ANALYSIS OF GRAPH AND TEXT REPRESENTATION TECHNIQUES FOR NEWS RECOMMENDATION AND NEWS CLASSIFICATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MUSTAFA AĞRIMAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

FEBRUARY 2022

Approval of the thesis:

ANALYSIS OF GRAPH AND TEXT REPRESENTATION TECHNIQUES FOR NEWS RECOMMENDATION AND NEWS CLASSIFICATION

submitted by **MUSTAFA AĞRIMAN** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Halit Oğuztüzün
Head of Department, **Computer Engineering** _____

Prof. Dr. Pınar Karagöz
Supervisor, **Computer Engineering, METU** _____

Examining Committee Members:

Assoc. Prof. Dr. İsmail Sengör Altıngövde
Computer Engineering, METU _____

Prof. Dr. Pınar Karagöz
Computer Engineering, METU _____

Assoc. Prof. Dr. Hacer Yalım Keleş
Computer Engineering, Hacettepe University _____

Date: 07.02.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: MUSTAFA AĞRIMAN

Signature :

ABSTRACT

ANALYSIS OF GRAPH AND TEXT REPRESENTATION TECHNIQUES FOR NEWS RECOMMENDATION AND NEWS CLASSIFICATION

AĞRIMAN, MUSTAFA
M.S., Department of Computer Engineering
Supervisor: Prof. Dr. Pınar Karagöz

February 2022, 67 pages

Developments in computer science leads to increase in the use of software applications in all areas of life. This also causes an increase in data usage. Applications using textual data involves tasks such as finding similarities between texts, detecting events from texts, and classifying texts. However, using graphs and graph vectors can be more successful than textual methods of representing textual information, due to capability to express additional features and complex relationships in graph structure. In this thesis, it is hypothesized that textual data expresse in graph structure will be more successful than direct text representation in areas such as news recommendation and news classification. Within the scope of the thesis study, different graph representation methods have been applied and the results obtained from these methods have been compared with the performance under text representations.

Keywords: graph, graph embedding, graph mining, news recommendation, news classification

ÖZ

HABER ÖNERME VE SINIFLANDIRMADA ÇİZGE VE METİN GÖSTERİM YÖNTEMLERİNİN ANALİZ EDİLMESİ

AĞRIMAN, MUSTAFA
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Pınar Karagöz

Şubat 2022 , 67 sayfa

Bilgisayar biliminde yaşanan gelişmeler, hayatın her alanında yazılım uygulamalarının kullanımının artmasına neden olmaktadır. Bu da veri kullanımının artmasına neden olmaktadır. Metinsel verileri kullanan uygulamalar, metinler arasındaki benzerlikleri bulma, metinlerden olayları algılama ve metinleri sınıflandırma gibi görevleri içerir. Bununla birlikte, grafik yapısındaki ek özellikleri ve karmaşık ilişkileri ifade etme yeteneğinden dolayı, grafikleri ve grafik vektörlerini kullanmak, metinsel bilgileri temsil eden metinsel yöntemlerden daha başarılı olabilir. Bu tezde, haber tavsiyesi ve haber sınıflandırması gibi alanlarda metinsel verilerin grafik yapıda ifade edilmesinin doğrudan metin gösteriminden daha başarılı olacağı varsayılmaktadır. Tez çalışması kapsamında farklı grafik gösterim yöntemleri uygulanmış ve bu yöntemlerden elde edilen sonuçlar metin gösterimlerinin performansı ile karşılaştırılmıştır.

Anahtar Kelimeler: çizge, çizge vektörleri, çizge madenciliği, haber önerme, haber sınıflandırma

To my nuclear family

ACKNOWLEDGMENTS

Firstly, I want to thank my supervisor Prof. Dr. Pınar Karagöz because of her support. Her guidance and moral support gives me an energy to finish this thesis. Also, I would thank my thesis committee members, Assoc. Prof. Dr. İsmail Sengör Altıngövde and Assoc. Prof. Dr. Hacer Yalım Keleş for their contribution.

Secondly, I owe gratitude to my friends, Fırat Çekinel, Mert Erdemir, and Ege Çıklabakkal for their contribution to project. Also, I am very thankful for everyone who fill my user study surveys. By means of their contributions, I am able to finish this academic work.

Above all, I owe deepest gratitude to my wife, Nurseli, for her unconditional support and love. Also, thank you to my dear son, Deniz Ayaz, who is the biggest source of happiness lately.

