.082857	0.086644	0.100184	0.106837

The second experiment is done in order to see the effect of content-based filtering over the existing CF system. The results are obtained using YouTube dataset. Table 4 demonstrates the results.

Table 4. YouTube Test Results with Hybrid System

User Group Y					
γ -value	**20**	**25**	**30**	**35**	**40**
precision	0.20744	0.184444	0.161333	0.122381	0.101429
recall	0.076589	0.081209	0.089757	0.118986	0.159599
F-Measure	0.111873	0.112767	0.115344	0.12066	0.124032

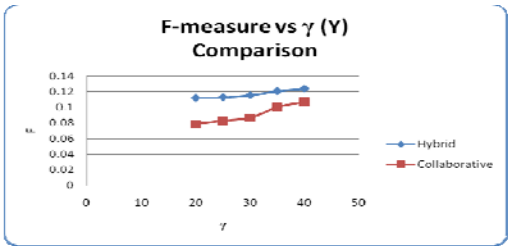


Fig. 2. F-measure vs. γ (Y) Comparison

Fig. 2 demonstrates the relationship between γ and F-measure. It can be concluded that F-Measure tends to increase with increasing γ values. As it is seen from the results, hybrid system curve has a similar form with pure CF curve. Similarly, it can also be seen that hybrid system has higher values which means hybrid system performs better results than pure collaborative system when YouTube data is used.

MovieLens Experiments. This part of the experiment evaluates the effectiveness of system using MovieLens dataset. For each user group 2 tests are done (one for pure CF and one for hybrid). Therefore MovieLens includes 6 different experiments. The results for the first experiment are for pure CF system using MovieLens user-set U1 are shown in Table 5. Secondly, in Table 6 there are results for hybrid system using MovieLens user-set U1.

Table 5. MovieLens Group U1 Test Results with pure CF System

User Group U1					
γ -value	**20**	**25**	**30**	**35**	**40**
precision	0.9373062	0.9327172	0.925128	0.918103	0.908077
recall	0.1147386	0.1198197	0.126901	0.150324	0.167748
F-Measure	0.2044499	0.2123591	0.223187	0.258348	0.283184

Table 6. MovieLens Group U1 Test Results with Hybrid System

User Group U1					
γ -value	**20**	**25**	**30**	**35**	**40**
precision	0.8006507	0.7562651	0.73188	0.730212	0.724491
recall	0.1999656	0.2285986	0.273631	0.309294	0.379793
F-Measure	0.320008	0.3516076	0.398335	0.434534	0.498344

It can be inferred from Fig. 3 that, F-measure increases while the size of the distribution list increases.

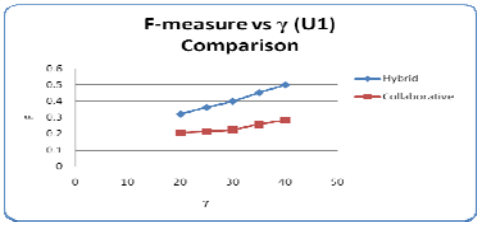


Fig. 3. F-measure vs. γ (U1) Comparison

For all user groups U1, U2 and U3 precision, recall and F-measure values do not change very much. For each user group precision, recall and F-measure graphs follow similar patterns. Therefore, there is not a certain relation considering only user groups, and this shows that Adsorption is insensitive to user groups. As a result, CF system gives coherent results with all user groups.

In all figures, hybrid curves have higher values than pure CF curves. This means that more accurate results are obtained by inserting CB approach in CF approach. So, it can be said that considering item similarities and applying CB filtering approach improves the results of the recommendation system.

4 Conclusion and Future Work

In this paper, a hybrid recommendation system is presented. The system uses both collaborative filtering and content-based recommendation techniques. Base is the collaborative part in which a graph based algorithm called Adsorption. Content information is retrieved from both IMDb and YouTube and this is used in order to propose a better system.

Enhancement is done on the distribution list which is retrieved from the collaborative filtering. To make use of content based approaches item-item

similarities are found. According to the similarity results new movies which are not included in the result of collaborative recommendation are inserted to the list and recommended to the user.

Results of experiments show that the hybrid system has a better performance on recommendations than using pure collaborative algorithm. It is also found out that system gives more successful results when MovieLens dataset is used which means good results are obtained when the data is not sparse.

As a future work the system can be extended so that even with sparse data the system can give more appropriate suggestions to users. Beside this, the recommendation system proposed in this paper works offline and makes offline predictions. Considering video domain, the next step can be to integrate this system to an online organization where users watch videos online.

Acknowledgments

This work is partially supported by The Scientific and Technical Council of Turkey Grant ‘‘TUBITAK EEEAG-107E234”.

References

1. Ozturk, G.: A Hybrid Video Recommendation System Based on a Graph Based Algorithm, Master Paper, Middle East Technical University (2010).
2. Adomavicius, G., Tuzhilin, A.: Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, IEEE Transactions on Knowledge and Data Engineering, VOL. 17, NO.6 (June 2005).
3. Balabanovic, M., Yoav Shoham, Fab, “Content-based, collaborative recommendation.(Special Section: Recommender Systems)”, Communications of the ACM vol.40, pp. 66 (1997).
4. Baluja S., Seth R., Sivakumar D., Jing Y., Yagnik J., Kumar S., Ravichandran D., Aly M.: Video Suggestion and Discovery for YouTube: Taking Random Walks Through the View Graph. In the Proceedings of WWW (2008).
5. YouTube API, http://code.google.com/apis/youtube/
6. Debnath S., Ganguly N., Mitra P.: Feature weighting in content based recommendation system using social network analysis, WWW (2008).
7. MovieLens, http://www.movielens.org/
8. C. J. van Rijsbergen: Information Retrieval. London; Boston. Butterworth, 2nd Edition (1979).
9. Harman D., Candela G.: Retrieving Records from a Gigabyte of Text on a Minicomputer Using Statistical Ranking. Journal of the American Society for Information Science (December 1990).

METADATA EXTRACTION FROM TEXT IN SOCCER DOMAIN

Ziya Ozkan Gokturk[1], Nihan Kesim Cicekli[2] , Ilyas Cicekli[3]

[1]SIEMENS EC, METU, Technopolis, Ankara, 06531, Turkey
[2]Department of Computer Engineering, METU, Ankara, 06531, Turkey
[3]Department of Computer Engineering, Bilkent University, Ankara, 06800, Turkey

Abstract — Event detection is a crucial part for soccer video searching and querying. The event detection could be done by video content itself or from a structured or semi structured text files gathered from sports web sites. In this paper, we present an approach of metadata extraction from match reports for soccer domain. The UEFA Cup and UEFA Champions League Match Reports are downloaded from the web site of UEFA by a web-crawler. Using regular expressions we annotate these match reports and then extract events from annotated match reports. Extracted events are saved in an MPEG-7 file. We present an interface that is used to query the events in the MPEG-7 match corpus. If an associated match video is available, the video portions that correspond to the found events could be played.

***Keywords* — Semantic querying of video content, MPEG-7, information extraction, video annotation**

I. INTRODUCTION

Sport fans could not watch every live game because of several reasons such as time and region differences, channel availability etc. There will be highlights of games but they are generally prepared by studio professionals and they do not cover the audience's appetite. On the other hand, people want to see events related to certain teams or players.

Unfortunately, not all multimedia have metadata available with them. Content based knowledge extraction from large multimedia repositories is an important research area. For multimedia data without semantic content tags, it is necessary to extract the metadata automatically. Web content is usually inside XML or HTML documents, which contain additional information that can be used to obtain the metadata by applying natural language processing techniques and information extraction algorithms.

In this paper, we present a system that annotates soccer game videos automatically by using the information in match summary texts. The proposed system downloads live match reports from UEFA by a web-crawler. It, then, tags these match reports by regular expressions. All events are extracted from match reports via a hand-written rule set. These extracted events are converted to valid MPEG-7 [7] files and match corpus is generated. A user interface is provided for querying and searching the match corpus. Relevant video segment of the game is displayed for successful search results.

The rest of the paper is organized as follows. The related work on metadata extraction in sports domain is given in Section II. Section III gives brief information about the web-crawler used for downloading minute by minute match reports. Section IV demonstrates annotating text using regular expressions. Section V explains the MPEG-7 ontology and describes how the mapping of football events to that ontology is done. Section VI summarizes the implementation of our system and finally Section VII concludes the paper and gives some comments about future work.

II. RELATED WORK

Information extraction from different kinds of sources is so popular nowadays. Especially for multimedia content annotation, information extraction can be preferred over video analysis if an associated text is available with the multimedia data. For instance, it is easy to find text describing videos in sports domain, specifically soccer videos. Popular sport sites publish match reports in a structured or semi structured format where events of the game are summarized along with their time information. The video segments can be annotated by aligning them with the time information and extracting the metadata from the text in summaries.

There exist many projects that have been developed for metadata extraction for sports events. In [3,4] a framework is proposed for detecting events from live sport videos and also live text analysis. They have four modules that are live text/video capturing, live text analysis, live video analysis and live text/video alignment. Live text analysis module extracts the events from text and then these events will be synchronized with video with the live text/video alignment module. Here the main concern is to increase the precision of their video analysis

techniques using text as a second source of information. Our work is focused more on semantic querying of the matches with a more detailed description of events using the textual descriptions.

Information extraction by template pattern matching is used to summarize football matches in [6]. They use GATE for intermediate analysis results. Their focus is semantic processing in information extraction. They process documents in English and use machine learning.

An automatic audio video summarization tool is presented in [5]. The tool uses content-based metadata which is extracted from match summaries manually. Once an ontological metadata is provided, the system tries to generate summaries of the game.

SOBA [2] is an ontology based approach for metadata extraction from match reports in soccer domain. SOBA automatically downloads match reports from UEFA and FIFA and sends them to a linguistic annotation web service. After that, applying the rule set to the annotated documents, events are extracted. The extracted events are stored in their own format. In our work, on the other hand, we use MPEG7 standard which makes our system more interoperable.

The work in [1] aims to extract events from both tabular match reports which are structured, and from minute by minute match reports which are unstructured. They use video analysis results and combine them with several textual resources. The aim is to discover the relations among six video data detectors and their behavior during a time window that corresponds to an event described in the textual data whereas we are concerned with semantic querying of the game contents in our work.

Compared with previous approaches, the contributions of our approach include the mapping of match events into the MPEG-7 ontology. The usage of MPEG-7 standard makes our corpus interoperable with other systems. Besides that, synchronizing match events and match videos with MPEG-7 standard provides convenience. The search and query operations on MPEG-7 files are managed by XQuery language. Since there are many search options, we obtain a dynamic XQuery generator over MPEG-7 files. If a match video is found as the result of querying, it could be displayed according to the time of the event. The synchronization of video and event is accomplished by the minute information of event that is gathered from the MPEG7 file.

III. WEB CRAWLER

The web-crawler is used for producing the match corpus. The match data is formed from HTML documents of official web site of UEFA[1]. There are two big organizations that UEFA arranges, UEFA Champions League and UEFA Cup. These organizations have their own web sites and match centers. In their match centers, for each match, the data source contains semi-structured data of player names, referees, match result and scorers of the match. Minute by minute match report is also provided at match center. Minute by minute match reports informs people about the events at the game in a textual form. These reports include the events, performers of events and their exact time point. The crawler is able to extract minute by minute match reports from Champions League and UEFA Cup match centers. At the fixtures and results part of each competition, there are links to the match days and in each of those match days, there are links to the match reports. Therefore, match report links could be found by crawling from fixtures and results site. For each match report link, minute by minute reports are extracted and saved to the match corpus.

IV. ANNOTATING TEXT USING REGULAR EXPRESSIONS

In minute by minute match report, an event is described by one sentence which contains the exact time point of the event, performers of the event and teams of the performers. The structure of these sentences is well-defined. Therefore, extracting events from sentences could be done by matching the sentences with a template that consists of labeled match events. Besides event types, time point of event, performers of event, teams of performers could be extracted from minute by minute match report. Before extracting event types, text must be labeled or in another words tagged. Since the text is well-defined, tagging of teams, players and minutes could be performed with regular expressions. In UEFA match reports minute information takes part at the beginning of the sentence and team information follows a player name. Player names are proper nouns and always start with uppercase letters. The team of a player is indicated by a team name in parentheses. Minute is represented by numeric values. However for extra time of normal match, numeric value is followed by a plus character and then another numeric value.

[1]http://www.uefa.com/

For extra time after the match, prefix Ex. is used and it is followed by numeric values. Apart from the annotated parts, other words and punctuations are labeled as token.

After a document containing match summaries is downloaded by the crawler, it is processed by tagging each sentence properly. The minute by minute text is transformed into an XML document where each sentence is represented as a separate element. The XML file contains labeled words and tokens under the sentence element. Converting plain match text into structured XML files ease applying information extraction algorithms on the match corpus. Figure 1 shows a tagged form of the sentence: "87: Crouch (Liverpool) has an effort on goal."

```
<Sentence>
    <Minute>87</Minute>
    <PlayerName>Crouch</PlayerName>
    <Token>(</Token>
    <Team>Liverpool</Team>
    <Token>)</Token>
    <Token>has</Token>
    <Token>an</Token>
    <Token>effort</Token>
    <Token>on</Token>
    <Token>goal</Token>
    <Token>.</Token>
</Sentence>
```

Figure 1*:* A Tagged Sentence

In order to extract event information from the processed texts we prepared a rule set for each soccer event. We have identified all distinct events appearing in match reports and identified different sentence structures for each of these events. For each event type there is a set of hand-written rules. These rules are applied to the data set to extract the events in another XML file for each match. Rule set can be thought as a template for match corpus events and the extracted information on that event. For each sentence in a match report, it compares the patterns of hand-written rules and if there is a match, it will extract it from the rule set and fill it with the specified information in the sentence.

Figure 2 shows an example rule for discerning the corner event. Under the *Rule* element there are two sub elements *Pattern* and *MatchEvent*. *Pattern* is the template for the event. If the tagged sentence is coherent with a pattern of the rule, the corresponding event will be extracted according to the *MatchEvent* element of the rule. Template matching is done by matching the field name of the sentence with the pattern sub-element of rule element first and if the field name is token match the content also. In Figure 2, for corner event, there are three extracted information: minute, player name and team. That information is filled from the tagged sentence by matching the field names. After all sentences are scanned, an *xml* file is generated according to the extracted information from the rule set. The *xml* file contains the events in the format of *MatchEvent* element that is described under the *Rule* element.

```
<Rule>
    <Pattern>
            <Minute></Minute>
            <PlayerName></PlayerName>
            <Token>(</Token>
            <Team></Team>
            <Token>)</Token>
            <Token>delivers</Token>
            <Token>the</Token>
            <Token>corner</Token>
            <Token>.</Token>
    </Pattern>
    <MatchEvent>
        <CornerEvent>
        <Minute></Minute>
        <PlayerName></PlayerName>
          <Team></Team>
        </CornerEvent>
    </MatchEvent>
</Rule>
```

Figure 2: A Sample Rule for Extraction

Table 1 lists all match events and the extracted fields for the events that we can process. For each of these events, all possible sentences are examined and rules that are similar to the one in Figure 2, are created. There are two important points in creating the rules: The first element *Pattern* is the template that will be matched with the sentences in match summaries and the second element *MatchEvent* represents the extracted information. As it is seen in Figure 2, the extracted information for corner event such as *Minute*, *PlayerName* and *Team* has no content. If this rule is matched with a sentence, they will be filled by gathering values for these fields from the sentence.

Match Events	Extracted Information
Cautioned	(Minute, PlayerName,Team)
Corner	(Minute, PlayerName,Team)
Foul	(Minute, FoulCommittedPlayerName, FoulCommittedTeam, FoulSufferedPlayerName, FoulSufferedTeam)
Free-Kick	(Minute, PlayerName,Team)
Goal	(Minute, PlayerName,Team)
FreeKickGoal	(Minute, PlayerName,Team)
PenaltyGoal	(Minute, PlayerName,Team)
OwnGoal	(Minute, PlayerName,Team)
GoalPosition	(Minute, PlayerName,Team)
Offside	(Minute, PlayerName,Team)
PenaltyEvent	(Minute, PlayerName,Team)
PenaltyMiss	(Minute, PlayerName,Team)
Redcard	(Minute, PlayerName,Team)
YellowCard	(Minute, PlayerName,Team)
Substition	(Minute, SubstitionInPlayerName, SubstitionOutPlayerName,Team)
SaveGoal	(Minute, PlayerName,Team)

Table 1: Match Events

V. CREATING MPEG-7 METADATA

It is preferable that we store match events in a standardized manner so that other systems can use the same match corpus for their systems. Besides, we want match events to be synchronized with the football videos. For this purpose, we use MPEG-7 standard to keep the semantic annotations of the games.

MPEG-7 (Multimedia Content Description Interface) developed by MPEG (Moving Picture Experts Group) aims at standardizing the annotation of multimedia content especially for interoperability purposes. XML syntax is used for Description Definition Language (DDL). It allows the creation of MPEG-7 Description Schemes and Descriptors. The use of XML smoothens the ability to work with other metadata standards.

MPEG-7 descriptions collaborated with audiovisual data content may comprise of pictures, graphics, 3D models, audio, speech, video, and composition information about how these elements are joined in a multimedia scenario. MPEG-7 descriptors do not depend on how described content is stored or coded. MPEG-7 description can be created for a picture or an analogue movie in the same way as a digitized content. MPEG-7 permits different granularity in its descriptions to have different level of discernment. It does not depend on the representation of material. If the material has certain relations in time and space, it will be possible to attach descriptions to elements within the scene. Since the descriptions can be attached to time and space relationship, it will be adapted to the context of an application.

The MPEG-7 standard allows querying the metadata and synchronizes it with the audiovisual (AV) content. It involves the descriptors that are associated with AV. Attributes for AV content such as location; time and quality are described in MPEG-7 Description Schemas. The description Schemas allow more complex descriptions by declaring relationships among the description components. In MPEG-7 descriptions are arranged into categories of multimedia, audio and visual domain. Their combination and possibly textual data related to them could be described in content of Description Schemas.

```
<SemanticBase id="Prica_AaB_45+1_1" xsi:type="AgentObjectType">
  <Label>
    <Name />
  </Label>
  <Definition>
    <FreeTextAnnotation />
  </Definition>
  <MediaOccurrence>
    <MediaLocator xsi:type="TemporalSegmentLocatorType">
      <MediaTime>
        <MediaTimePoint>45+1</MediaTimePoint>
      </MediaTime>
    </MediaLocator>
  </MediaOccurrence>
  <Relation type="urn:...:agentOf" target="Cautioned" />
  <Relation type="urn:...:memberOf" target="AAB" />
  <Relation type="urn:...:hasAccompaniertOf" target="ANDERLECHT" />
  <Agent xsi:type="PersonType">
    <Name>
      <GivenName>Prica</GivenName>
      <FamilyName />
    </Name>
  </Agent>
</SemanticBase>
```

Figure 3: Example MPEG-7 Descriptors

In our work, match corpus have an MPEG-7 file for each match that is extracted from minute by minute match reports. In each file, all events and their time points are represented according to MPEG-7 Descriptions Schemas. MPEG-7 standard allows us to define semantic content under the *Semantic*

Description. *SemanticBase* element under the *Semantics* is used for performers of events with type *AgentObjectType* which let us to describe performer under the *Agent* element. *MediaTimePoint* under the *MediaTime* is used for the minute information of the event. For the team information, event and opponent team information, we used the *Relation* element. For event name, relation type will be *agentOf*; for team information of the performer relation type will be *memberOf* and for the opponent team relation type will be *hasAccompaniertOf*. Player name is stored under the *Agent* element with type *PersonType*.

There are some events that have hierarchical representation such as foul event which is a combination of *FoulCommitted* and *FoulSuffered* event. For these events, two separate events are created and the relationship between these events is guaranteed by another *SemanticBase* element.

VI. IMPLEMENTATION

In our work, we download match reports from UEFA web site by crawling and we extract match events from these reports. We use the NekoHTML for parsing the UEFA web site which is a simple HTML scanner that enables parsing the HTML documents and accessing the information using standard XML interfaces. We transformed the extracted match events into MPEG-7 files for each match.

In our system, there is a user interface for querying the matches. We use XQuery for searching the MPEG-7 files. To include XQuery into our system, we use XQEngine library that is full-text search engine for XML documents, utilizing XQuery as its front-end query language.

There are minute based, player name based, team based, match based and event based search options in our system. User can select one of them or a combination of them. Since there are multiple search options and we are using XQuery, we implemented a module that dynamically generates XQuery according to search options. This module adjusts the necessary joins in an efficient manner.

After the search operation, the user can select one of the results, and if a match video is associated with that match, its video is shown at the top of search results. While playing the video, we use Java Media Player API, a portion of the Java Media Framework (JMF) that enables audio and video within applications.

In Figure 4, Corner events that Inter team has taken against Liverpool are shown. The result has shown the event name, minute of the event, the player who takes the corner and the match result.

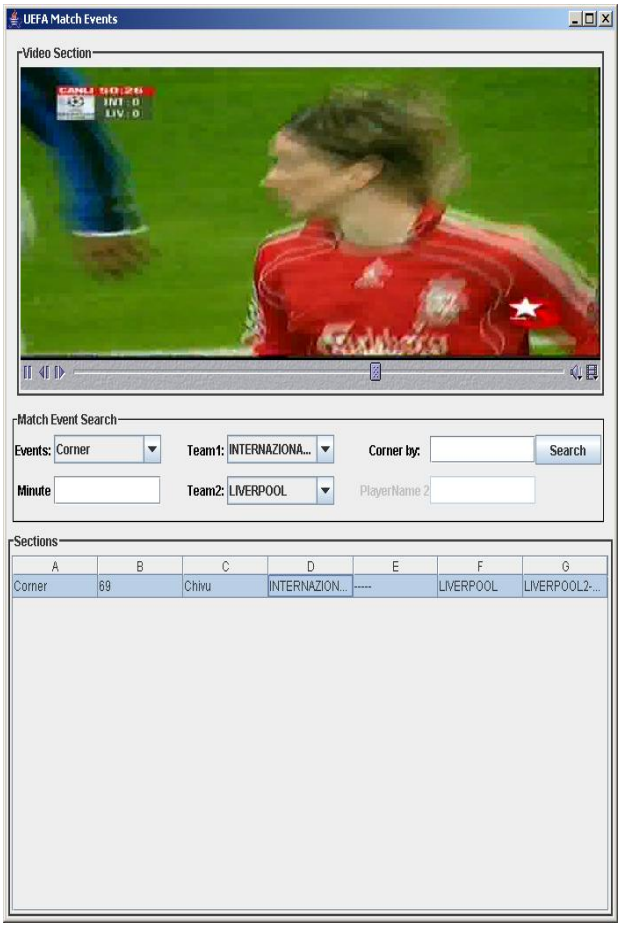


Figure 4: Displaying results for event-based querying

Figure 5 illustrates querying of all events that player Gerrard was involved in games where Liverpool and Internazionale are opponents. Goal Position at 25 minute is selected and that event is shown at the top of search results.

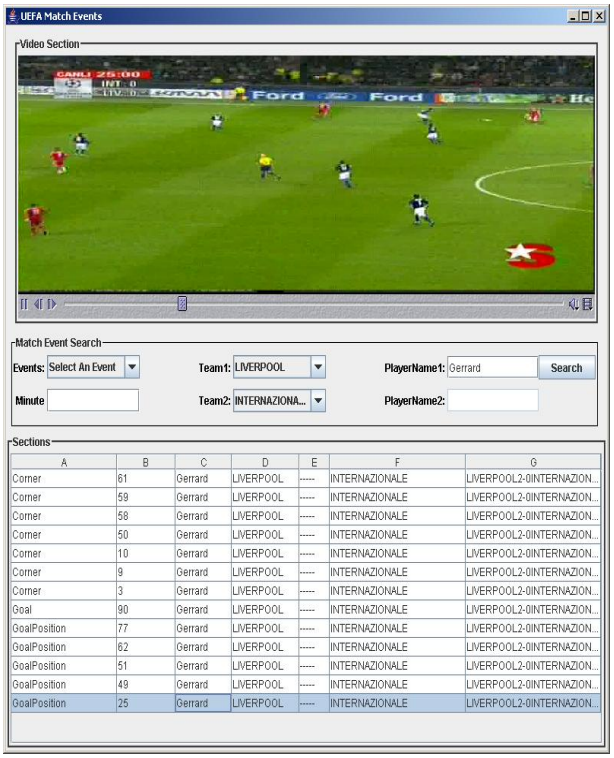


Figure 5: Player-based querying

VII. CONCLUSION AND FUTURE WORK

We present an MPEG-7-based approach to information extraction in soccer domain that targets the automatic generation of MPEG-7 files from match reports. We choose the MPEG-7 standard for our match corpus to make the system more interoperable.

A web crawler is used for downloading the match reports from UEFA web site. These match reports are annotated using regular expressions. According to the hand-written rule set, match events are extracted and converted to the MPEG-7 standard.

Finally, we adapted an interface for querying match events over MPEG-7 files by using XQuery Language. However since there are several kinds of query types we need a module that dynamically prepares XQueries according to search criteria.

Future work includes the replacement of hand-written rules with machine learning algorithms. We plan to learn the template rules for each soccer event from the match summaries. In this way the system will be more flexible and it will be easy to adapt it to a more variety of game narrations. Since we have a modular architecture, after labeling match reports we associate them with events and run information extraction algorithms on that match reports corpus.

ACKNOWLEDGMENTS

This work is partially supported by The Scientific and Technical Council of Turkey Grant ‘‘TUBITAK EEEAG-107E234.

REFERENCES

[1] Jan Nemrava, Paul Buitelaar, Vojtech Svatek and Thierry Declerck, “Event Alignment for Cross-Media Feature Extraction in the Football Domain” In: Proceedings of WIAMISS'07, Santorini, 2007, IEEE Computer Society.

[2] Paul Buitelaar, Thomas Eigner, Greg Gulrajani, Alexander Schutz, Melanie Siegel and Nicolas Weber, “Generating and Visualizing a Soccer Knowledge Base”, In Proceedings of the EACL’06 Demo Session, Trento, Italy, 2006.

[3] Changsheng Xu, Jinjun Wang and Yifan Zhang, “A Novel Framework for Semantic Annotation and Personalized Retrieval of Sports Video”, IEEE Transactions on Multimedia Vol 10(3) 421-436.

[4] Changsheng Xu, Jinjun Wang, Kongwah Wan, Yiqun Li and Lingyu Duan, “Live Sports Event Detection Based on Broadcast Video and Web-casting Text”, In Proceedings of the 14th annual ACM international conference on Multimedia, 2006, pp.221-230.

[5] Catherine Dolbear and Michael Brady “Soccer Highlights generation using a priori semantic knowledge”, In Proceedings of International Conference on Visual Information Engineering, 2003.

[6] Milena Yankova and Svetla Boytcheva, “Focusing on Scenario Recognition in Information Extraction”, In Proceedings of EACL, 2003, pp.41-48..

[7] http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm

Ontological Video Annotation and Querying System for Soccer Games

Ozgur Alan, Samet Akpınar, Orkunt Sabuncu, Nihan Cicekli and Ferda Alpaslan

Department of Computer Engineering, Intelligent Systems Lab, M.E.T.U, Ankara, Turkey.

Abstract **-** This paper describes a video annotation and querying system which is capable of semi-automatic annotation of videos from text. The extracted metadata is aligned with the corresponding video segments. This allows users to query videos according to their semantic content. We have chosen soccer domain to demonstrate the use of the system. The soccer videos are very suitable for our framework, since it is easy to find web-cast match reports for soccer games. The annotated videos are stored in MPEG-7 format in an object-oriented database. The keyword-based indexing allows fast retrieval of video segments. The system accepts match reports in Turkish which makes querying in Turkish possible.

***Index Terms*— video semantic annotation, information extraction, ontological querying, MPEG7**

I. INTRODUCTION

Multimedia content is being used in a wide number of domains ranging from commerce, security, education, entertainment and sports. Especially, with the advances in digital technologies, there is a natural increase on demands to store, organize, and query videos. People want to search and find the audio-visual content according to its content semantics. They want to find quickly the right spot in the right video by only describing what they want, preferably in free text format. In order to achieve this there should be knowledge about the content. This knowledge comes from the metadata of the content, which are stored along with the videos as annotations.

There has been significant amount of work on semantic annotation of videos automatically. Most of current research focuses on event recognition and classification based on the extraction of low-level features. Such approaches are, however, mostly limited to a very small number of different event types, e.g. detecting a moving person, a goal event, recognizing specific objects, etc. On the other hand, for some videos, there are vast textual and semi-structured data sources that can serve as a valuable source for more semantic event recognition and classification.

The idea behind the research in this paper is the use of the accompanying text to extract semantic metadata for videos and align the extracted metadata with the frames of the video. This will allow querying the video segments according to their semantic content described by words. The metadata associated with the video should be rich enough to search content semantically. The capacity of semantic search depends very much on the amount and detail of the metadata. For this reason, we aim to develop an ontological annotation environment where video metadata is generated automatically from text using information extraction techniques and queried in free format. This paper presents our preliminary experimental results of our research to achieve this aim. We have implemented a video annotation and querying system for videos in soccer domain. Since we want to do ontological annotation and querying, we restricted ourselves to a certain domain. Soccer domain is attractive due to its available ontology, well-defined events, and semantic structure. Furthermore, there is a wide range of information resources to match reports. For instance, web-casting text can be used along with the actual video of the game. In this work, we focused on Turkish match reports available on different web sites, such as sports pages of national newspapers or Sporx.com. The match reports are usually in the form of a semi-structured text. For instance, 'Dakika 60: Gol, Türkiye 1-0 öne geçti' (Minute 60: Goal, Turkey scored 1-0) can be the description of a goal event.

We implemented an annotation tool which can be useful for two types of users: Those users who want to create match reports manually, and those who want to annotate a match (semi)automatically given a match video and a downloaded match report. Once the video is annotated, it can be queried semantically. The implemented system allows the users to query the games using keywords or a combination of keywords, and view the corresponding segments. The annotated videos are stored in MPEG-7 format. An indexing component is implemented for fast retrieval of queried events and video segments.

The rest of the paper is organized as follows. Section 2 summarizes the recent work on semantic annotation and retrieval of sports videos and compares our approach with them. Section 3 describes our approach to annotate and store the videos. Automatic annotation tool is presented in Section 4. Indexing and keyword-based querying are explained in Section 5. We discuss our current work on the extension of

the system for ontological querying in Section 6. Section 7 concludes the paper with some remarks about future work.

II. Related Work

Sports events, especially soccer games, have long been studied as an application domain for video analysis, video annotation and semantic querying. There has been a considerable amount of work on automating the annotation of soccer matches.

In [3,4] a framework is proposed for detecting events from live sport videos and also live text analysis. They have four modules that are live text/video capturing, live text analysis, live video analysis, and live text/video alignment. Live text analysis module extracts the events from text and then these events are synchronized with video using the live text/video alignment module. This work is concerned with the improvement of their video analysis techniques by aligning it with the broadcast text. They focused on detecting only a few soccer events. However, they are not interested in semantic querying of the match contents.

Information extraction by template pattern matching is used to summarize soccer matches in [6]. They use GATE (General Architecture for Text Engineering) intermediate analysis results. Their focus is semantic processing in information extraction. They process documents in English and use machine learning algorithms to find templates for the description of soccer events. On the other hand, our work is concerned with match summaries in Turkish, and currently, we do not employ any machine learning techniques to automatically extract events from text. However, we plan to extend our framework with that facility applied to Turkish texts as a future work.

An automatic audio video summarization tool is presented in [5]. The tool uses content-based metadata which is extracted from match summaries manually. Once an ontological metadata is provided, the system tries to generate summaries of the game.

SOBA [2] is an ontology based approach for metadata extraction from match reports in soccer domain. SOBA automatically downloads match reports from UEFA and FIFA, and sends them to a linguistic annotation web service. After that, events are extracted by applying the rule set to the annotated documents,. The extracted events are stored in their own format. In our work, on the other hand, we use MPEG-7 standard which makes our system more interoperable.

The work in [1] aims to extract events from both tabular and from minute-by-minute match reports. They use video analysis results and combine them with several textual resources. The aim is to discover the relations among six video data detectors and their behaviour during a time window that corresponds to an event described in the textual data whereas we are concerned with semantic querying of the game contents in our work.

III. Video Annotation

In order to make the video archives searchable, video metadata plays an important role for representing the video content. As the video metadata should include the content descriptions, metadata creation is a part of the studies including video querying. Herein, video annotation is the action of creating metadata for video objects. Regarding its definition, video annotation is an important part of this study.

The annotation process of a video varies depending on its creation methodology. In this study, two ways of annotation is used:

1) Manual Annotation
2) Automatic Annotation

Manual annotation is carried out by accepting the users' manual descriptions about the video content by using an annotation tool developed for this purpose. This annotation tool includes the dynamically created data fields according to the target domain. Therefore, the structure of the tool depends on the domain where the group of videos resides. The annotation tool using soccer domain is shown in Fig. 1.

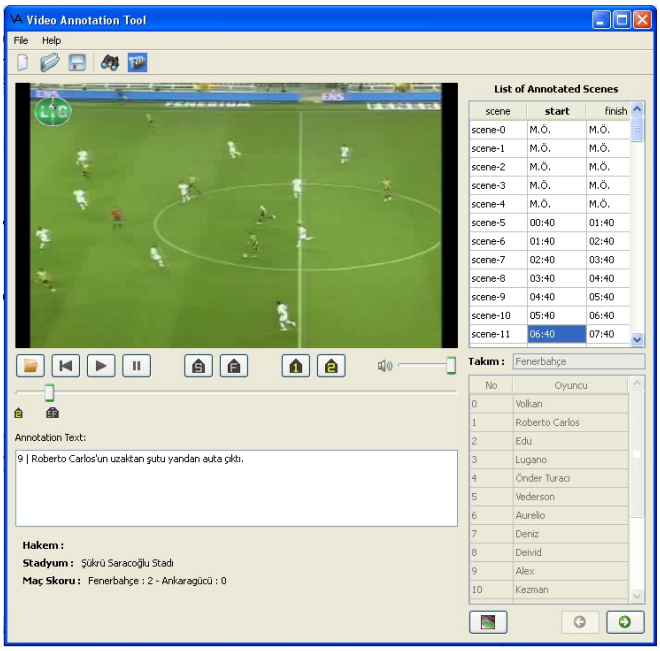


Fig. 1: Annotation Tool

Automatic annotation, on the other hand, makes use of web casting to get the information about the video from the live web sources. For instance, text based live soccer match reports are important web casting sources for the soccer domain. While web casting contributes to the system with its text-based source, a synchronization problem arises. For this reason, text/video alignment between the video and the metadata source is provided.

The creation of semantic metadata is an important feature of the video annotation in this study. Semantic metadata is needed to query the archive more semantically. In order to

obtain semantic metadata, we need a conceptual data model for videos. We create a soccer ontology to represent the concepts of the domain. The proposed model is an ontology-based model and includes high-level features of the relevant domain. Video segments, objects, events, and their relations form the basic part of the model.

The standards for representing the metadata come into prominence when determining a model for the metadata,. At this point, MPEG-7 appears as a common standard for describing the video content which supports the semantic and free-text annotation patterns. An MPEG-7 file is created and stored in an object oriented video database (db4o[1]) at the end of the annotation process. . The database design is formally in line with the data model, in this way.

In broad view, the annotation process is a combination of different actions. When a video segment is needed to be annotated, its manual or automatic annotation process starts. After obtaining the annotation data, video content is represented with MPEG-7 according to the predefined video data model. Then, the MPEG-7 information is indexed using the indexer module. The database storage is composed of raw video and metadata storage. Metadata is stored using the object oriented database whereas, the raw video data is not stored directly in the database; instead, its existing file pointer in metadata is stored to match the path with the metadata..

IV. Automatic Annotation of Video Content

Since extracting the high level events from soccer videos automatically is a very difficult task, we propose to use external text sources giving the highlights of soccer games. There are many advantages of using external text sources to detect important events of the soccer games:

1) The analysis of a text source in a specific domain eases the information extraction compared to video analysis since the textual patterns of the events in soccer domain are restricted.
2) There is a great amount of web sources for soccer games. Therefore, redundancy can be exploited in text analysis.
3) The text sources also give the time of the events in a detectable way. Therefore, the time information of events provides an easy way of alignment of events extracted from text with the related video segment.

We aim to match the extracted information from Turkish text with the concepts in our soccer ontology and align this semantic information with the related video segments. As a result, our model will not only provide a keyword-based search but also a semantic search allowing logical queries on videos.