And lastly, I would like to express my gratitude to my family and my sister, who have always stood by me and supported me.

This work (No:117E566) is funded by the Scientific and Technological Research Council of Turkey (TUBITAK).

TABLE OF CONTENTS

ABSTRACT v

ÖZ vi

ACKNOWLEDGMENTS viii

TABLE OF CONTENTS ix

LIST OF TABLES xii

LIST OF FIGURES xiv

LIST OF ABBREVIATIONS xv

CHAPTERS

1 INTRODUCTION 1

1.1 Motivation and Problem Definition 1

1.2 Proposed Methods and Models 2

1.3 Contributions and Novelties 2

1.4 The Outline of the Thesis 3

2 RELATED WORK 5

2.1 Graph Representation 5

2.2 Graph Embeddings 7

2.3 Recommendation Systems 8

3 PRELIMINARIES 11

3.1 Named Entity Recognition (NER) . . . 11
3.2 TF-IDF Term Weighting . . . 12
3.3 Cosine Similarity . . . 12
3.4 Pearson Similarity . . . 13
3.5 Euclidean Distance . . . 13
3.6 Jaccard Similarity . . . 13
3.7 Naive Bayes Classifier . . . 14
3.8 Linear Classifier (Logistic Regression) . . . 14
3.9 Support Vector Machine (SVM) Classifier . . . 15
3.10 Random Forest Classifier . . . 15
4 DATASET AND GRAPH MODEL . . . 17
4.1 Dataset . . . 17
4.1.1 Turkish Dataset . . . 17
4.1.2 English Dataset . . . 19
4.2 Graph Model . . . 21
4.2.1 Named Entity Graph Model . . . 21
4.2.2 TF-IDF Based Graph Model . . . 23
5 METHODS . . . 27
5.1 Graph Class . . . 28
5.2 Similarity Measurement (Ranking Aggregation) . . . 31
5.3 Graph Embedding Techniques . . . 32
5.3.1 Graph2vec . . . 32
5.3.2 Node2vec . . . 33

5.4 Text Embedding Techniques 34
5.4.1 Bag of Words 34
5.4.2 Doc2vec 35
5.4.3 Specter 36
6 EXPERIMENTS 39
6.1 Experimental Settings 39
6.2 Experiments and Results 39
6.2.1 Experiment 1: News Recommendation Analysis 39
6.2.2 Experiment 2: News Classification by using News Similarity . 45
6.2.3 Experiment 3: News Classification with Different Classifiers . 49
7 CONCLUSIONS 53
REFERENCES 55
APPENDICES
A SAMPLE SURVEY QUESTIONS 61

LIST OF TABLES

TABLES

Table 4.1 Named Entities and their corresponding Node IDs. We used their ids as their labels in the project for the Turkish Dataset 18

Table 4.2 Named Entities and their corresponding Node IDs. We used their ids as their labels in the graph model for the English Dataset 20

Table 5.1 Table representation of $nodes\_0.csv$. Each line represents a node in the Graph 0. 29

Table 5.2 Table representation of $relationships\_0.csv$. Each line represents an edge in the Graph 0. 30

Table 5.3 Table representation of $graph\_0.attrs$. File contains date, title, first paragraph and content information. 30

Table 6.1 An example of News Recommendation System for English dataset with Named Entity Graph Model 40

Table 6.2 An example of News Recommendation System for Turkish dataset with Named Entity Graph Model 41

Table 6.3 An example of News Recommendation System for English dataset with TF-IDF based Graph Model 42

Table 6.4 An example of News Recommendation System for Turkish dataset with TF-IDF based Graph Model 43

Table 6.5 Aggregation Count of User Studies for Turkish and English Datasets with Named Entity Graph Model and TF-IDF based Graph Model 44

Table 6.6 Aggregation Count of User Studies by using majority voting for Turkish and English Datasets with Named Entity Graph Model and TF-IDF based Graph Model 44

Table 6.7 Classification Accuracy of graph2vec, doc2vec, Specter and Hybrid Model for the Turkish Dataset with Named Entity Graph Model 46

Table 6.8 Classification Accuracy of graph2vec, doc2vec, Specter and Hybrid Model for the English Dataset with Named Entity Graph Model 46

Table 6.9 Classification Accuracy of graph2vec, doc2vec, Specter and Hybrid Model for the Turkish Dataset with TF-IDF based Graph Model 47

Table 6.10 Classification Accuracy of graph2vec, doc2vec, Specter and Hybrid Model for the English Dataset with TF-IDF based Graph Model 47

Table 6.11 Category Matching Accuracy of User Surveys for the Turkish and English Datasets with Named Entity Graph Model and TF-IDF based Graph Model 48

Table 6.12 Accuracy of Naive Bayes Classifier, Linear Classifer, SVM Classifier, Random Forest Classifier with different feature vectors and data . . . 50

LIST OF FIGURES

FIGURES

Figure 3.1 Support Vector Machine (SVM) 15

Figure 4.1 Category Distribution of the Turkish Dataset 18

Figure 4.2 Category Distribution of the English Dataset 20

Figure 4.3 Named Entity Graph Model for the Turkish Dataset 22

Figure 4.4 Named Entity Graph Model for the English Dataset 23

Figure 4.5 TF-IDF based Graph Model for the Turkish Dataset 24

Figure 4.6 TF-IDF based Graph Model for the English Dataset 25

Figure 5.1 The General Pipeline 28

Figure A.1 Information section of the user study 61

Figure A.2 A question with choices from user study for Turkish dataset . . . 62

Figure A.3 Continuation of choices of the question in Figure A.2 63

Figure A.4 A question with choices from user study for Turkish dataset . . . 64

Figure A.5 Continuation of choices of the question in Figure A.4 65

Figure A.6 A question with choices from user study for English dataset . . . 66

Figure A.7 A question with choices from user study for English dataset . . . 67

LIST OF ABBREVIATIONS

API	Application Programming Interface
CRF	Conditional Random Fields
IDF	Inverse Document Frequency
IRV	Instant-runoff Voting
NER	Named Entity Recognition
NRS	News Recommendation System
POS	Part-of-Speech
RA	Ranking Aggregation
SVM	Support Vector Machine
TF	Term Frequency

CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition

Advances in computer science have paved the way for a slew of new applications in recent years. The amount of data that has to be processed is growing by the day. All kinds of information in daily life is stored for later use. This leads to problems in fulfilling faster access to existing data, faster processing of data, and establishing relationships between data.

With the decrease in the need for traditional methods such as newspapers and television, and people starting to follow the news from computers and mobile phones, millions of different news articles appear online. News recommendation systems (NRS) are being developed to assist users find the proper and relevant material and to reduce the problem of information overload by recommending news articles that may be of interest to news readers. In the literature, there are so many different techniques in order to recommend news to users. Within the scope of this work, we aim to recommend news to users by using graph based methods and we hypothesize that graph based methods gives more accurate results than text based methods. We used graphs because there are numerous hidden relationships inside texts, embedding a text on a graph may give a stronger foundation, as graphs have been demonstrated to be a good data structure for describing textual data [1]. We may also add more attributes to the graph structure, such as named entities of nodes of the graph, the date the article was published, the author of the article, and so on. Therefore, graph-level and node-level vectorial representations that are in different dimensions can be exported from graphs and these vectorial representations can be used for similarity calculation between two

graphs.

1.2 Proposed Methods and Models

In this thesis, we aim to develop a news recommendation system by using graph based methods and hypothesize that these methods give more accurate results than text based methods. First of all, on the basis of literature review on graph modeling the news are converted to graph by using two different technique. One of them is Named Entity Graph Model, the second one TF-IDF based Graph Model. After that, node-level and graph-level vector representations of graphs are exported by using different graph embedding methods. Different similarity calculation algorithms such as cosine similarity, Euclidean distance and Pearson similarity are used in order to find similar news articles. The analysis was done for graph based and text based methods are applied on two different datasets (the Turkish and English datasets). And finally, the ground truths are obtained via user studies.

In addition to news recommendation study, news recommendation system is used for classifying the news. We have used the similarity results between news to classify the news. The classification is performed by using the graph based and text based representations that are proven to be successful in the news recommendation analysis.

In the thesis study, different graph models are analyzed on two datasets (the Turkish and English datasets). The ground truths of news recommendation are obtained through user studies, which has been a challenging and time consuming process. Also, time complexities of embedding methods and similarity calculation between news by using the embeddings are very high.

1.3 Contributions and Novelties

To the best of our knowledge, this is the first work to study comparing text and graph-based news recommendation system. Additionally, news classification is performed based on generated new recommendations. Our contributions are as follows:

1. Comparison of graph-based and text-based news recommendation systems is made.

2. Different graph embedding techniques are used for graph based news recommendation.

3. k-NN based classification method is applied by using news similarity results for different embedding techniques.