Automatic annotation starts by crawling the web site related with the chosen domain which is Turkish soccer games, in our case. Our tool parses the downloaded web pages by converting html files into text files. This pre-processing prepares the external sources for information extraction. In first phase of information extraction, the general information about the soccer games is extracted. For example, the teams of the game, the squads of the teams, the referee of the game, score of the game, location of the game, etc. In addition to this kind of general information, the event minutes and the free text corresponding to the events are extracted in this phase. In the second phase of the information extraction, the free text labeled with minutes is analyzed. The events and their attributes are extracted from the free text and this information instantiates the concepts in the ontology (ontology population). At the end of this task, the video associated with the extracted events is annotated with the concepts and their instances in the ontology. The annotation files are indexed for keyword-based search and also stored in a database. By this way, the extracted information becomes ready to use for knowledge base of logical reasoning systems.

The modules in automatic annotation use various technologies and methods. The technologies and the methods in the modules are summarized in the following subsections.

A. *Web Source Crawling & Parsing*

This module converts a website into a set of text file. Each text file involves the necessary information for a soccer game. Nutch[2] is used to crawl and parse the web pages taken from the source web site sporx.com[3]. The issues in crawling and parsing are resolved by the Nutch's related components. NekoHTML is chosen in order to parse the web pages.

B. *Information Extraction & Free-text Annotation*

Sporx.com presents the players, team info and game's info in a structured way. However, the events and corresponding minute information are edited in freestyle. The information extraction in first level only obtains the structured information in the text files (team info, game info and minute info). This extracted information is written into an MPEG-7 file. This file involves minutes and event's freestyle annotation, not related with the terms and concepts in the ontology. After this process, the MPEG-7 files become ready to be indexed for keyword-based search.

C. *Information Extraction & Ontology-based Annotation*

The free texts in the minutes are processed to detect the events and their attributes. Events (Goal event, corner event, etc.) are concepts in the ontology and the attributes of the events are the terms (teams, players, referees, stadiums etc.) in the ontologies. The events are instantiated with the results of information extraction. We plan to use semi-automated methods for ontology building.

We designed some grammars to identify the patterns that soccer game ontology events and their attributes. Currently, the grammar for the patterns of the events is extracted manually from the large corpus of "sporx.com". However,

[1] http://www.db4o.com

[2] http://lucene.apache.org/nutch/

[3] http://www.sporx.com/

these patterns are planned to be extracted in a semi-automated or automated manner. The patterns found manually are used to detect the events and their attributes such as actors, location in the field etc. After this process, the minutes in the MPEG-7 files are annotated with the concepts and the terms of the ontology in addition to the freestyle annotation. Therefore, the MPEG-7 files become ready to be used by the knowledge base and also by logical reasoners.

D. Video Data- Metadata Alignment

Although there are several ways of determining the event boundaries in videos, we only used explicit synchronization method in this paper. Currently, the first half and second half of the videos are marked with the annotation tool. The minutes of the events and video content are linked according to these synchronization marks. Minute 1 is synchronized with the first mark and minute 46 is synchronized with the second mark. After this synchronization, the minutes of the events are marked on the video as a start of the event's video segment. Video analysis methods are planned to be used to detect event boundaries in the future.

V. Indexing and Keyword-based Search

This module uses the general information about the match and the events with minutes in MPEG-7 files. Only the free text tag of events is used in indexing rather than ontology-based tags of events in the MPEG-7. MPEG-7 files that are annotated automatically or manually are parsed with an XML parser. The indexer employs this parser with an XML configuration file. The index fields and corresponding tags in the MPEG-7 file are given in this configuration file. Therefore, the indexer module becomes compatible with other domains. Lucene, a state-of-the-art indexer, is used in our work. Lucene also has a query parser and allows Google like queries (phrase queries, Boolean queries, exclude queries). Moreover, Lucene provides similarity searches, Kleene star searches and other queries. Each event with a minute is indexed as an entry in event index and the general information about soccer games are indexed in another index. Both indices have a game id field not to loose the relation between the match information and the events information. However, our search module behaves as if it uses only one index. Our specialized query parser manipulates the queries by using the fields in the query.

Example. +Besiktas +goal is a query that is used for bringing the soccer videos where Besiktas is one of the teams and there is a goal action.

Search module displays search results as in the Fig. 2. The video segments related with queries are presented with a snapshot of the start of the video segment. Each event entry in the annotation file and index has an image file which represents the video segment of this event. Therefore, the image file of the corresponding event is a field in the index but index only stores the path of the image, not the image itself. The images are created when the video is synchronized with the minutes of the text source.

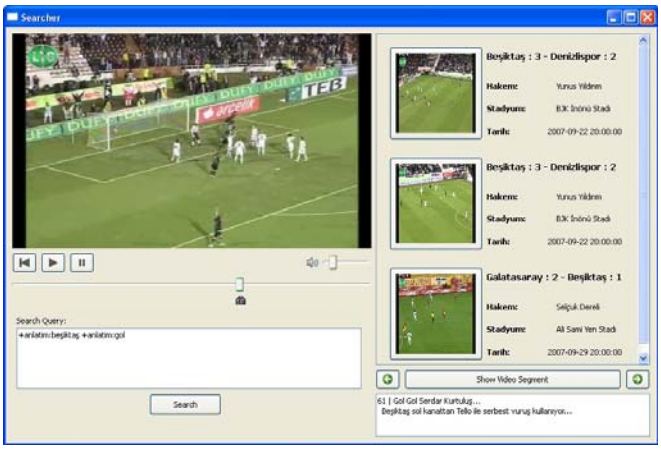


Fig. 2: Querying and Search Results

VI. Discussion on Semantic Search Of Soccer Videos

The key feature of our framework is to enable users search within the video archive. The desired scenes from the video archive should be retrieved as a result of the user query. In our application domain, these scenes can be goals of a player, or a team in a match or a season, missed goals or penalties, etc. Although some of the queries can be answered by keyword search, some scenes or videos can be missed or incorrectly displayed as a result. In this sense, we want to use semantic search in order to increase the quality of search capabilities of the framework.

Assume a soccer video has a goal event which is annotated with the knowledge that a player has made an "assist". The rules of the domain state that an assist is a pass that results in a goal. We can reason from the knowledge of assist that the corresponding scene is a goal event. This kind of reasoning is a form of semantic reasoning. Using semantic reasoning a query that searches the scenes with the goal event can be processed in a sound way.

We will utilize the domain ontology for semantic search. The ontology based annotation will produce semantic level metadata for videos. We use this metadata to form a knowledge base for the videos in the database. The reasoning capability is necessary for generating inferences from the knowledge base. These inferences will constitute answers for semantic queries. The nature and form of the knowledge base will vary according to the reasoning scheme selected for semantic search. We plan to use logic based reasoning schemes. For a description logic reasoner, the knowledge base should be composed of description logic facts that are generated from the MPEG-7 metadata of videos. The rules of the domain will also be represented in the knowledge base. For logic programming based reasoning scheme the knowledge base will hold the metadata of videos as logical facts (for instance, Prolog facts). Moreover, semantic queries should be transformed to knowledge base queries. The reasoner answers these queries by displaying the related

videos and scenes. Below is an example that shows the advantage of semantic search over keyword based search:

Example. Consider two soccer videos (V1 and V2). In V1 there is a goal scene that is annotated by 'Player A makes an assist to Player B'. In V2 there is a scene where the player B misses a goal. It is annotated by 'Player B missed a goal'. If the user searches for the goals of the player B, he expects to see the scene in V1. If the system uses keyword search, the query will be similar to the form 'B goal'. Since the annotation of V2 includes both keywords B and goal, the keyword search of this query will output V2 as the answer. However, if the system uses semantic search, it will infer from the assist made to the player B that he has scored a goal. The output of the query will be V1, which is correct. The keyword based search cause the system to have low precision (it has incorrectly retrieved V2) and recall (it has missed the correct answer V1).

The following queries are also suitable for semantic search:

Q1: all the goals by Hakan or Semih
Q2: all the matches where Lincoln could not score a goal
Q3: all the matches where Hakan missed a goal more than 5
Q4: all the matches where Nihat scored a goal and Turkey has won
Q5: all the goals by Hakan to Rustu

There are some keywords in these queries which need special attention in terms of the semantic reasoning scheme. The semantic of the connectives and/or should be represented in language of the reasoner. The query Q2 is interesting because it needs a scheme with the reasoning capability with negation. Since it searches for the matches where Lincoln could not score a goal, it can be hard to use keyword based search. However, Q2 can be represented as a complex query of logic programming scheme which features reasoning with negation. The query Q3 has common-sense concepts like numbers and counting. The underlying representation used in the knowledge base and the reasoner should be capable of dealing with concepts such as aggregates and comparison. In order to answer the query Q5 the system should follow several reasoning steps. First it should reason from the domain ontology that a player scores a goal to goalkeeper. The goalkeepers of the teams will be in the knowledge base as facts. The reasoner should infer from these facts and the goals of Hakan to answer the query Q5.

VII. CONCLUSION

In this paper, we present a video annotation and querying system developed for soccer videos. The system allows users to annotate games either manually or semi-automatically. Manual annotation involves playing, pausing, forwarding, or rewinding the video so that events of interest can be viewed and corresponding annotations can be entered in free format in Turkish. Semi-automatic annotation involves processing downloaded match reports and extract general information about the match, aligning the video segments and event descriptions. The extracted information is stored in MPEG-7 format in an object-oriented database.

The system allows the users to query the games according to their general information (e.g. referee, teams, players, location etc.). It is also possible to view the parts of the games according to important events. So, a user may query the scenes where a certain player scores a goal, or is involved in a foul, etc.

The work in this paper presents the preliminary results of a part of a broader research project, yet it has many distinctive features. First, the annotations and querying are done in Turkish which makes our system very attractive for football coaches, football clubs, and sports programs in Turkey. Second, we use MPEG-7 standard to store videos that adds interoperability to our system. Third, the implemented framework is designed in such a flexible way that all planned extensions can be done smoothly. For instance, the ontological features can easily be adapted and information extraction algorithms being developed for Turkish texts can efficiently be integrated with the annotation environment. The framework also allows importing different ontologies if a different application domain is used.

Future work involves exploiting video analysis techniques together with text processing. We are also planning to use natural language processing techniques to express queries in free format Turkish sentences. The ontological framework being developed for querying will help us here a lot to retrieve all relevant segments.

ACKNOWLEDGEMENTS

We'd like to thank Burak Bayburtlu and Digitürk (Digital Platform İletişim Hizmetleri A.Ş Genel Müdürlüğü) for providing us with sample videos of soccer matches to support the development of our system.

Part of this work is supported by The Scientific and Technical Council of Turkey Grant ''TUBITAK EEEAG-107E234.

REFERENCES

[1] Nemrava, J., Buitelaar, P., Svatek, V., Declerck, T., "Event Alignment for Cross-Media Feature Extraction in the Football Domain," Image Analysis for Multimedia Interactive Services, 2007. WIAMIS '07. Eighth International Workshop on, vol., no., pp.3-3, 6-8 June 2007.

[2] Paul Buitelaar, Thomas Eigner, Greg Gulrajani, Alexander Schutz, Melanie Siegel, Nicolas Weber, Philipp Cimiano, Gnter Ladwig, Matthias Mantel, Honggang Zhu "Generating and Visualizing a Soccer Knowledge Base" In: Proc. of the Demo Session at EACL06, Trento, Italy, April 2006.

[3] Xu, C.; Wang, J.; Lu, L.; Zhang, Y., "A Novel Framework for Semantic Annotation and Personalized Retrieval of Sports Video", IEEE Transactions on Multimedia, vol.10, no.3, pp.421-436, April 2008.

[4] Xu, C., Wang, J., Wan, K., Li, Y., and Duan, L. 2006. "Live sports event detection based on broadcast video and web-casting

text", In Proceedings of the 14th Annual ACM international Conference on Multimedia, Santa Barbara, CA, USA, October 23 - 27, 2006, pp. 221-230.

[5] Dolbear, C.; Brady, M., "Soccer highlights generation using a priori semantic knowledge," International Conference on Visual Information Engineering, 2003, (VIE 2003), 7-9 July 2003, pp. 202-205.

[6] Yankova, M. and Boytcheva, S. 2003. "Focusing on scenario recognition in information extraction". In Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - Volume 2 (Budapest, Hungary, April 12 - 17, 2003). European Chapter Meeting of the ACL.

Analysis of Face Recognition Algorithms for Online and Automatic Annotation of Personal Videos

Mehmet C. Yılmaztürk[1], İlkay Ulusoy[1], Nihan Kesim Çiçekli[2]

[1] Department of Electrical and Electronics Engineering, Middle East Technical University, Ankara, Turkey

[2] Department of Computer Engineering, Middle East Technical University, Ankara, Turkey

e134623@metu.edu.tr, ilkay@metu.edu.tr, nihan@ceng.metu.edu.tr

Abstract. Different from previous automatic but offline annotation systems, this paper studies automatic and online face annotation for personal videos/episodes of TV series considering Nearest Neighbourhood, LDA and SVM classification with Local Binary Patterns, Discrete Cosine Transform and Histogram of Oriented Gradients feature extraction methods in terms of their recognition accuracies and execution times. The best performing feature extraction method and the classifier pair is found out to be SVM classification with Discrete Cosine Transform features

Keywords: Facial Feature Extraction, Classification, Support Vector Machines with Multiple Kernels.

1 Introduction

In order to be able to query the semantic content of multimedia data, it must be annotated with some metadata. An efficient approach is content-based indexing and retrieval which provides some degree of automation by automatically extracting features the data. In order to speed up the labeling process, face recognition methods are employed. The literature [5, 6] considers face recognition methods for mostly offline annotation applications.

For video based recognition, methods have been tested on the videos of movies, news and TV series which include many characters and scenes. In this way, the proposed methods were used for automatic naming of characters [8, 10]. The recognition of faces in videos is challenging because of the dynamic nature of the videos involving bizarre conditions which distort the faces. Some works combined facial features with others such as information extracted from clothing or hair [8, 10] but we restrict ourselves to facial features alone because clothing or hair may show more variations than the appearance of the face.

This work focuses on the evaluation of face recognition methods that can be used for an online and semi automatic face annotation system for personal videos. All combinations of the considered state-of-the-art facial-feature extraction methods and classification methods are evaluated and compared in terms of recognition accuracies and

execution times. We have evaluated Nearest Neighborhood, Linear Discriminant Analysis and Support Vector Machines with single and multiple kernels as face recognition methods where various features such as DCT, LBP and HOG are used. We have made our tests on datasets which are composed of face images extracted from an episode of "How I Met Your Mother" TV series. We have observed that SVM with DCT features performs the best.

2 Considered Algorithms for Face Recognition

Three facial feature extraction algorithms, namely DCT, LBP and HOG features are used to extract information from face images. These features have been selected as they are robust state-of-the-art features and can be computed fast enough to use in an online learning system. For the classification Nearest Neighborhood, LDA, SVM and multiple kernel SVM are selected since these are also among the most popular algorithms and can be extended for online learning by using their sequential variants.

2.1 Feature Extraction

Since direct pixel values are sensitive to noise and localization errors, three alternative feature extraction methods have been implemented to represent data. These include DCT Features [3], LBP Features [2] and HOG features [9]. For the DCT Features 48 blocks are used which produces feature vectors of 480 dimensions. Basic LBP features with 8 neighbours are of 256 dimensions and HOG features with 35 blocks have 1260 dimensions.

2.2 Classification

Nearest Neighbor (NN). NN method is widely used in annotation applications especially as a baseline method. We take the dot product of the two vectors and normalize the result with their magnitudes to calculate their similarity measure.

Linear Discriminant Analysis (LDA). We have implemented LDA to reduce the dimension of input data. We retain 95% of the total energy. During classification, the nearest class center, which is the mean value of the projected samples for a given class, is used to find the nearest neighbor for the test samples.

Support Vector Machines with Single and Multiple Kernels. "One vs. the Rest" method has been considered for multi-class classification. Gaussian RBF Kernel function is used for the single kernel SVM and also for the Multiple Kernel SVM as base kernels.

For the multiple kernel SVM, instead of using a single kernel, linear combinations of base kernels are used. Each base kernel corresponds to a different block of the feature vector. For the DCT and HOG features, these base kernels are applied for each of the data blocks created during the feature extraction stage. Hence the number of base kernels is 35 for HOG features and 48 for DCT features. are the base kernels and are the weights corresponding to each base kernel.

3 Experiments and Results

Recognition accuracies are plotted. The execution times are also plotted for both training and testing phases as the number of training and testing samples changes. All tests are made for two distinct face datasets. The first dataset is a collection of hand-labeled face detection outputs. Viola-Jones face detector [1] has been run for every single frame throughout an episode of "How I Met Your Mother" TV series and the detected faces of target people are manually clustered. The second dataset is created by using a face tracker algorithm on the same episode. OpenCV implementation of Camshift Color Based face tracker [4] is used to track faces throughout the video. The resulting face tracks are also manually labeled and clustered. Samples from both datasets are fed to an illumination compensation algorithm before recognition is performed. In all tests, 5-fold cross validation technique is used.

3.1 Illumination Compensation

Sample face images are converted to gray-scale and resized to have a standard size of 64 (height) x 48 (width) pixels. Next, a basic and fast algorithm of illumination compensation is performed (1).

$$I_{comp}(x,y) = a + b\ log\left(\ I_{raw}(x,y) * H_{hp}(x,y)\right). \quad (1)$$

H_{hp} is a high-pass filter. In our application a = 10 and b = 2 and H_{hp} is a binomial high-pass filter of size 5 x 5 pixels.

3.2 Execution Times

All of the tests are performed on an Intel Core 2 Duo 2.20 GHz PC with 1 GB RAM. DCT and HOG feature extraction methods has been implemented in MATLAB. For LBP feature extraction, a MATLAB implementation available in [29] has been used. Nearest Neighborhood and LDA algorithms are implemented in MATLAB environment. For Single Kernel and Multiple Kernel SVM, a MATLAB interface for LIBSVM implementation [14] is used. For the calculation of the execution times, the number of classes is selected as 6.

Nearest Neighborhood. For NN, there is no training. The execution times are shown for different numbers of gallery samples (Figure 1). For each method, three different graphs are plotted with different numbers of testing samples used.

LDA. A major change in terms of execution times is observed for different features (Figure 1). This is expected since most of the computation is due to the construction of the scatter matrices and calculation of the eigenvectors where the number of feature vector

dimensions determines the size of the scatter matrices. 3000 samples are used to construct an eigenspace model.

SVM. SVM training times are shown on Figure 1 for 6 SVM models (6 classes) with a total of 3000 training samples used. Most of the computation is due to the kernel construction. In the testing phase, most of the computation is due to the construction of kernels as well; hence the number of returned support vectors is indicated for each method. Having returned a smaller number of support vectors, Multiple Kernel SVM models have smaller testing times compared to the SVM with DCT features.

3.3 Recognition Precisions

The general trend of decreasing recognition precision in Figure 2 is due to the fact that while the number of samples per person increases, the diversity of the model also becomes more complex with different looking samples of the face.

In all tests, LDA classification is observed to degrade and perform poorly with the insufficient number of training samples. SVM performs as the best classification method, but SVM training is a heavy process compared to NN and LDA. Single Kernel SVM with DCT features works best. But the testing time is higher than other SVM methods. This is due to the high number of returned support vectors.

4 Conclusions

Several state-of-the-art feature extraction and classification methods have been implemented and compared in terms of precision and execution times. Our focus is on determining a fast and robust face recognition method which can be used in an online learning application where face recognition for personal videos is to be performed. We have observed that single kernel SVM trained with DCT features gives the highest recognition accuracy. On the other hand in this method, the number of Support Vectors found is great and this yields relatively long testing times. SVM with Multiple Kernels, on the other hand, have comparable recognition accuracy to single kernel SVM, though training times are longer due to a more complex process of kernel construction. But tests show that in multiple Kernel SVM methods, fewer number of support vectors is sufficient to define the separating hyperplane which led to shorter testing times.

There is a tradeoff between testing times and training times when we consider the usage of SVM with single and Multiple Kernels. If long testing times are acceptable, Single Kernel SVM with DCT coefficients has the highest recognition accuracy. For the online automatic face annotation system where encountered face tacks are to be classified, the number of testing samples per query is not large, residing between typical ranges of 10-100 samples. And training times can be considered more important for an online learning system as the newly encountered samples are sequentially learnt in data chunks, repetitive sessions of long trainings may discourage the user from working with the system.

As a conclusion, we have decided to use a SVM with DCT features for our online automatic annotation purposes.

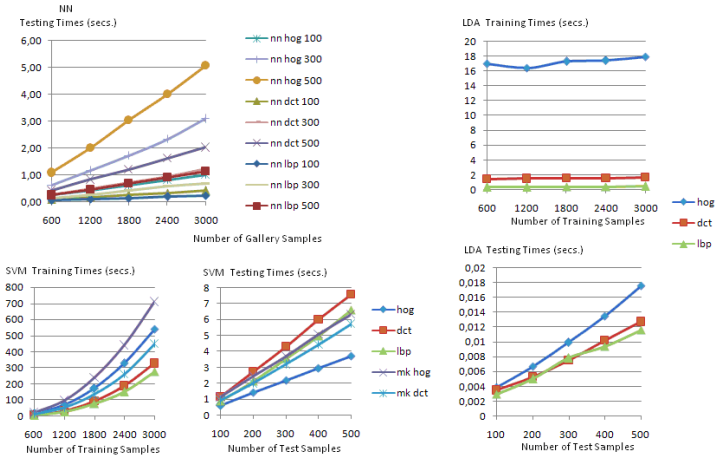


Figure 1. Training and testing times for the considered methods.

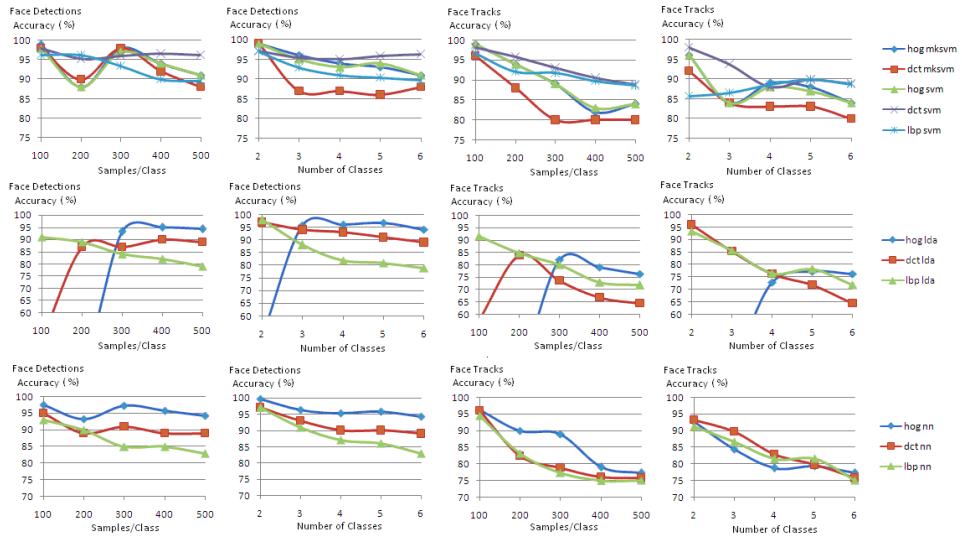


Figure 2. Recognition accuracies for different numbers of training samples and classes.

References

1. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 511--518 (2001)
2. Wolf, L., Hassner, T., Taigman, Y.: Descriptor Based Methods in the Wild. Real Life Images Workshop at the European Conference on Computer Vision (ECCV) (2008)
3. Ekenel, H.K., Stiefelhagen, R.: Local Appearance Based Face Recognition Using Discrete Cosine Transform. in Proceedings of the 13th European Signal Processing Conference (2005)
4. Bradski, G.R.: Computer Vision Face Tracking for Use in a perceptual user interface. Intel Technology Journal, 2nd Quarter (1998)
5. Jiang, R. M., Sadka, A.H., Zhou, H.: Automatic human face detection for content based image annotation. International Workshop on Content-Based Multimedia Indexing, CBMI 2008, pp. 66--69 (2008)
6. Poh, N., Chan, C.H., Kittler, J.: Face video Competition at ICB2009. Int'l Conf. on Biometrics (ICB) (2009)
7. Satoh, S.: Comparative Evaluation of Face Sequence Matching for Content-based Video Access. Fourth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 163--168 (2000)
8. Ramanan, D., Baker, S., Kakade, S.: Leveraging archival video for building face datasets. IEEE 11th International Conference on ICCV 2007, pp. 1--8 (2007)
9. Dalal, N., Triggs, B.: Histogram of Oriented Gradients for human detection. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 886--893 (2005)
10. Everingham, M., Sivic, J., Zisserman, A.: Taking the bite out of automated naming of characters in TV video. Image and Vision Computing, vol. 27, issue 5, pp. 545--559 (2009)
11. Sivic, J., Everingham, M., Zisserman, A.: "Who are you?" – Learning person specific classifiers from video. IEEE Conference on Computer Vision and Pattern Recognition, pp. 1145--1152 (2009)
12. Bach, F., Lanckriet, G., Jordan, M.: Multiple kernel learning, conic duality and the SMO algorithm. In International Conference on Machine Learning (2004)
13. University of Oulu, Machine Vision Group, http://www.ee.oulu.fi/mvg/page/lbp_matlab
14. Chang, C.C., Lin, C.J.: LIBSVM : a library for support vector machines. 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

A Content Boosted Collaborative Filtering Approach for Movie Recommendation Based on Local & Global Similarity and Missing Data Prediction

Gözde Özbal, Hilal Karaman, Ferda Nur Alpaslan

Department of Computer Engineering, Middle East Technical University
06531 Ankara, Turkey
gozbalde@gmail.com, hilal_karaman@yahoo.com, alpaslan@ceng.metu.edu.tr

Abstract. Many recommender systems lack in accuracy when the data used throughout the recommendation process is sparse. Our study addresses this limitation by means of a content boosted collaborative filtering approach applied to the task of movie recommendation. We combine two different approaches previously proved to be successful individually and improve over them by processing the content information of movies, as confirmed by our empirical evaluation results.

Keywords: Recommender Systems, Collaborative Filtering, Pearson Correlation Coefficient, Floyd Warshall Algorithm

1 Introduction

An important shortcoming of Collaborative Filtering (CF) is that when users in the system have rated just a few items in the collection, the user-item rating matrix becomes very sparse. This leads to a reduced probability of finding a set of similar users. Our study addresses this problem with a content boosted CF approach applied to the task of movie recommendation. Our main motivation is to investigate whether further success can be obtained by combining 'Local & Global User Similarity' [1] and 'Effective Missing Data Prediction' [2] approaches.

With sparse data, Global User Similarity (GUS) improves the performance of the algorithm introduced by [2], which uses only Local User Similarity. But the approach of [1] is only an improvement of user-based algorithm. Therefore, [1] asserts that approaches using both user and item-based algorithms can employ its approach to replace the traditional user-based approach to obtain a higher performance. Based on this assertion, we use a combination of EMDP and GUS concepts in our prediction technique. In addition, we process the content information of each movie to enhance these approaches.

3 System Description

We extract all necessary movie metadata from IMDb [4] by using a Python package called IMDbPY [5]. We represent each movie by a set of features including type, country, cast, genre, language, company, writer and keyword. We use two different methods for the distance measure calculation. The first one, applied to strings, checks whether two strings are equal. The second, which is used for lists, measures the cardinality of the intersection of the two lists divided by the length of the first list.

For user and item similarity calculations, we use the Pearson Correlation Coefficient (PCC) method and adopt the solution of [2], which proposes a correlation significance weighting factor in order to devalue the similarity weights based on a small number of co-rated items.

Traditional CF approaches do not take the content information into account while calculating the similarity of two items with PCC algorithm. This algorithm can work without problem for a dense user-item matrix, while there might be crucial problems with sparse data. As a solution, we process the content information of the items while calculating their similarity. We adopt the definition of [3] for the similarity between items. We use the distance measures previously mentioned and the weight values introduced by [3]. The formula that we use for the overall item similarity calculation is:

$$OverallItemSim(i,j) = (1 - \beta).CollabSim(i,j) + \beta.ContentSim(i,j) \quad (\mathbf{1})$$

where determines the extent to which item similarity relies on CF methods or content similarity.

To prevent the possibility of generating dissimilar users with the Top-N algorithm, we use the thresholds introduced in [2] with an update: if the similarity between the neighbor and user is bigger than , the former is added to the potential neighbor list sorted in terms of similarity values. The real neighbors are determined as the minimum of N and the size of the list. Item similarity calculations are done similarly.

In order to find more neighbors of users with few or no immediate neighbors, we adopt the approach of [1] so that first a user graph is constructed considering the users as nodes and the local similarity values as the weight of edges. Pairwise user maximin distance is calculated as their GUS, using Floyd-Warshall algorithm [7].

EMDP addresses data sparsity by using available information to predict the rating for a movie unrated by a user. Each prediction is assessed independently of other predictions.

4 Evaluation

4.1 Data Set, Metrics and Comparison

We conducted our experiments using the MovieLens [6] dataset containing 100,000 ratings on a scale of 1 to 5 for 1682 movies by 943 users, where each user has rated at least 20 movies. We created the same 9 configurations as [1] and [2], and used Mean Absolute Error (MAE) metrics to make the results comparable.

Table 1 - MAE comparison with state-of-the-art algorithms on MovieLens

Training Users	Methods	Given5	Given10	Given20
100	CBCFReM	0.7889	0.7653	0.7541
	CFReM	0.7893	0.7665	0.7553
	LU&GU	0.791	0.7681	0.7565
	EMDP	0.7896	0.7668	0.7806
200	CBCFReM	0.7816	0.7628	0.7533
	CFReM	0.7884	0.7637	0.7588
	LU&GU	0.7937	0.7733	0.7719
	EMDP	0.7997	0.7953	0.7908
300	CBCFReM	0.7637	0.7562	0.7384
	CFReM	0.7653	0.7616	0.7394
	LU&GU	0.7718	0.7704	0.7444
	EMDP	0.7925	0.7951	0.7552

The parameters and thresholds used for the prediction process were set to = 0.6, = 30, = 25, = = 0.6, numberOfNeighbors = 35, and = 0.5 like the experimental setup of [1]. was set to 0.5 for evaluating our CBCF approach. In Table 1, MAE comparison of our two separate prediction techniques including the one using a pure CF approach without content information (CFReM), and the other one exploiting content information (CBCFReM), with Effective Missing Data Prediction (EMDP) [2], and Local & Global User Similarity (LU&GU) [1] are summarized. It can be observed that either by using content information or not, our approach improves the recommendation quality and outperforms these algorithms in various configurations. And just like [1] asserted, when EMDP employed LU&GU to replace traditional user-based approaches, a better performance is achieved. As another conclusion, using content information in item similarity calculations improves the recommendation accuracy for all configurations.

4.4 Impact of β

To determine the sensitivity of , we conducted several experiments on all configurations in which varied from 0 to 1. The results of these experiments on movieLens100 are shown in Figure 1. Similar trends have been observed also for movieLens200 and movieLens300. During the item based prediction of a rating for a specific item, the ratings of other users in the system for that item are not processed directly. These ratings only have a contribution to the calculation of the average rating of the item. As a design issue, while making user based prediction, the users who have not rated that item are not considered as similar to the user, whereas while making item based prediction, the items who have not been rated by the user are not considered as similar to the item. Due to the second statement, in order to be able to use the content information of the items similar to the item for which rating will be predicted, the user should have rated these items. Thus, the number of ratings given by a user has importance for our overall prediction mechanism.

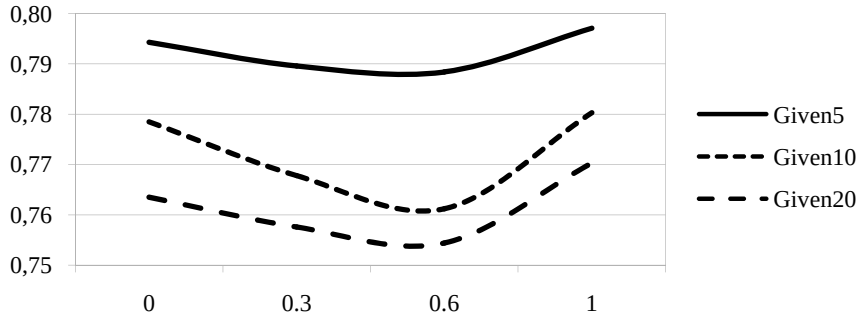


Figure 1 - Impact of Beta (*x* axis) on MAE (*y* axis) on movieLens100

For these reasons, a decrease in the MAE was observed for all configurations, when the number of user ratings increased. Experimental results also show that more accurate predictions are obtained for $\lambda \approx 0.5$. In this way, the prediction can exploit both CF and content based similarity in similar amounts, which shows that both approaches have an important and indispensable role for rating prediction.

5 Conclusion

In this paper, we presented a movie recommender system which uses a CBCF approach combining the local/global user similarity and EMDP techniques and exploits content information of the movies to handle the sparsity problem.

Empirical analysis shows that when LU&GU is employed by EMDP to replace traditional user-based approaches, a better performance is achieved. Moreover, using content information during item similarity calculations improves the recommendation quality of CF approach.

References

1. Heng Luo, Changyong Niu, Ruimin Shen, Carsten Ullrich, "A collaborative filtering framework based on both local user similarity and global user similarity," in *Proc*. of *ECML/PKDD* 2008.
2. Ma, H., King, I., and Lyu, M. R., "Effective missing data prediction for collaborative filtering," in *Proc. of SIGIR* 2007.
3. Souvik Debnath, Niloy Ganguly, Pabitra Mitra, "Feature weighting in content based recommendation system using social network analysis," *WWW*, 2008
4. The Internet Movie Database (IMDb), http://www.imdb.com
5. IMDbPY, http://imdbpy.sourceforge.net/
6. MovieLens, www.movielens.umn.edu
7. Floyd, Robert W. (June 1962). "Algorithm 97: Shortest Path". *Communications of the ACM* **5** (6): 345.

Gender and Age Groups Classifications for Semantic Annotation of Videos

Gökhan Yaprakkaya [1], Nihan Kesim Cicekli [1], İlkay Ulusoy [2]

[1] Department of Computer Engineering, METU, Ankara, Turkey
[2]Department of Electrical and Electronics Engineering, METU, Ankara, Turkey
e134811@metu.edu.tr , nihan@ceng.metu.edu.tr, ilkay@metu.edu.tr

Abstract. This paper presents a combination of methods for gender identification and age group classification for semantic annotation of videos. The system has two different running modes as 'Training Mode' and 'Classification Mode'. The gender classifier achieves over 96% accuracy and the age group classifier achieves over 87% accuracy in age group classification

Keywords: Age Group Classification, Gender Classification, LBP features, DCT Mod2 Features, Adaboost, Random Forest, Face Tracking

1 Introduction

As the vast majority of the videos contain humans, the extraction of faces from videos has become a necessity. Human faces provide lots of information about the gender and age of that human such as facial landmarks, wrinkles, eyebrows, hair, lips. The prediction of gender and age group of a person requires the detection of frontal faces, the extraction of facial features and training classifiers with these features, and use of the trained classifiers for prediction. In this study we aim to decribe a robust method to identify genders from human face, and determine the age group under uncontrolled illumination or non-uniform background. The suggested algorithms are highly competitive with the best currently available classification methods in terms of both accuracy and computational cost. DCT Mod2 and LBP feature extraction methods are extensively used in face identification and verification.