4. A hybrid model is constructed by using ranking aggregation method with different graph and text embedding techniques for news classification.

5. The performance of graph based news recommendation and news classification methods are compared against text based representations on Turkish and English news collections.

1.4 The Outline of the Thesis

The remainder of the thesis is laid out as follows. In Chapter 2, the literature review and related studies are given. The suggested model's mathematical and technical elements are provided in Chapter 3. The dataset, data preparation, and proposed graph models are all explained in Chapter 4. This study's approach, workflow and implementation details are discussed in Chapter 5. The findings of the experiments are described in Chapter 6. The conclusion is covered in Chapter 7.

CHAPTER 2

RELATED WORK

In this chapter, information will be given about graph representations, obtaining graph embeddings from graphs, making news recommendation using graph embeddings, and the studies have been done in the literature on these issues.

2.1 Graph Representation

Graphs are useful data structures for representing complex information about entities and their interactions. Graphs are popular way to describe online data in a variety of real-world applications including social networking, natural language processing, online document interactions, and recommendation. Rousseau et al. [2] propose a novel approach for the text categorization problem by classifying the graphs. In this approach, textual documents are modeled as a "graph-of-words". Within a fixed-size sliding window, vertices represent unique document words and edges reflect co-occurrences between the terms. In this study, gSpan [3] algorithm is used for extracting frequent subgraphs in order to give graph classifier as an input. In order to do text categorization, they used linear SVM classifier by feeding the classifier with frequent subgraphs. They say that this method is performed best on almost all datasets.

In another study, Genc et al. [4] proposed a novel graph model in order to detect events from social media data. In this study they name the graph model as "Snake-Graph" because the graph model contains a chain of nodes. Each unique node in a tweet and tweet id that is the starting node is added to graph model one by one like a form of snake. They also add year, month, date information to the graph model. After that, node embeddings of the each node in the graphs are generated by using

the node2vec [5] method. And finally, k nearest node embeddings are found for each date embeddings by using cosine similarity and connection between the similar nodes and events are evaluated by a human inspecter.

Huang et al. [6] proposed a novel graph model in order to do recommendation for e-commerce systems. The graph model contains user-product information in order to solve the challenges of capturing various information about user-items while also offering a flexible model that can accommodate different recommendation methodologies. To show its flexibility, they created three analytical techniques: direct retrieval, association mining, and high degree association retrieval, which cover a wide spectrum of regularly used and recently proposed recommendation approaches. They employ a two-layer graph model, in which the two levels of nodes represent products and consumers. The input information is captured through three types of links between nodes: product information, customer information, and transaction information. Each link between two products represents their similarity. Each link between two customers, likewise, captures their similarities. Their methodology may be used to execute content-based, collaborative, and hybrid techniques by selecting different types of links to employ in a recommendation generation process. According to their results, they reach the conclusion that association mining technique achieved the best precision in their e-commerce recommendation system.

Schenker et al. [7] proposed a new graph model in order to cluster web documents. Firstly, each unique word in the documents is a node for the graph and each node is connected to following node in the document. Edges are labelled directed and labels are title, link and text. The matching edge is labeled title if the terms exist in the title. If the phrases appear in the URL link section, the link is marked on the appropriate edge. Otherwise, it is categorized as text, implying that both phrases are contained in the textual body of the document. And then, they use k-means algorithm in order to cluster similar graphs. Their model's experimental findings are compared to the vector-based k-means algorithm, and the suggested model outperforms the baseline model, according to the researchers. The reason for this is that a graph representation captures structural information such as the index of the words and the order in which they appear. However, with a vector-based representation, this type of information is lost.

2.2 Graph Embeddings

Graphs may be found in a broad range of real-world applications, such as social graphs/diffusion graphs in social media networks, citation graphs in research fields, user interest graphs in electronic commerce, knowledge graphs, and so on. Analyzing these graphs gives insight on how to make effective use of the information hidden in graphs, and has thus earned a lot of attention in recent decades [8].

Recently, several methods are proposed to represent the graph (i.e nodes, subgraph, entire graph) in a latent vector space. To start with node embeddings, many of node embedding algorithms rely on shallow embedding [9]. In this embedding technique, there is an encoder and decoder. There are two types of approaches in this technique: matrix factorization-based approach and random walk based approach. Laplacian Eigenmaps [10], Graph Factorization (GF) [11], GraRep [12] and HOPE [13] are the methods for matrix-factorization based approach. This approach is based on traditional dimensionality reduction [14] techniques. The Graph Factorization (GF) technique uses the adjacency matrix to define node similarity directly; GraRep uses the adjacency matrix's different powers to capture higher-order node similarity; and the HOPE algorithm offers general similarity measures. DeepWalk [15] and node2vec [5] are the methods for random walk based approach. The random walk methods try to maximize the probability of visiting a vertex v_j from an initial vertex v_i, $p(v_j|v_i)$ if v_j is visited by a walk of length T. These techniques employ negative sampling and update just a small fraction of the weights of negative instances (vertices) at each iteration since optimizing embeddings for each vertex at each step is exceedingly expensive.

Also, whole graph or subgraph can be represent as a single latent vector. Sub2vec [16] is an unsupervised method for learning feature representations of arbitrary subgraphs. This method wants to preserve two different properties named neighborhood and structural of subgraphs. Graph2vec [17] learns different sized graphs and embeds them into a fixed length vector using unsupervised learning methods. The notion relies on the skipgram networks, which is mostly associated with doc2vec [18], an extention of word2vec [19] technique.

2.3 Recommendation Systems

Recommendation systems assist people in making decisions when they lack adequate personal experience with the alternatives, and they are especially important in this age of information overload [20]. Because of the increased quantity of information that consumers must analyze before deciding which things satisfy their needs, recommendation systems have begun to be widely used in real life. Pongpech et al. [21] created a recommendation system that suggests a set of prioritized recommendation objectives for each individual based on his or her completed courses in one of their works in the field of recommendation systems. The majority of recommendation systems involve rating things based on collaboration filtering, i.e. model based items. However, in the education domain, where specialists must pay more attention to the learners' educational backgrounds and needs in order to provide appropriate recommendations, such a ranking technique may not be suitable. Because of this reason, they base their recommendations on each learner's educational background, which they depict as a collection of completed courses. In their recommendation process, these completed courses are regarded as a key index.

As a different usage area of recommendation systems, recommendation systems are frequently used by e-commerce sites to propose products to their customers. A recommender system for an e-commerce site takes information about the items a customer is interested in and suggests products that are likely to meet her needs [22]. Sarwar et al. [22] did a study in order to compare existing recommendation system for e-commerce sites and they achieve that collaborative filtering based algorithms with dimensionality reduction techniques have the potential to scale to enormous data sets while still producing high-quality suggestions. The collaborative filtering approach compares individuals based on their known preferences and suggests things that are chosen by similar users and collaborative filtering is widely used in recommendation systems. In this study [22], user-based collaborative filtering has been used. Cluster-based collaborative filtering [23] , Item-based collaborative filtering [24] , Dimension reduction based collaborative filtering [25] , Bayesian network based recommendation [26] , Horting Graph-theoretic collaborative filtering [27], Association rules based recommendation [28] have been used in different studies too.

Since the internet allows access to news materials from millions of sources throughout the world, online news reading has grown in popularity. One of the biggest challenges for news websites is assisting readers in finding interesting content to read. In this research area, Li et al. [29] proposed a novel approach for news recommendation. They offer a scalable two-stage customized news recommendation strategy based on a two-level representation that takes into account the unique properties of news items (e.g., news content, access patterns, named entities, popularity, and recency) when making recommendations. A systematic framework for news selection based on the inherent feature of user interest is also offered, with a good balance between the proposed result's originality and variety. The recommendation is carried out in two stages, the first of which divides the news collection into groups, and the second of which seeks to propose news items.

In another study, Liu et al. [30] proposed a click based personalized news recommendation study. They presented a study on how to create an efficient information filtering system for news recommendations in a large-scale website like Google News. They started with a log analysis to see how user interests in news subjects changed over time. The log analysis revealed differences in users' news interests and revealed that individual users' news interests are impacted by the local news trend. A Bayesian framework is presented for modeling a user's genuine interests based on her previous click history and predicting her current interests by combining her genuine interest with the local news trend. In content-based news recommendation, the mechanism for predicting user interests was merged with the current collaborative filtering process to provide tailored news suggestions. They used the hybrid technique to test the news recommender on a small portion of real traffic on the Google News website. The experiment revealed that the hybrid strategy increased the quality of news suggestions and drew more frequent visits to the Google News website when compared to the conventional collaborative filtering method.

CHAPTER 3

PRELIMINARIES

In this chapter, technical background for the approaches, paradigms, and algorithms employed in this thesis is described in order to serve as a foundation for the rest of the thesis.