2 Face and Facial Landmark Detection and Normalization

The system mainly consists of two running modes. The first mode is the 'Training Mode' and the second mode is the 'Classification Mode'. The descriptions of the two modes are summarized in Fig. 1 and Fig.2. In order to detect faces in videos, a robust face detector is used. The main assumption of the face detection method of the system is that, whatever the ethnic group, the skin color is localized in a precise subset of the chrominance space [3,4]. Therefore, a skin-color probability model is constructed in the form of a bi-dimensional Gaussian function. The function's parameters are determined on FERET color face image database. A threshold has to be set on the probability values in order to reach a binary skin/non-skin decision for each pixel. A LUT-type boosted cascade classifier based on the concept of Viola and Jones [2] is used as face detector and skin color identifier helps to eliminate false positives. The detected frontal faces are tracked by the condensation tracking algorithm. By this way every detected face is tracked as a human and we collect four different face images of

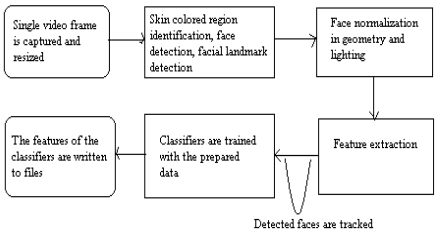


Fig. 1. Training Mode

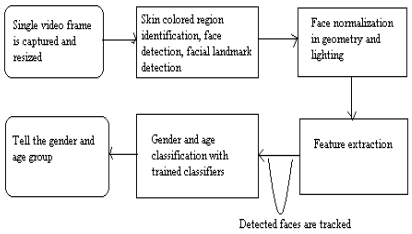


Fig. 2. Classification Mode

that human. Eyes, mouth and nose are searched on the resized face images. Cheeks and forehead positions are approximated with the available positions of eyes, mouth and nose. Eyes, mouth and nose detectors are boosted cascade classifier similar with the face detector. We used the available classifiers for eyes [8], for nose and mouth [9]. The detected eye coordinates are used to rotate the face to equalize both eyes' y-coordinates. Then histogram equalization method is executed on the face images to equalize the brightness distribution of the image. Finally, the required rigid transforms are computed on images (mapping facial landmarks to defined positions; like left eye at 25% of image width, and right eye at 75% of image width).

3 Gender Classification

First, 256 bins LBP feature vectors for all images are calculated. The calculated feature vector is processed using Random Forest [5]. In random trees there is no need for any accuracy estimation procedures, such as cross-validation or bootstrap, or a separate test set to get an estimate of the training error, and there are fewer parameters than SVM to be set. In some situations, random forest outperforms SVM. When the training set for the current tree is drawn by sampling with replacement, some vectors are left out (out-of-bag data). The classification error is estimated by using this out-of-bag data.

Our second classifier is an adaboost classifier which is trained with the data extracted from comparisons of LBP values of pixels on 20x20 pixels face image. We used ten types of pixel comparison operators: (Let L_1 is LBP value of $pixel_1$ and L_2 is LBP value of $pixel_2$) $L_1 > L_2$, $L_1 < 5 * L_2$, $L_1 < 10 * L_2$, $L_1 < 25 * L_2$, $L_1 < 50 * L_2$, $L_2 > L_1$, $L_2 < 5 * L_1$, $L_2 < 10 * L_1$, $L_2 < 25 * L_1$, $L_2 < 50 * L_1$. This algorithm is same as the algorithm in [1] except the comparison operators. Each comparison yields a binary feature. We use these binary features as weak classifiers which are only required to have accuracy slightly better than random chance. The output of the classifier is the value of the binary feature. If the value of any binary feature is 1, the output is male, otherwise female. In this method, we used 20x20 pixels face image and this yields to 10x400x399 = 1596000 distinct weak classifiers. Adaboost algorithm is used to combine these weak classifiers together. Its primary goal is to form a single strong classifier with better accuracy. The computation of the accuracy in each iteration is a time consuming process but it only effects the training time, not the classification time. Randomly selecting weak classifiers in all iterations can reduce the training time. We select the best 1000 weak classifiers to construct a strong classifier. We sorted weak classifiers by their accuracy on the training images. If a training image is a male face image and the weak classifier gives the correct output for that image, the

weak classifier's point is incremented by one. All the randomly selected weak classifiers are graded in this way and finally they are sorted by their total point. We select the best 1000 of the weak classifiers and write their pixel coordinates and their operators to a file which will be used to load to build strong classifier.

Gender classification module takes two face image sets which are prepared by the "face, facial landmark detection and face normalization module". In "Classification Mode", if the classifier prediction on face images for a human contains more "male" result, the "male" outputted for random tree classifier. The second classifier, namely "Adaboost Classifier" takes the 20x20 pixels image set as input and, calculates the 1000 selected (weak classifiers) binary operation results, and calculates the genders of the faces. Finally, if the results of these two classifications are same, the gender of the face is determined as the result of any classifier.

4 Age Classification

We divide human ages into four classes, 0–20, 20–40, 40–60 and 60–100. In order to classify the age group of faces, we use two distinct age classifiers. The first classifier is based on DCT Mod2 features and Random Forest. In this method, the set of face images are 40x40 pixel size, are given to DCT Mod2 feature extraction method. In this method the given face image is analyzed on a block by block basis [6,7]. In our implementation, the block size is 8x8 pixels size. Therefore, the feature vector of a face image has a size 1458. "Random Forest" classifier is trained with these features. Our second classifier is based on LBP features of selected face regions and "Random Forest". In this classification, LBP features of some selected regions of the face are computed first. Wrinkle structures around the eyes, cheeks and forehead have different characteristics which can differentiate by the age.

The calculation of LBP features yields to a 256 bin histogram. "Random Forest" classifier is trained with these features. In "Classification Mode" the DCT Mod2 Features Random Forest Classifier takes the 40x40 pixels image set as input and it extracts its DCT Mod2 feature vectors, then it predicts the age group of the face images separately with the trained classifier. The output of this classifier is the average of the results of all predictions on the given face image set of a person. The Region's LBP Features Random Forest Classifier takes the 80x80 pixels image set as input and it extracts its LBP feature vectors, then it predicts the age group of the face images separately with trained classifier. The output of this classifier is the average of the results of all predictions on a given face image set of a person. Finally, the final output is determined by averaging the results and taking floor of the average value.

5 Experiments and Results

We divided our experiments into two parts: *Gender Classification* and *Age Classification*. The captured face images from videos are grouped as males and females in the first experiment. Both groups contained 2500 images. The detected faces are stored with eyes, nose and mouth coordinates. Then face normalization methods are executed. Feature extraction methods are applied and feature vectors are stored in related files. Classifier methods are trained with stored values and classifier data are stored in xml files (for random forest), text file (for adaboost). In *Age Classification*, there were 750 images in all groups. Classifier methods are trained

with extracted features and classifier data are stored in xml files. Then, we tested the classifiers with 600 detected faces in testing videos. The results were satisfactory (see Tables 1 and 2). The running time of our method changes between 2 and 4 ms on Intel(R) Core(TM) 2 Duo CPU T5850 @ 2.16GHz, 2.99 GB RAM notebook.

Table 1. Real-time Gender Classification in Videos

Classifier	# of training images	Success Rate (%)
LBP and RF	5000	91.5
Pixel comp.& Adaboost	5000	92.0
Combination	5000	96.5

Table 2. Real-time Age Group Classification in Videos

Classifier	# of training images	Success Rate (%)
DCT Mod2 & RF	3000	80.75
Selected Region LBP and RF	3000	83.25
Combination	3000	87.25

6 Conclusions and Future Work

The accuracy of gender classification is found as 96.5% in our experiments. This ratio is higher than our expectations and most of the accuracy rates reported in the literature. This result shows the impressiveness of the proposed method on gender classification. "Age Group Classification" module contains two distinct age group classifiers as gender classification module. The combination of two classifiers has 87.25% accuracy rates on 600 detected faces of test videos. 87.25% is a high success rate for a process of classification of age groups on videos. The main contribution of this work is adopting LBP and DCT Mod2 used mainly on face verification and recognition to gender and age classification from faces. Future work can be done in adding these methods to an ontological semantic video annotation framework. Our methods can be used as a personal information extractor for a semantic video annotation framework.

Acknowledgments

This work is partially supported by The Scientific and Technical Council of Turkey Grant ‘‘TUBITAK EEEAG-107E234.

References

1. Baluja, S., Rowley, H., Boosting Sex Identification Performance, Intl Journal of Computer Vision, v.71 n.1, p.111-119, 2007.
2. Viola, P., Jones, M., Rapid Object Detection using a Boosted Cascade of Simple Features, CVPR 2001.
3. Yang, M.-H., et al. Detecting faces in images: a survey, IEEE Trans. Pattern Analysis and Machine Intelligence, 24(1), 2002, pp. 34-58.
4. Hjelmås, E., Low, B. K., Face detection, a survey, Computer Vision and Image Understanding, 83(3), 2001, pp. 236-274.
5. http://stat-www.berkeley.edu/users/breiman/wald2002-2.pdf
6. Sanderson, C., Paliwal,, K. K., Fast feature extraction method for robust face verification, Electronics Letters, vol. 38, no. 25, pp. 1648-1650.
7. Eickler, S., Muller, S., and Rigoll, G., Recognition of JPEG compressed face images based on statistical methods, Image Vis. Comput., 2000, 18, (4), pp. 279–287.
8. http://www-personal.umich.edu/~shameem/haarcascade_eye.xml
9. http://mozart.dis.ulpgc.es/Gias/modesto.html

Summarization of Documentaries

Kezban Demirtas[1] Ilyas Cicekli[2] Nihan Kesim Cicekli[1]

[1] Department of Computer Engineering, Middle East Technical University, Ankara, Turkey
[2] Department of Computer Engineering, Bilkent University, Ankara, Turkey
kezbandemirtas@gmail.com, ilyas@cs.bilkent.edu.tr, nihan@ceng.metu.edu.tr

Abstract. Video summarization algorithms present condensed versions of a full length video by identifying the most significant parts of the video. In this paper, we propose an automatic video summarization method using the subtitles of videos and text summarization techniques. We identify significant sentences in the subtitles of a video by using text summarization techniques and then we compose a video summary by finding the video parts corresponding to these summary sentences.

Keywords: Video Summarization, Text Summarization.

1 Introduction

Video content is being used in a wide number of domains ranging from commerce, security, education and entertainment. People want to search and find the video content according to its semantics. Creating searchable video archives becomes an important requirement for different domains as a result of the increase in the amount of multimedia contents. Video summarization helps people to decide whether they really want to watch a video or not. Video summarization algorithms present a condensed version of a full length video by identifying the most significant parts of the video.

In this paper, we propose an automatic video summarization system in order to present summaries to the users so that they can decide easily whether the selected video is of any interest to them. We aim to use text information only to determine how only the text data associated with the video is helpful in searching the semantic content of videos. The subtitles provide the speech content with the time information which is used to retrieve the relevant video pieces. For this purpose, we have chosen documentary videos as the application domain. In documentary videos, the speech usually consists of a monolog and it mentions the things seen on the screen.

For automatic summarization, we make use of two text summarization algorithms [1,3] and combine the results of these two algorithms to constitute a summary. Text summarization techniques identify the significant parts of a text to constitute a summary. We extract a summary of video subtitles with these summarization algorithms and then we find the video parts corresponding to these summary parts. By combining the video parts, we create a moving-image summary of the original video. In our summarization approach, we take the advantage of the documentary video characteristics. For example, in a documentary about “animals”, when an animal is seen on the screen, the speaker usually mentions that animal. So, when we find the video parts corresponding to the summary sentences of a video, those video parts are

closely related with the summary sentences. Hence we obtain a semantic video summary giving the important parts of a video.

Text features associated with a video can be viewable text placed on the screen or transcript of the dialog which can be provided in the form of closed captions, open captions or subtitles. Text features plays an important role in video summarization as it contains detailed information about the video content. Pickering et al. [4] make summarization of television news by using the accompanying subtitles. They extract news stories from the video and provide a summary for each story by using lexical chain analysis. Tsoneva et al. [5] creates automatic summaries for narrative videos using textual cues available in subtitles and scripts. They extract features like keywords, main characters' names and presence, and according to these features they identify the most relevant moments of video for preserving the story line. In our video summarization system, we extract moving-image summaries of documentaries using video subtitles and text summarization methods.

The rest of the paper is organized as follows. Section 2 describes our video summarization approaches and we present evaluations of these approaches in Section 3. Finally in Section 4, conclusions and possible future work are discussed.

2 Video Summarization

We find the summary sentences of the subtitle file by using the text summarization techniques [1,3]. Then we find the video segments corresponding to these summary sentences. By combining the video segments of summary sentences, we create a video summary. Subtitle files contain the text of the speech, the number and time of speech. In the text preprocessing step, the text in the subtitle file is extracted by striping the number and time of the speech, and it is given to the "Text Summarization" module. "Text Summarization" module finds the summary sentences of the given text. We use three algorithms for finding the summary sentences; TextRank algorithm [3], Lexical Chain algorithm [1] and a combination of these two algorithms. After the summary sentences are found by one of these approaches, the output can be given to the "Text Smoothing" module. This module applies some techniques to make summary sentences more understandable and smoother. "Video Summarization" module creates the video summary by using the summary sentences. This module finds the start and end times of sentences from the video subtitle file. Then the video segments corresponding to start and end times are extracted. By combining the extracted video segments, a video summary is generated.

The TextRank algorithm [3] extracts sentences for automatic summarization by identifying sentences that are more representative for the given text. To apply TextRank, we first build a graph and a vertex is added to the graph for each sentence in the text. To determine the connection between vertices, we define a "similarity" relation between them, where "similarity" is measured as a function of their content overlap. The content overlap of two sentences is computed by the number of common tokens between them. To avoid promoting long sentences, the content overlap is divided by the length of each sentence.

In [1] automated text summarization is done by identifying the significant sentences of text. The lexical cohesion structure of the text is exploited to determine the importance of sentences. Lexical chains can be used to analyze the lexical

cohesion structure in the text. In the proposed algorithm, first, the lexical chains in the text are constructed. Then topics are roughly detected from lexical chains and the text is segmented with respect to the topics. It is assumed that the first sentence of a segment is a general description of the topic, so the first sentence of the segment is selected as the summary sentence.

We also propose a new summarization approach by combining the two summarization algorithms, TextRank algorithm [3] and Lexical Chain algorithm [1]. In this approach, we find the summary sentences of a text by using both the TextRank algorithm and the Lexical Chain algorithm. Afterwards, we determine the common sentences of two summaries and select these sentences to be included in the summary. Both algorithms determine the summary sentences of a text in a sorted manner, that is, the summary sentences are sorted with respect to their importance scores. After selecting the common sentences, we select the most important sentences of the two algorithms up to the length of the desired summary.

In order to improve the understandability and completeness of the summary, some smoothing operations are done after text summarization. It is observed that some of the selected sentences start with a pronoun and if we do not have the previous sentences in the summary, these pronouns may be confusing. In order to handle this problem, if a sentence starts with a pronoun, the preceding sentence is also included in the summary. If the preceding sentence also starts with a pronoun, its preceding sentence is also added to the summary sentence list. The backward processing of the sentences goes at most two steps. We observed that if a sentence starts with a pronoun, including just the preceding sentence solves the problem in most cases and the summary becomes more understandable.

3 Experiments and Evaluation

The evaluation of video summaries is a hard job because summaries are subjective. Different people will compose different summaries for the same video. The evaluation of video summaries could be conducted by requesting people watch the summary and asking them several questions about the video. However, in our summarization system, since we use text summarization algorithms, we prefer to evaluate the text summarization algorithms only. We believe that the success of the text summarization directly determines the success of video summarization in our system. For the evaluation of text summarization, we use ROUGE (Recall-Oriented Understudy for Gisting Evaluation) algorithm [2] which makes evaluation by comparing the system generated output summaries to model summaries written by humans.

In our video summarization system, we tried six algorithms (three text summarization algorithms with or without smoothing the result) by using the documentaries from BBC. We asked students to compose summaries of the selected documentaries by selecting the most important twenty sentences from the subtitles. The same documentaries were also summarized by our video summarization system which generated summaries composed of twenty sentences by using our algorithms. In order to compare the system outputs with human summaries, the ROUGE scores are calculated, and given in Table 1. From Table 1, we can observe that smoothing improves the performance of all the algorithms. Our best method is the combination

of two algorithms using smoothing, and our best scores are comparable with the scores of the state of the art systems in the literature.

Table 1. ROUGE Scores of Algorithms in Video Summarization System

Summarization Algorithm	*ROUGE-1*	*ROUGE-L*	*ROUGE-W*
TextRank	0,33877	0,33608	0,13512
TextRank_Smooth	0,34453	0,34184	0,13686
LexicalChain	0,24835	0,24600	0,10413
LexicalChain_Smooth	0,25211	0,24976	0,10529
Mix	0,34375	0,34140	0,13934
Mix_Smooth	0,34950	0,34716	0,14108

4 Conclusions

This paper presents a system which performs automatic summarization of documentary videos with subtitles. We perform video summarization by using video subtitles and employing text summarization methods. In this work, we take the advantage of the characteristics of the documentary videos. In documentary videos, the speech and the display of the video have a strong correlation in the way that mostly both of them give information about the same entities.

In the evaluation of video summaries, we evaluate the text summaries of videos. We compare the program summaries with human generated summaries and find the ROUGE score of program summaries. As a future work, we want to perform the detailed user evaluation of video summaries. Video summaries could be watched by viewers and the viewers could evaluate the results.

Acknowledgments

This work is partially supported by The Scientific and Technical Council of Turkey Grant ‘‘TUBITAK EEEAG-107E234, and The Scientific and Technical Council of Turkey Grant “TUBITAK EEEAG-107E151”.

References

1. G. Ercan, and I. Cicekli. Lexical cohesion based topic modeling for summarization. In Proceedings of the CICLing 2008, pp. 582–592.
2. C.Y. Lin, and E.H. Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In Proceedings of HLT-NAACL-2003, Edmenton, Canada, 2003.
3. R. Mihalcea, and P. Tarau. TextRank - bringing order into texts. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2004), Barcelona, Spain.
4. M. Pickering, L. Wong, and S. Ruger. ANSES: Summarization of News Video. In Proceedings of CIVR-2003, University of Illinois, IL, USA, July 24-25, 2003.
5. T. Tsoneva , M. Barbieri, and H. Weda. Automated summarization of narrative video on a semantic level, In Proceedings of the International Conference on Semantic Computing, pp.169-176, September 17-19, 2007.

JIS

Automatic categorization and summarization of documentaries

Kezban Demirtas and Nihan Kesim Cicekli

Department of Computer Engineering, METU, Ankara, Turkey

Ilyas Cicekli

Department of Computer Engineering, Bilkent University, Ankara, Turkey

Abstract.

In this paper, we propose automatic categorization and summarization of documentaries using subtitles of videos. We propose two methods for video categorization. The first makes unsupervised categorization by applying natural language processing techniques on video subtitles and uses the WordNet lexical database and WordNet domains. The second has the same extraction steps but uses a learning module to categorize. Experiments with documentary videos give promising results in discovering the correct categories of videos. We also propose a video summarization method using the subtitles of videos and text summarization techniques. Significant sentences in the subtitles of a video are identified using these techniques and a video summary is then composed by finding the video parts corresponding to these summary sentences.

Keywords: video categorization; video summarization; text summarization; WordNet domains

1. Introduction

Video content is being used in a wide number of domains ranging from commerce to security, education and entertainment. People want to search and find this content according to its semantics. Creating searchable video archives had become an important requirement for different domains as a result of the increase in the amount of multimedia content. Narrowing down the user's search space by categorizing videos can help people to solve this problem. Since there is a huge amount of videos to categorize, automatic categorization is an important research area [1–4]. Video summarization helps people to decide whether they really want to watch a video. Summarization algorithms present condensed versions of a full length video by identifying the most significant parts. In order to have an idea about the content of a video, using such a summary is much easier than going through all of the footage.

Correspondence to: Ilyas Cicekli, Deptartment of Computer Engineering, Bilkent University, 06800 Bilkent, Ankara, Turkey. Email: ilyas@cs.bilkent.edu.tr

In video categorization, video is generally classified into one of several broad categories such as documentary type (e.g. geography, animals religion) or movie genre (e.g. action, comedy, drama, horror). In order to classify videos automatically, features are drawn from three modalities: visual, audio or text. Also some combinations of these features can be exploited together. Therefore video classification approaches could be divided into four groups: text-based approaches, audio-based approaches, visual-based approaches and those that use some combination of visual, audio and text features. Text-based approaches [2, 4–6] are the least common in the video classification literature but have several benefits over other approaches. First of all, text processing is a more lightweight process than video and audio processing. Also text categorization techniques have been studied extensively in the computational linguistics literature [1, 3]. This accumulation can be exploited in video classification domain. Beside this, the human language in a video carries more semantic information than its visual/audio features. Words have meaning to humans and some tend to be associated with certain categories. Another benefit of using text features is that, by using some lexicon, such as WordNet, concept learning can be performed.

Video summaries are either used individually or integrated into various applications, such as browsing and searching systems. There are two main trends in video summarization: still image summaries and moving image summaries. The former are based on extracting individual key frames representing the content of the video in a static way [7]. Generally video is segmented into shots and the key frames representing these shots are selected to be included in the summary. The latter are a collection of original video parts [8, 9]. These summaries can be classified into two subtypes: previews and summaries. Video previews present the most interesting parts of a video, for example a movie trailer, whereas video summaries keep the semantic meaning of the original. Since video has a multimodal nature, summarization can be performed by using the image features, audio features or text features of video. A combination of these can be exploited together.

In this paper we propose to use automatic categorization and summarization techniques in one framework to help users first find the category of the video and then present its summary so that they can decide easily whether the selected video is of any interest to them. We aim to use text information only in order to determine how the data associated with the video are helpful in searching the semantic content of videos. For this purpose, we have chosen documentary videos as the application domain as the speech usually consists of a monologue and it mentions the things seen on the screen. The subtitles provide the speech content with the time information which is used to retrieve the relevant video pieces.

Two methods for video categorization, both based on text processing, are proposed. The first, category label assignment, makes categorization by applying natural language processing techniques on video subtitles and uses the WordNet lexical database and WordNet domains. The method is based on an existing video categorization algorithm [6] and makes some extensions to this. The TextRank algorithm [10] is used for keyword selection and one third of words are selected as keywords. In our implementation, we do not use this keyword rate but instead determine the number of keywords experimentally. Additionally, our algorithm makes use of the title of a documentary video in addition to the subtitles. The title gives important clues about the type of video because generally they are selected in order to reflect the content of the documentary. For example, the category of the documentary 'War of the century' is 'War', or the category of the documentary 'Planet Earth – mountains' is 'Geography'.

The second method, categorization by learning, has the same steps for extracting WordNet domains but performs categorization by using a learning module which learns the general WordNet domain distributions of categories. When categorizing a video, its WordNet domain distribution is analysed and the most similar category is assigned.

For automatic video summarization, we make use of two text summarization algorithms [10, 11] and combine the results to constitute a summary. Text summarization techniques identify the significant parts of a text to constitute a summary. We extract a summary of video subtitles and then the corresponding video parts are found. By combining the video parts, we create a moving image summary of the original. In our summarization approach, we take the advantage of the documentary

video characteristics. For example, in a documentary about 'animals', when an animal is seen on the screen, the speaker usually mentions that animal. So, when we find the video parts corresponding to the summary sentences of a video, those video parts are closely related with the summary sentences. Hence we obtain a semantic video summary giving the important parts of a video.

The rest of the paper is organized as follows. In Section 2, we discuss the related work in video categorization and video summarization. Section 3 describes our algorithms for video categorization and presents an evaluation of the algorithms. In Section 4, we give the description of our video summarization approaches and an evaluation of these approaches. Finally in Section 5, conclusions and possible future work are discussed.

2. Related work

In this section, we discuss the related work in video categorization and video summarization. We performed the latter by utilizing text summarization techniques and therefore a summary of the literature on both is presented.

2.1. Related work in video categorization

Video categorization algorithms assign a meaningful label to a video such as 'sports video' or 'comedy video'. The required features are drawn from three modalities: visual, audio and text. So, video categorization approaches can be classified as visual-based, audio-based and text-based. Some approaches use a combination of these three features.

Since the main topic of this paper is the categorization of videos using text features, we present here the related work on video categorization based on text processing. The text associated with a video can be viewable text or a transcript of the dialogue. The former is the text placed on the screen and some optical character recognition (OCR) methods should be used in order to use this. The latter can be provided in the form of closed/open captions or subtitles. Alternatively, it can be obtained by using speech recognition methods.

Zhu et al. [4] performed automatic news video story categorization based on the closed-captioned text. They segmented news video into stories using the demarcations which indicate the topic changes in the text. Then for each story, a category is assigned by extracting a list of keywords and further processing them.

Brezeale and Cook [1] used text (closed captions) and visual features separately in video classification. To classify a movie, the closed captions are firstly extracted and stop words are subsequently removed. Each word is then stemmed by removing the suffixes to find the root. By using these stemmed words, a term feature vector is generated. Classification is performed using a support vector machine (SVM). There are 15 genres of movies from the entertainment domain and the evaluation is performed on 81 movies.

Wang et al. [12] used text features for classification purposes. News videos were assigned to one of 10 categories and the spoken text was extracted using speech recognition methods. Text derived from speech recognition, however, generally has a fairly high error rate.

Qi et al. [13] classified a news video into types of news stories. First, the shots and, if necessary, scenes of video were detected using audio and visual features. The closed captions and scene text detected by the OCR methods were then used for classification.

Katsiouli et al. [6] used subtitles for documentary classification. They performed categorization by using the WordNet lexical database and WordNet Domains [14] and applied natural language processing techniques on subtitles. They predefined documentary categories as geography, history, animals, politics, religion, sports, music, accidents, art, science, transportation, technology, people and war. Their categorization approach has achieved 69.4% accuracy. In this paper, a similar approach is followed with a different categorization algorithm and better results are obtained.

2.2. Related work in video summarization

In the literature, there are several approaches using the image, audio or text features in video summarization. Also some approaches use a combination of these features [1, 15]. Image features include changes in colour, texture, shape and motion of objects generated by the image stream of the video. By using these features, the shots of a video can be identified, such as cuts or fades. Cuts are represented by sharp changes while fades are identified by slower changes in image features. For instance, Ekin et al. [16] observed that the important scenes of a football game conform to long, medium and close-up view shots and these are then used in their summarization system.

In addition to shots, specific objects and events can be identified and this information could improve summarization performance. Knowledge of content domain could be helpful in the identification of objects within the video (e.g. anchor person) and events (e.g. the news headlines). The techniques presented in [14, 17] analyse image features from the video stream, and are domain specific. The systems in [16, 18–20] use image features to identify representative key frames for inclusion in the video summary and all are non-domain specific.

Audio features associated with a video include speech, music, sounds and silence. These are used to select candidate segments to be included in a video summary and domain-specific knowledge can be used to enhance the summary success. For example, excited commentator speech and excited audience sounds may show a number of potential events such as the start of a free kick, penalty kick, foul or goal [5]. Rui et al. [21] analysed the speech track to find exciting segments and events, such as baseball hits in baseball videos.

Text features play an important role in video summarization as they contain detailed information about the content. Pickering et al. [22] used accompanying subtitles to summarize television news. They extracted news stories and provided a summary for each one by using lexical chain analysis. Tsoneva et al. [23] created automatic summaries for narrative videos using textual cues available in subtitles and scripts. They extracted features like keywords, main characters' names and presence, and according to these features they identified the most relevant moments of video for preserving the storyline. In our video summarization system, we extracted moving image summaries of documentaries using video subtitles and text summarization methods.

2.3. Related work in text summarization

Text summarization techniques can be useful in video summarization since some videos have text related to the content and the summary is therefore an important resource. Text summarization techniques investigate different clues that could be used to identify important topics and ideas of the text. The summarization methods can be classified by the clues that they use in summarization. Summaries can be created by selecting the first sentences of text and this simple technique gives very good results in news articles and scientific reports [24].

In text, to emphasize the importance of a sentence some phrases are used such as 'significantly' and 'in conclusion' – these phrases are called 'bonus phrases'. On the other hand, some phrases reflect the unimportance of a sentence such as 'hardly' and 'impossible' – these phrases are called 'stigma phrases'. In addition to cue phrases, some formatting features like bold words and headers could enhance the summarization performance. The systems in [2, 25] make use of both in their summarization systems.

Weighted vectors of TF*IDF (Term Frequency * Inverse Document Frequency) values can be used to represent sentences. The TF*IDF value takes advantage of word repetition in the text which is a lexical cohesion type. Radev et al. [26] used such weighted vectors to find the important sentences in a summarization task. The summarization system in [27] uses an algorithm which is similar to Google's Pagerank [28] in order to select the summary sentences. Mihalcea and Tarau [10] proposed a summarization algorithm named TextRank which also relies on the Pagerank algorithm and uses the word repetition feature.

Lexical chains, which are sets of related words, can also be used for modelling lexical cohesion. Barzilay and Elhadad [29] used lexical chains to extract summaries and achieved good results. Many lexical cohesion-based algorithms [11, 30–32] are developed following the Barzilay/Elhadad

algorithm. Silber and McCoy [32] proposed a summarizer based on lexical chains and tried to improve the running time of the lexical chaining algorithm. Chali and Kolla [33] used lexical chains and offered a different sentence selection approach. In Ercan and Cicekli [11] the lexical cohesion structure of the text is exploited to determine the importance of sentences. Their summarization algorithm constructs the lexical chains of a text and identifies topics from them. The text is segmented with respect to these topics and the most important sentences are selected from these segments.

3. Automatic video categorization

Two algorithms for video categorization are proposed: category label assignment and categorization by learning. The first is based on the algorithm presented in [6]. The algorithm is extended by adding video name processing and changing the way the number of keywords in subtitle processing is determined. With these extensions, better results are obtained. According to this algorithm, a video is assigned a category label using the WordNet lexical database and WordNet domains [34] and applying natural language processing techniques on subtitles. In the second algorithm, categorization is done by learning. A learning module is implemented, which can be trained by using the videos of known categories. The algorithm starts with the preprocessing steps of the first algorithm and the categorization is performed by the learning module.

The common preprocessing steps of the two algorithms are given in Section 3.1. The first video categorization algorithm is given in Section 3.2, the second video categorization algorithm is given in Section 3.3, and the evaluation of these algorithms is given in Section 3.4.

3.1. Extracting WordNet domains

Initially, WordNet domains of a video are extracted and are used in both of the proposed categorization algorithms. The overview of extracting WordNet domains is given in Figure 1. The method for extracting WordNet domains starts with 'text preprocessing'. In this step, the sentences in the subtitle file are split, the words in every sentence are tagged with part of speech (POS) tags and the stop words are removed. The processed text is given to a 'keywords extraction' module which finds the keywords of the given text. Since these may carry more than one meaning, the 'word sense disambiguation' module finds the correct sense by using an adaptation of the Lesk algorithm [35]. Then the 'WordNet domains extraction' module finds the WordNet domains of the keywords corresponding to their correct senses. This module uses WordNet domains and considers the effect of the video title on categorization. Since titles give important clues about the category, this information is taken into consideration. Hence we obtain the WordNet domains of the video.

In the text preprocessing step, a subtitle is processed to find its sentences and the types of the words in these sentences are determined. A sample subtitle file is shown in Figure 2. After the sentences are extracted, a POS tagger is applied to the words, which determines the word class of each one in the sentence. The Stanford Log-linear Part-Of-Speech Tagger [36] was used for this purpose. The assigned part of speech tags consist of coded abbreviations conforming to the scheme of the Penn Treebank [37], the linguistic corpus developed by the University of Pennsylvania. For example, 'JJ' means 'Adjective', 'NNS' means 'Noun, plural' and NN means 'Noun, singular or mass'. After POS tagging, stop words (ones that do not contribute to the meaning of the sentence, such as 'above', 'the' and 'her') are removed from the sentences since these carry no semantics.

In order to select the most important words in the subtitle file for classifying the video, a keyword selection algorithm, namely the TextRank [10], is used. This algorithm builds a graph representing the text and applies a ranking algorithm to the vertices of the graph. The words are added to the graph as vertices for keywords extraction. Two vertices are connected if they have a co-occurrence relation. Two vertices co-occur if they are within a window of maximum N words, where N can be set to a value from 2 to 10. In our implementation N is set to 2. After building the graph, a graph-based ranking algorithm, derived from the PageRank algorithm [28], is used in order to decide the importance of a vertex. The basic idea of the algorithm is 'voting': when a vertex links to another

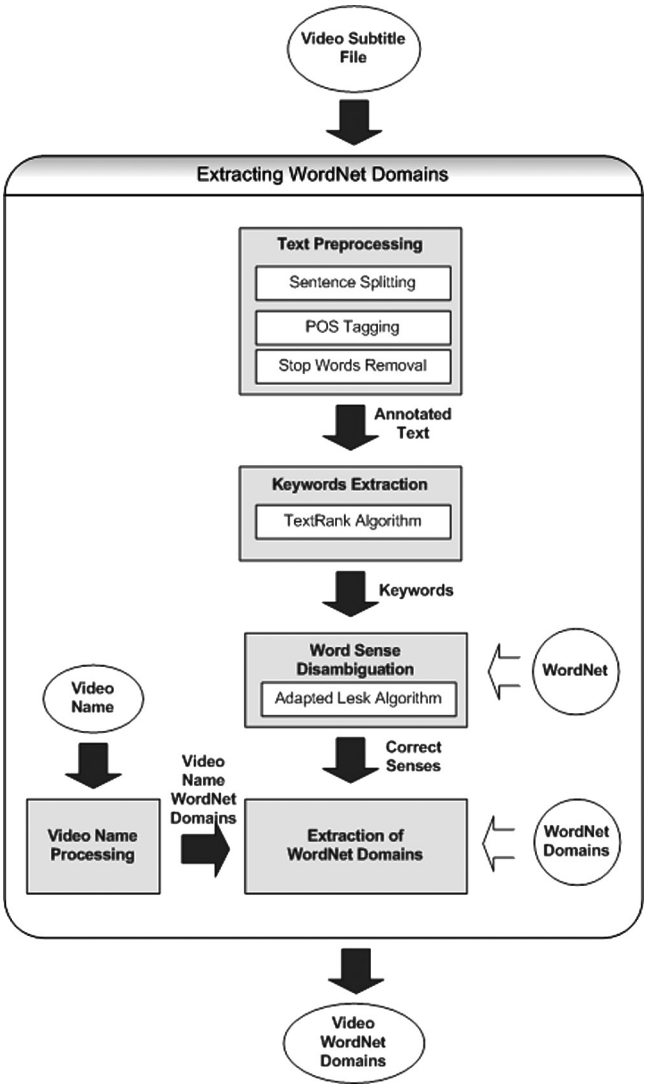


Fig. 1. Extracting WordNet domains.

one, it casts a vote for that vertex. Also, the importance of the vertex casting the vote determines the importance of the vote. Hence, the score of a vertex is computed by the votes that are cast for it and the score of the vertices casting these votes. Once the score of each vertex is computed, the vertices are sorted based on their scores and the top T vertices are selected as keywords. Generally, T is set to a third of the number of vertices in the graph. In our implementation, the number of the vertices selected as keywords is determined experimentally. Figure 3 gives a part of a subtitle file and the extracted keywords by the TextRank algorithm.

Word sense disambiguation (WSD) is the task of determining the correct sense of a word in a text. In order to find the correct senses of the keywords, we applied a WSD algorithm, which is presented in [35]. This algorithm is an adaptation of Lesk's dictionary-based algorithm. The adapted algorithm uses WordNet to include the glosses of the words that are related to the word being disambiguated through semantic relations, such as hypernym, hyponym, holonym, meronym, troponym, and attribute of each word. This supplies a richer source of information and increases disambiguation accuracy. The adapted Lesk algorithm compares glosses between each pair of words in the window of context. These glosses are the ones associated with the synset, hypernym, hyponym, holonym,

Part of a subtitle file	*Extracted and tagged sentences*
1 00:00:25,600 –> 00:00:31,080 Human beings venture into the highest parts of our planet at their peril. 2 00:00:31,640 –> 00:00:34,480 Some might think that by climbing a great mountain 3 00:00:34,560 –> 00:00:36,320 they have somehow conquered it, 4 00:00:36,720 –> 00:00:39,800 but we can only be visitors here. 5 00:00:42,160 –> 00:00:46,240 This is a frozen alien world.	Human/JJ beings/NNS venture/NN into/IN the/DT highest/JJS parts/NNS of/IN our/PRP$ planet/NN at/IN their/PRP$ peril/NN. Some/DT might/MD think/VB that/IN by/IN climbing/VBG a/DT great/JJ mountain/NN they/PRP have/VBP somehow/RB conquered/VBN it/PRP, but/CC we/PRP can/MD only/RB be/VB visitors/NNS here/RB. This/DT is/VBZ a/DT frozen/JJ alien/JJ world/NN.

Fig. 2. Part of a subtitle file and its extracted sentences.

Part of a subtitle file	*Keywords assigned by TextRank*
Most of us would agree that a tiger is one of the world's most beautiful creatures. Sadly, in the wild, it's threatened with extinction. But, fortunately, it breeds very well in captivity, as this little cub proves. But is a tiger in a cage truly a tiger? I doubt it. To see the true essence and beauty of a tiger, you have to see it in the wild. This is the story of a tigress in the heart of India. Our tigress lives in Kanha National Park.	tiger, tigress, wild, national, kanha, captivity, breeds, lives, little

Fig. 3. Keywords of part of a subtitle file.

meronym, troponym, and attribute of each word. For example, the gloss of a synset of one word can be compared with the gloss of a hypernym of the other.

In our video categorization algorithm, WSD is essential for finding the WordNet domains of the words. Since we try to find the WordNet domains of keywords in the next step, we need to find the correct senses of these words. In our implementation, the correct sense of the keywords is assigned by using the adapted Lesk algorithm. For the keywords in Figure 3, the senses assigned by the adapted Lesk algorithm are given in Table 1.

By augmenting WordNet with domain labels, WordNet domains were created [34]. The synsets in WordNet have been annotated with at least one domain label by using a set of about 200 labels hierarchically organized. If there is no appropriate domain label for a synset, the label 'factotum' was assigned to it.

In the last step, the WordNet domains of the keywords are found. In finding the domains of a word, we should know the synset (gloss) of that word. Since we found the synsets of keywords in the WSD step, we made use of this information in finding the WordNet domains. The WordNet domains of the words are given in the last column of Table 1. Then, we calculated the occurrence score of each domain label (i.e. how many times a domain label appears in the keywords' domains) in a subtitle file and sorted them in a descending order.

We observed that video titles give important clues about categories of documentaries. For example, the category of the documentary 'Art of Spain' is 'Art'. As an extension to the approach of Katsiouli et al. [6], we decided to make use of the video title when categorizing the video which

Table 1
Senses of the keywords in Figure 3

Word	Pos Tag	Sense	Synset	WordNet Domains
tiger	Noun	1	tiger, Panthera tigris	animals, biology
tigress	Noun	0	tigress	animals
wild	Adjective	1	wild, untamed	factotum
national	Noun	0	national, subject	politics
kanha	Noun	–1	Not Found In WordNet Dictionary	—
captivity	Noun	0	captivity, imprisonment, incarceration, immurement	factotum
breeds	Verb	3	breed, multiply	factotum
lives	Verb	0	dwell, shack, reside, live, inhabit, people, populate, domicile, domiciliate	town_planning
little	Adjective	3	little, small	factotum

increased the performance of our algorithms. For this purpose, the WordNet domains are found for each word in the video title. Hence a list of WordNet domains which describes the video title is acquired. For example, the WordNet domains of the video title, 'Wildlife specials – tiger', are 'animals', 'biology' and 'factotum'.

Previously, we obtained WordNet domains of the video keywords and the occurrence scores of these domains. If one of these also exists in the video title domains, the occurrence score is increased by the ratio of one fourth. This ratio is determined experimentally. At the end of this step, we obtained the WordNet domains of a video with their occurrence scores.

3.2. Category label assignment method

Our first video categorization algorithm is category label assignment which uses mappings between categories and WordNet domains. In this algorithm, we took the approach of Katsiouli et al. [6] as a basis – some enhancements in implementing the steps were made and better results were obtained. In this video categorization algorithm, we find the WordNet domains related to the categories and a category label is assigned to the video by comparing the two. The overview of the algorithm is given in Figure 4.

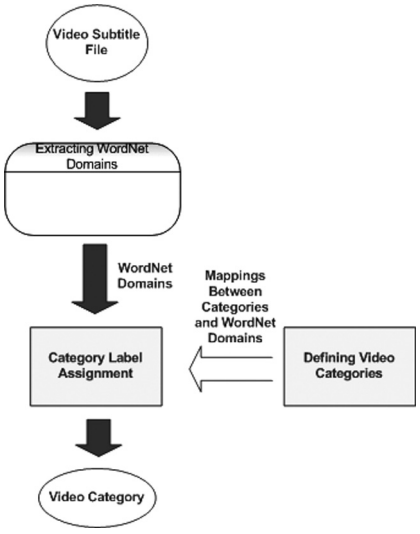


Fig. 4. Category label assignment.

Table 2
Category labels in the documentary collection and their corresponding WordNet domains

Category	Top rank WordNet domains
Geography	geography
Animals	animals, biology, entomology
Politics	politics, psychology
History	history, time_period
Religion	religion
Transportation	transport, commerce, enterprise
Accidents	transport, nautical
Sports	sport, play, swimming
War	military, history
Science	medicine, biology, mathematics
Music	music, linguistics, literature
Art	art, painting, graphic_arts
Technology	engineering, industry, computer_science
People	sociology, person

In order to assign a category label to a documentary video, a mapping is defined between the category labels and WordNet domains. First, the senses related to each category label were acquired from WordNet. The senses related with the category label through hypernym and hyponym relations and the WordNet domains corresponding to the senses of each category label were obtained. For each category, the occurrence scores of the derived domains were calculated and sorted in decreasing occurrence order. Table 2 shows the category labels and corresponding top ranked WordNet domains determined by Katsiouli et al. [6].

In the category label assignment step, a category label is assigned to the video. For a category label to be assigned, the sorted WordNet domains of the video were compared with the top rank domains of the categories. The algorithm compared the first domain of the video with the first domains of the categories.

- if the first domain of a category is equal to the first domain of the video, this category label is assigned to the video;
- if the first domain of more than one category is equal to the first domain of the video, the second domain of the corresponding sets are compared, and so on;
- if none of the category's first domains is equal to the first domain of the video, then the second domain of the video is compared to the first domain of the categories.

The algorithm continues as described above until a category label is assigned to the video. For example, when we consider the top rank WordNet domains for the categories in Table 3, if the sorted WordNet domains of a video are:

- 'animals, entomology, biology', then it is assigned to the 'Animals' category;
- 'transport, nautical, geography', then it is assigned to the 'Accidents' category;
- 'geography, animals', then it is assigned to the 'Animals' category.

At the text preprocessing step, while Katsiouli et al. [6] used Mark Hepple's POS tagger [38], we used the Stanford Log-linear Part-Of-Speech Tagger [36]. In the keyword extraction phase, they used one third of the number of words as the keyword count. In our system, we determined this number experimentally and observed that changing the number of keywords affected the system's classification accuracy (CA). Also in our implementation, we considered the effect of the video title since video titles give strong clues about the categories of documentary videos.

We implemented the approach of Katsiouli et al. [6] and evaluated with 130 documentary subtitles from National Geographic and the BBC. In this situation, we get 60% CA; after making changes to the algorithm this improved to 73.1% accuracy on the same experiment set.

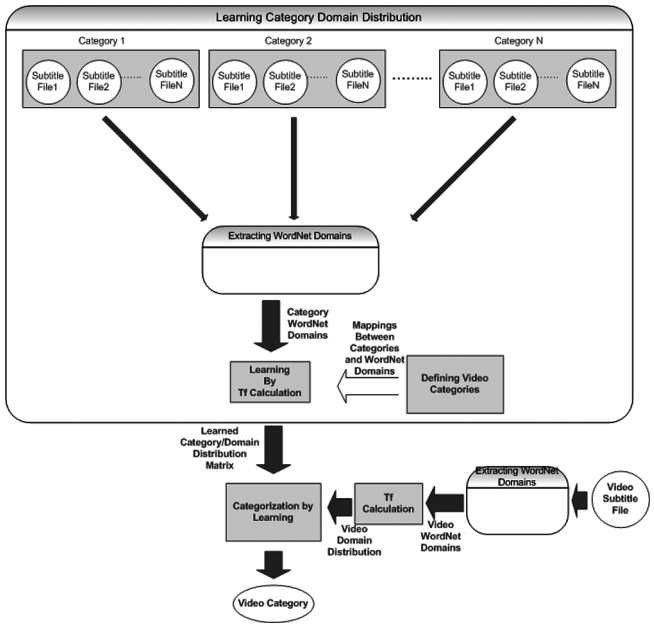


Fig. 5. Categorization by learning.

3.3. Categorization by learning method

Our second video categorization algorithm named as categorization by learning uses a learned category/domain distribution matrix. We propose a learning mechanism to assign a category label to videos. The preprocessing steps of the algorithm are the same as those used in the category label assignment method. This algorithm includes a learning phase. When a video is to be categorized, the domain distribution of the video is compared with the learned domain distribution of categories and the most similar category is assigned. The overview of the algorithm is given in Figure 5.

In the learning category domain distribution phase, documentaries with known categories are used as a training set. Our training set contains documentaries from all category labels. First of all, the documentary subtitles belonging to a specific category are processed using the extracting WordNet domains module. Hence, the domains and domain occurrence scores of the category are collected.

In order to determine the domain distribution of the category, we have used the term frequency (TF) weight of domains to determine the domain distribution of categories. The TF weight is computed for each category and domain pair. Hence a matrix showing the domain TF weights of all categories is obtained. Table 3 shows a sample part of the computed matrix representing the TF values of category domain pairs.

When we categorize a video, we compare the video's domain distribution with the domain distribution of categories and the category which has the most similar domain distribution with the video is selected. The subtitle of the video is processed in order to obtain the WordNet domains and the domain occurrence scores (TF values of the domains) of the video. Using the learned category/domain matrix, we try to find the category which has the most similar domain distribution to the video. For this purpose, we used the cosine similarity which is a measure of similarity between two vectors. For example, in order to categorize a documentary video named 'Everest', first we computed the domain distribution of the video. Table 4 shows the domain distribution of the documentary 'Everest'. Then, by using the learned matrix, we computed the similarities of categories. Table 5 shows the cosine similarities between the documentary 'Everest' and the categories. Since the 'Geography' category is most similar to the 'Everest' documentary, it is assigned as the documentary category.

Table 3
Sample part of the matrix representing the domain distribution of categories

	Geography	Animals	Politics
geography	0.0552444	0.032574	0.039201
animals	0.0302953	0.053447	0.009224
biology	0.0315682	0.041429	0.016141
entomology	0.0022912	0.000949	0
politics	0.0068737	0.008223	0.037663
psychology	0.0043279	0.007906	0.008455
history	0.0099287	0.008223	0.01691
time_period	0.0313136	0.023087	0.017294
religion	0.0089104	0.004744	0.021522
transport	0.0129837	0.012334	0.013451

Table 4
Domain distribution of the documentary 'Everest'

Domain	TF value
geography	0.036053131
animals	0.024667932
biology	0.032258065
entomology	0
politics	0.009487666
psychology	0.004743833
history	0.006641366
time_period	0.030360531
religion	0.011385199
transport	0.018975332
commerce	0.0028463
enterprise	0
nautical	0.003795066
sport	0.010436433
play	0.0056926
swimming	0
military	0.008538899
medicine	0.012333966
mathematics	0.0028463
music	0.0056926
linguistics	0.004743833
literature	0.004743833
art	0.003795066
painting	0
graphic_arts	0
engineering	0.000948767
industry	0

3.4. Experiments and evaluation

In order to evaluate the effectiveness of our categorization algorithms, we used documentaries from the BBC and National Geographic. The evaluation was performed using the CA metric, which reflects the proportion of the programme's correct assignments that agree with the original assignment.

For our first categorization algorithm (category label assignment), we conducted several experiments by changing some of the parameters. First of all, for keyword extraction, we changed the number of keywords selected and observed the results. As stated above, although Katsiouli et al. [6] selected a third of the words as keywords, we observed that this does not produce the best results.

Table 5
Cosine similarities between the documentary 'Everest' and the categories

Category	Cosine similarity
Geography	0.9590928
History	0.9452289
People	0.9376402
Animals	0.9267696
Science	0.9037822
Music	0.871545
Religion	0.825404
Politics	0.8154419
War	0.8115139
Art	0.7872766

Using the TextRank algorithm [10] all words are assigned a weight and selecting words above a certain weight could be an alternative for determining the number of keywords. Therefore, words above a certain weight are selected as keywords and the CA of the system is computed for changing weights. The diagram in Figure 6 shows the CA with changing keyword weights. For example, if we select the words with weight bigger than '5' as keywords, we get '50%' CA. As seen from Figure 6, we get the best results when using the weights between 0 and 0.4. Therefore using any weight between 0 and 0.4 does not change the CA, but selecting higher weights decreases the number of keywords. Using fewer keywords decreases the computation time. Therefore the upper bound value 'weight > 0.4' could be preferred to the others. So we used the experimentally determined value 'weight > 0.4'as the keyword selection parameter in our video categorization algorithm.

The effect of the video title when categorizing was subsequently considered. In the algorithm, we extracted the WordNet domains of a video and the occurrence scores of these domains. If one of these domains also existed in the video title domains, the occurrence score of increased by some ratio. The diagram in Figure 7 shows the effect of this ratio on the performance of the video categorization system. When the occurrence score of domains which also exist in the video title domains was increased by the ratio of 'one third' or 'one fourth', a CA of 75% was obtained. Selecting 'one third' or 'one fourth' does not make a significant difference in computation time, so any of them could be used in the video categorization algorithm – 'one fourth' was selected in our implementation.

In extracting the WordNet domains part of the categorization by learning algorithm, we used the parameters which obtained the best results in the first categorization experiment. Namely, keywords with 'weight > 0.4' in the TextRank algorithm and title effect by the ratio of 'one fourth'. To evaluate this algorithm, 65 documentaries were studied and categorized for learning purposes. An accuracy of 77% was achieved. It was noted that the performance of the system increases if the dataset used

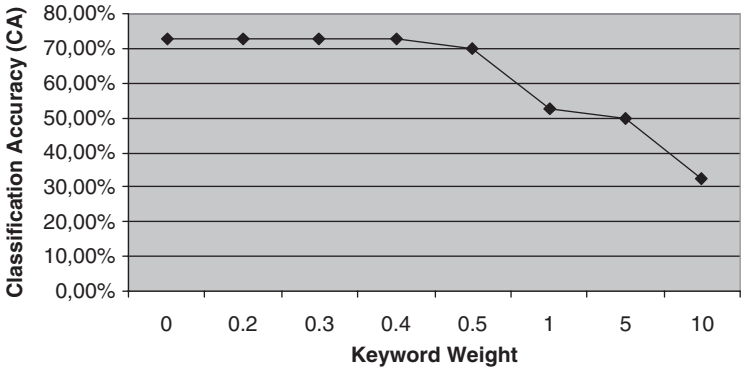


Fig. 6. Classification accuracy with keyword weights.

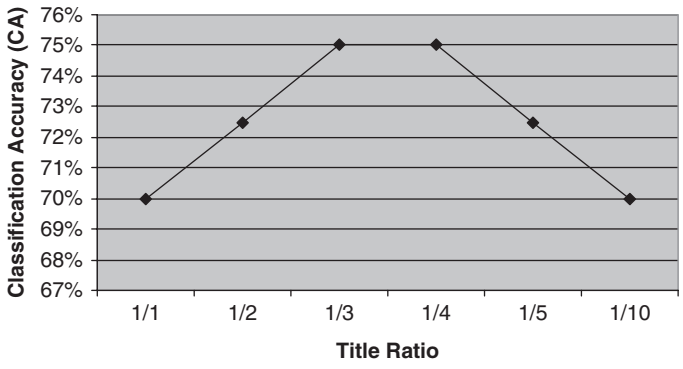


Fig. 7. Classification accuracy with title ratio.

for learning is enlarged. Also in our implementation, if more documentaries for learning could be used, better results would be maintained.

In order to see the performance of some of the well-known categorization algorithms on our subtitle dataset, we applied *K*-nearest neighbour (KNN) and SVM with polynomial kernel. We used half of the documents for training and half for the test set. KNN achieved an accuracy of 63%, and SVM achieved an accuracy of 70%. These results indicated that the usage of WordNet domains in our two categorization algorithms helps to increase the accuracy.

4. Video summarization

Video summarization algorithms present users with a condensed version of a video. In this paper, we used the subtitles of documentary videos to make summarizations. The summary sentences of the subtitle file were found by using text summarization techniques [10, 11], while the video segments corresponding to these summary sentences were extracted. By combining the video segments of summary sentences, we created a video summary. The overall approach is shown in Figure 8.

Subtitle files contain the text of the speech, the number and time of speech. In the text preprocessing step, the text in the subtitle file is extracted by striping the number and time of the speech, before being handed to the text summarization module. This module finds the summary sentences of the given text. Three algorithms were used to find the summary sentences: TextRank [10], Lexical Chain [11] and a combination of these two algorithms. After the summary sentences were found by one of these approaches, the text smoothing module applied some techniques to make summary sentences more understandable and smoother. The video summarization module used the summary sentences to create the video summary. The module found the start and end times of sentences from the video subtitle file. Then the video segments corresponding to the start and end times were subsequently extracted. By combining the extracted video segments, a video summary was generated.

4.1. Text summarization with the TextRank algorithm

The TextRank algorithm [10] extracts sentences for automatic summarization by identifying sentences that are more representative for the given text. To apply TextRank, we first built a graph and added a vertex to this graph for each sentence in the text. To determine the connection between vertices, we defined a 'similarity' relation between them, where 'similarity' is measured as a function of their content overlap. This relation can be thought of as a 'recommendation': a sentence mentioning certain concepts 'recommends' other sentences in the text that mention the same concepts and a connection is made The content overlap of two sentences is computed by the number of common tokens between them. To avoid promoting long sentences, the content overlap is divided by the length of each sentence.

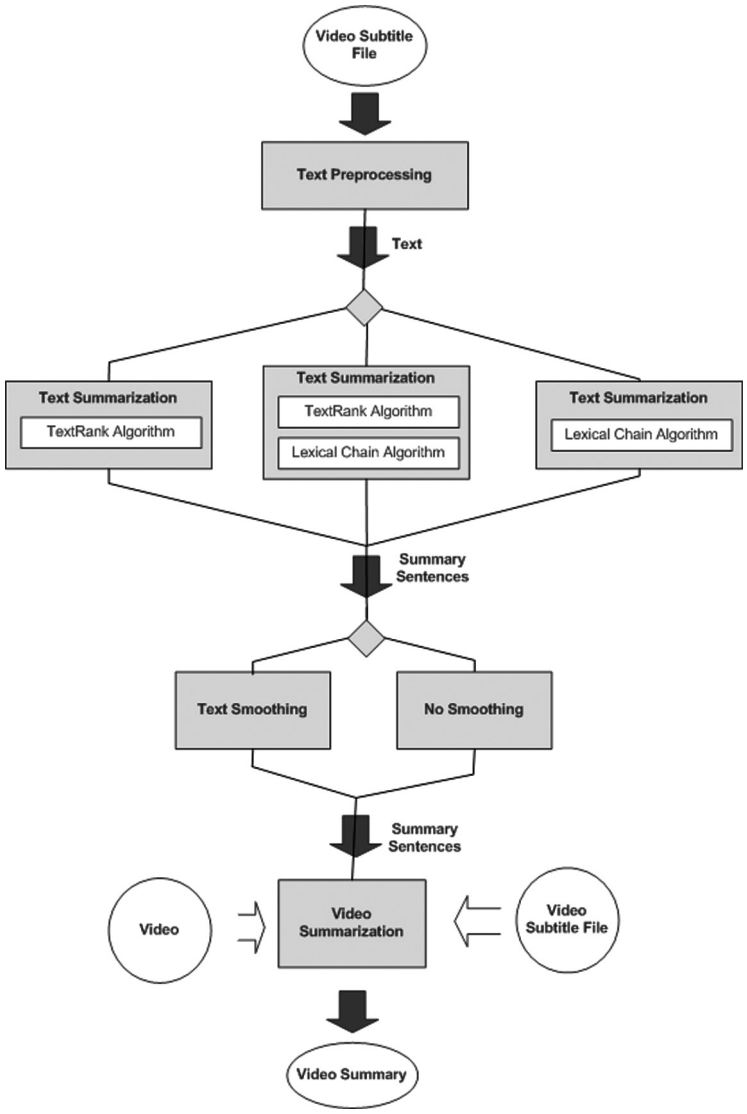


Fig. 8. Overall approach for video summarization.

A sentence composed of 'n' words is represented by $S_i = w_1,w_2,\ldots,w_n$, and two sentences S_i and S_j are given. Then the similarity of these sentences is defined formally as:

$$Similarity(S_i,S_j)=\frac{|\{w_k \mid w_k \in S_i \,\&\, w_k \in S_j\}|}{\log(|S_i|)+\log(|S_j|)}$$

The resulting graph is weighted since the edges have a similar weight. A weighted graph-based ranking algorithm is then used for deciding the importance of a vertex. Formally, the weighted score of a vertex V_i is defined as:

$$WS(V_i)=(1-d)+d*\sum_{v_j \in In(v_i)} \frac{w_{ji}}{\sum_{v_k \in Out(V_j)} w_{jk}} WS(V_j)$$

Here, $In(V_i)$ is the set of vertices that point to V_i and $Out(V_i)$ is the set of vertices that V_i points to. The weight of the edge between the vertices V_i and V_j is w_{ji}, and d is the damping factor that can be set between 0 and 1. The value of d is usually set to 0.85 and this value is also used in our implementation. After the ranking algorithm, sentences are sorted using their score and top ranked sentences are selected as the summary sentences.

4.2. Text summarization with the lexical chain algorithm

In [11], automated text summarization is done by identifying the significant sentences of text. The lexical cohesion structure of the text is exploited to determine the importance of sentences. Lexical chains can be used to analyse the lexical cohesion structure in the text. In the proposed algorithm, first, the lexical chains in the text are constructed. The lexical chaining algorithm is an implementation of Galley et al.'s algorithm [39] with some small changes. Topics are then roughly detected from lexical chains and the text is segmented with respect to the topics. It is assumed that the first sentence of a segment is a general description of the topic, so the first sentence of the segment is selected as the summary sentence.

4.3. Text summarization with a combination of algorithms

We proposed a new summarization approach by combining the two summarization algorithms, the TextRank algorithm [10] and the lexical chain algorithm [11]. Once the summary sentences of a text were found we determined the common sentences of the two summaries and selected these to be included in the main summary. Both algorithms sorted the summary sentences with respect to their importance. After selecting the common sentences, the most important sentences of the two algorithms up to the length of the desired summary were extracted. An overview of the summarization, with the combination of the algorithms, is given in Figure 9.

4.4. Text smoothing

In order to improve the understandability and completeness of the summary, some smoothing operations were carried out after the text summarization phase. It is observed that some of the selected

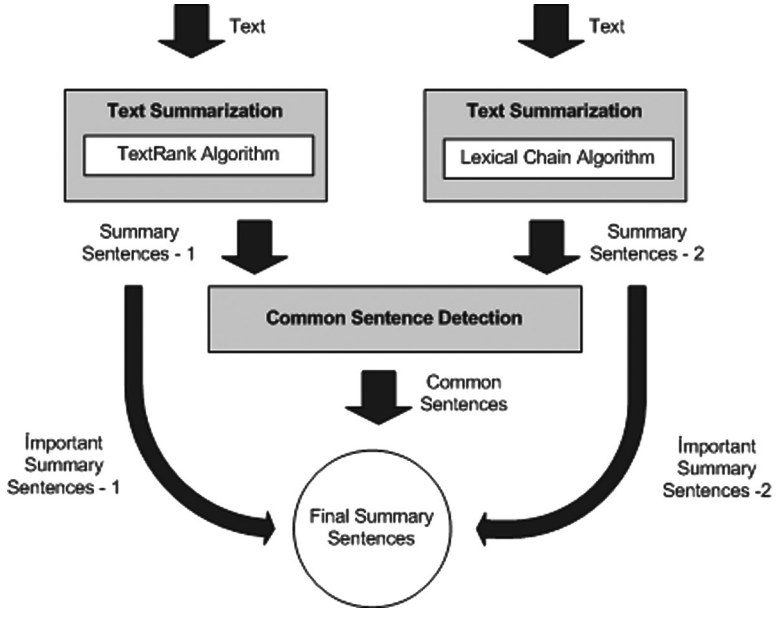


Fig. 9. Overview of the text summarization by combination of algorithms.

sentences start with a pronoun and if they are not included in the previous sentences in the summary these pronouns may be confusing.

In order to handle this problem, if a sentence starts with a pronoun, the preceding sentence is also included in the summary. If the preceding sentence also starts with a pronoun, its preceding sentence is also added to the summary sentence list. The backward processing of the sentences contains only two steps. We observed that, if a sentence starts with a pronoun, including just the preceding sentence solves the problem in most cases and the summary becomes more understandable.

4.5. Clip generation

Our video summarization approach is based on the summary sentences found by the text summarization algorithms. After finding the summary sentences, the start and end times of these sentences are found from video subtitle file. For each summary sentence, the video segment corresponding to the sentence is extracted from the video by using the start and end time of the sentence. Then, by combining the extracted video parts, a video summary is created. A screenshot of our video summarization system is presented in Figure 10. The system lets the user select the summarization algorithm from the summary method group box. The user can select the algorithms 'TextRank', 'LexicalChain' or 'Mixed'. The user can also select the options 'Normal' or 'Smooth'. The former indicates that the summarization system will not use text smoothing after text summarization.

4.6. Experiments and evaluation

The evaluation of video summaries is difficult because they are so subjective. Different people will compose different summaries for the same video. The evaluation of video summaries could be conducted by requesting people watch the summary and asking them several questions about the video. However, in our summarization system, since we used text summarization algorithms, we prefered to evaluate the text summarization algorithms only. We believe that the success of the text summarization directly determines the success of video summarization in our system. For the evaluation of text summarization, we use ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [40], which makes evaluation by comparing the system generated output summaries to model summaries written by humans.

Fig. 10. Video summarization system screenshot.

Table 6
ROUGE scores of algorithms in a video summarization system

	ROUGE-1	ROUGE-L	ROUGE-W
TextRank	0,33877	0,33608	0,13512
TextRank_Smooth	0,34453	0,34184	0,13686
LexicalChain	0,24835	0,24600	0,10413
LexicalChain_Smooth	0,25211	0,24976	0,10529
Mix	0,34375	0,34140	0,13934
Mix_Smooth	0,34950	0,34716	0,14108

ROUGE is the most popular summarization evaluation methodology. All of the ROUGE metrics aim to find the percentage of overlap between the system output and the model summaries. ROUGE calculates the ROUGE-N score (calculated using N-grams), ROUGE-L score (calculated using longest common subsequences) and ROUGE-W score (calculated using weighted longest common subsequences).

In our video summarization system, we used six approaches for finding the summary of the subtitle text of a video:

- the TextRank algorithm;
- the TextRank algorithm and smoothing the result;
- the LexicalChain algorithm;
- the LexicalChain algorithm and smoothing the result;
- a mix of the TextRank and LexicalChain algorithms;
- a mix of the TextRank and LexicalChain algorithms and smoothing the result.

All six approaches were studied using the BBC documentaries. Students were asked to compose summaries of the selected documentaries by selecting the 20 most important sentences from the subtitles. The same documentaries were also summarized by our video summarization system, using the algorithms mentioned above, which generated summaries composed of 20 sentences. We calculated ROUGE scores in order to compare the system outputs with human summaries. While calculating ROUGE scores, we applied Porter's Stemmer and stop word list on the input. ROUGE scores of the algorithms in our video summarization system can be seen in Table 6. We observed that smoothing improves the performance of all algorithms. When the TextRank algorithm is used, better results were obtained than when the LexicalChain algorithm was implemented. The best results were obtained by using the mixed TextRank and LexicalChain algorithms to find the summary sentences and smoothing the results. Our best ROUGE scores were comparable with the ROUGE scores of the state of the art systems in the literature.

5. Conclusions

This paper presented a system which performs automatic categorization and summarization of documentary videos with subtitles. We wanted to handle these problems together because their outputs support each other. Presenting both the category and the semantic summary of a video would give viewers quick and satisfactory information about that content.

The automatic video categorization was performed by two categorization methods, category label assignment and categorization by learning. The CA of the former was evaluated on documentary videos and promising results were obtained. The second method used a limited number of videos for learning. In future work, we want to improve this by using more videos. It is known that using more data for learning increases the performance of the system and gives better results.

We performed video summarization by using video subtitles and employing text summarization methods. Two text summarization algorithms [10, 27] were used and their results were applied to

the video summarization domain. In this work, we took advantage of the characteristics of the documentary videos where the speech and display of the video have a strong correlation.

Video summary is produced by extracting the video parts corresponding to the summary sentences. This extraction could be improved by employing a shot identification mechanism. An extracted video part could be extended by finding the start and end of the residing shot. In this way, the video parts could show a more complete presentation. In the evaluation of video summaries, the programme summaries were compared with human generated summaries and the ROUGE score recorded. In future work, we want to perform the evaluation by using the video summaries alongside the text summaries. Video summaries could be watched by viewers who could then evaluate the results.

Both algorithms are currently used in English, but it is possible to convert them into different languages. The language dependency of the algorithms is caused by the WordNet and the natural language processing (NLP) tools such as POS tagger. If the WordNet and the required NLP tools are available for other languages, our video categorization and summarization algorithms can be used for videos with other language subtitles.

Acknowledgements

This work is partially supported by the Scientific and Technical Council of Turkey Grant TUBITAK EEEAG-107E234, and the Scientific and Technical Council of Turkey Grant TUBITAK EEEAG-107E151.

References

[1] D. Brezeale, D.J. Cook, Automatic video classification: a survey of the literature, *IEEE Transactions Systems, Man and Cybernetics Part C: Applications and Reviews* 38(3) (2008) 416–430.

[2] S. Teufel and M. Moens, Sentence extraction as a classification task, *Proceedings of ACL/EACL 97 WS* (Madrid, Spain, 1997).

[3] X. Yuan, W. Lai, T. Mei, X.S. Hua, X.Q. Wu and S. Li, Automatic video genre categorization using hierarchical SVM, *Proceedings of IEEE International Conference on Image Processing* (2006) 2905–2908.

[4] W. Zhu, C. Toklu and S.P. Liou, Automatic news video segmentation and categorization based on closed-captioned text, *ISIS Technical Report Series* 20 (2001).

[5] F.N. Bezerra and E. Lima, Low cost soccer video summaries based on visual rhythm, *Proceedings of the 14th Annual ACM International Conference on Multimedia* (Santa Barbara, CA, 23–27 October 2006) 71–77.

[6] P. Katsiouli, V. Tsetsos and S. Hadjiefthymiades, Semantic video classification based on subtitles domain terminologies, *Proceedings of SAMT Workshop on Knowledge Acquisition from Multimedia Content (KAMC)* (Genoa, Italy, 2007).

[7] C. DeMenthon, V. Kobla and D. Doermann, Video summarization by curve simplification, *Proceedings of ACM Multimedia* (1998) 211–218.

[8] B. Barbieri, N. Dimitrova and L. Agnihotri, Movie-in-a-minute: automatically generated video previews, *Proceedings of IEEE Pacific Rim Conference on Multimedia* (9–18 February 2004).

[9] K. Fujimura, K. Honda and K. Uehara, Automatic video summarization by using color and utterance information, *Proceedings of IEEE ICME* (2002) 49–52.

[10] R. Mihalcea and P. Tarau, TextRank – bringing order into texts, *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (Barcelona, Spain, 2004).

[11] G. Ercan and I. Cicekli, Lexical cohesion based topic modeling for summarization, *Proceedings of the CICLing* (2008) 582–592.

[12] P. Wang, R. Cai and S.Q. Yang, A hybrid approach to news video classification multimodal features, *Proceedings of the Joint Conference of the Fourth International Conference on Information, Communications and Signal Processing and the Fourth Pacific Rim Conference on Multimedia* (2003) 787–791.

[13] W. Qi, L. Gu, H. Jiang, X.R. Chen and H.J. Zhang, Integrating visual, audio and text analysis for news video, *Proceedings of 7th IEEE International Conference Image Processing* (2000) 520–523.

[14] N. Benjamas, N. Cooharojananone and C. Jaruskulchai, Flashlight and player detection in fighting sport for video summarization, *Proceedings of the IEEE International Symposium on Communications and Information Technology* (Beijing, China, 12–14 October 2005) 441–444.

[15] A. Money and H. Agius, Video summarisation: a conceptual framework and survey of the state of the art, *Journal of Visual Communication and Image Representation* 19(2) (2008) 121–143.
[16] A. Ekin, M. Tekalp and R. Mehrotra, Automatic soccer video analysis and summarization, *IEEE Transactions on Image Processing* 12(7) (2003) 796–807.
[17] G. Ciocca and R. Schettini, Dynamic storyboards for video content summarization, *Proceedings of the 8th ACM SIGMM International Workshop on Multimedia Information Retrieval* (Santa Barbara, CA, 26–27 October 2006).
[18] Z. Cernekova, I. Pitas and C. Nikou, Information theory-based shot cut/fade detection and video summarization, *IEEE Transactions on Circuits and Systems for Video Technology* 16(1) (2006) 82–91.
[19] A. Girgensohn, A fast layout algorithm for visual video summaries, *Proceedings of the IEEE International Conference on Multimedia and Expo* (Baltimore, MD, 6–9 July 2003) 77–80.
[20] B. Ngo, Y. Ma and H. Zhang, Video summarization and scene detection by graph modeling, *IEEE Transactions on Circuits and Systems for Video Technology* 15(2) (2005) 296–305.
[21] Y. Rui, A. Gupta and A. Acero, Automatically extracting highlights for TV baseball programs, *Proceedings of the 8th ACM International Conference on Multimedia* (Los Angeles, CA, 30 October 2000) 105–115.
[22] M. Pickering, L. Wong and S. Ruger, ANSES: summarisation of News Video, *Proceedings of CIVR-2003* (University of Illinois, IL, 24–25 July 2003).
[23] T. Tsoneva, M. Barbieri and H. Weda, Automated summarization of narrative video on a semantic level, *Proceedings of the International Conference on Semantic Computing* (17–19 September 2007) 169–176.
[24] F.N. Bezerra and E. Lima, Low cost soccer video summaries based on visual rhythm, *Proceedings of the 14th Annual ACM International Conference on Multimedia* (Santa Barbara, CA, 23–27 October 2006) 71–77.
[25] J. Kupiec, J.O. Pedersen and F. Chen, A trainable document summarizer, *Proceedings of SIGIR 1995* (ACM Press, New York, 1995) 68–73.
[26] D.R. Radev, H. Jing and M. Budzikowska, Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies, *Proceedings of ANLP/NAACL00-WS*, (Seattle, WA, 2000).
[27] G. Erkan and D.R. Radev, Lexrank: graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research* 22 (2004) 457–479.
[28] L. Page, S. Brin, R. Motwani and T. Winograd, The pagerank citation ranking: bringing order to the web, Technical Report, *Stanford Digital Library Technologies Project*, 1998.
[29] R. Barzilay and M. Elhadad, Using lexical chains for text summarization. In: I. Mani and M.T. Maybury (eds), *Advances in Automatic Text Summarization* (The MIT Press, Cambridge, MA, 1999) 111–121.
[30] M. Brunn, Y. Chali and C.J. Pinchak, Text summarization using lexical chains, *Proceedings of the Document Understanding Conference (DUC01)* (New Orleans, LA, 2001).
[31] W.P. Doran, N. Stokes, J. Carthy and J. Dunnion, Assessing the impact of lexical chain scoring methods and sentence extraction schemes on summarization, *Proceedings of CICLing* (2004) 627–635.
[32] G.H. Silber and K. McCoy, Efficient text summarization using lexical chains, *Proceedings of the ACM Conference on Intelligent User Interfaces (IUI'2000)*, 9–12 January 2000.
[33] Y. Chali and M. Kolla, University of Lethridge summarizer at DUC04, *Proceedings of DUC04* (Boston, July 2004).
[34] L. Bentivogli, P. Forner, B. Magnini and E. Pianta, Revising WordNet Domains hierarchy: semantics, coverage, and balancing, *Proceedings of COLING Workshop on Multilingual Linguistic Resources* (Geneva, 2004) 101–108.
[35] S. Banerjee and T. Pedersen, An adapted Lesk algorithm for word sense disambiguation using WordNet, *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLING-02)* (Mexico City, 2002).
[36] *Stanford Log-linear Part-Of-Speech Tagger*, Available at: http://nlp.stanford.edu/software/tagger.shtml
[37] *Penn Treebank*, Available at: www.cis.upenn.edu/~treebank/
[38] M. Hepple, Independence and commitment: assumptions for rapid training and execution of rule-based part-of-speech taggers, *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics* (Hong Kong, 2000).
[39] M. Galley and K. McKeown, Improving word sense disambiguation in lexical chaining, *Proceedings of the 18th International Joint Conference on Artificial Intelligence* (2003) 1486–1488.
[40] C.Y. Lin and E.H. Hovy, Automatic evaluation of summaries using n-gram co-occurrence statistics, *Proceedings of HLT-NAACL-2003* (Edmenton, Canada, 2003).