3.1 Named Entity Recognition (NER)

Named entity recognition (NER) is a subtask of information extraction that aims to discover and categorize identified entities referenced in unstructured text into predefined categories such as person names, organizations, places, time expressions, quantities, percentages, and so on. Named entities are significant targets for most language processing algorithms since they hold key information in a phrase. Accurate named entity recognition is a valuable source of data for a variety of NLP applications [31]. Named entity recognition research employ a variety of methodologies. Machine learning and deep learning-based systems are being produced more commonly today, although statistical and rule-based research were dominating in the earlier systems [32].

In this study, CRF-based NER model that was proposed by Cekinel et al. [32] is used to identify entities for the Turkish dataset, Spacy[1] that is a NLP library is used to identify entities for the English dataset.

[1] https://spacy.io/

3.2 TF-IDF Term Weighting

The term frequency value and inverse document frequency value are combined to create the TF-IDF measure. The words with the highest TF-IDF term weight ratings are considered to be more significant [33]. The number of times a word appears in a document is directly proportional to its term frequency (TF). The inverse document frequency (IDF), on the other hand, gives rare words greater credit.

In our model, normalized term frequency (TF) value is calculated using the formula given in Equation 3.1 and inverse document frequency (IDF) value is calculated using the formula given in Equation 3.2.

$$TF(w,d) = 0.5 + 0.5 * \frac{f_{w,d}}{max(f_{w',d} : w' \in d)} \tag{3.1}$$

$$IDF(w,D) = log(\frac{N}{1 + |d \in D : w \in d|}) \tag{3.2}$$

Using the algorithm in Equation 3.3, we calculate the TF-IDF score of each token in the whole dataset. Then, because the TF-IDF score may be utilized as a significance indicator, words with low TF-IDF scores are excluded.

$$TF - IDF(w,D) = max(tf(w,d) : d \in D) * idf(w,D) \tag{3.3}$$

3.3 Cosine Similarity

The cosine of the angle between two n-dimensional vectors is named as cosine similarity [34]. It is the dot product of the two vectors divided by the product of the two vectors' magnitudes. The values range from -1 to 1, with -1 being the most different and 1 being the most similar. Calculation of cosine similarity in this study is given in Equation 3.4.

$$similarity(A,B) = \frac{A \cdot B}{||A|| * ||B||} \tag{3.4}$$

3.4 Pearson Similarity

Pearson similarity is defined as the product of the covariances of two n-dimensional vectors divided by their standard deviations [35]. The values range from -1 to 1, with -1 being the most different and 1 being the most similar. Calculation of Pearson similarity in this study is given in Equation 3.5.

$$similarity(A, B) = \frac{cov(A, B)}{\sigma_A \sigma_B} \tag{3.5}$$

3.5 Euclidean Distance

The Euclidean distance is the distance between two coordinates in n-dimensional space measured in straight lines [36]. The smaller the distance between two points, the more similar they are. Calculation of euclidean distance in this study is given in Equation 3.6.

$$distance(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2} \tag{3.6}$$

3.6 Jaccard Similarity

The size of the intersection divided by the size of the union of two sets is known as Jaccard similarity [37]. The values range from 0 to 1, with 0 being the most different and 1 being the most similar. Calculation of Jaccard similarity in this study is given in Equation 3.7.

$$similarity(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{3.7}$$

3.7 Naive Bayes Classifier

By assuming that characteristics are independent of class, the Naive Bayes classifier dramatically simplifies learning. Although independence is often over simplifying assumption, Naive Bayes often outperforms more complicated classifiers in a variety of real world problems [38]. It is a classification method based on Bayes' Theorem and the assumption of predictor independence. A Naive Bayes classifier, in basic words, posits that the existence of one feature in a class is independent to the presence of any other feature. Bayes' Theorem is given in Equation 3.8.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{3.8}$$

where, $P(A|B)$ is the posterior probability of class, $P(B|A)$ is the likelihood, $P(A)$ is the prior probability of the class and $P(B)$ is the prior probability of the predictor.

Since it has a better success rate than other algorithms, Naive Bayes classifiers are commonly employed in text classification. As a result, it is commonly utilized in spam filtering and sentiment analysis applications. Additionally, Naive Bayes classifier is used with different techniques as a hybrid solution. Ghazanfar et al. [39] proposed a recommendation system that is constructed on the combination of the Naive Bayes Classifier and Collaborative Filtering. The system employs machine learning and data mining techniques to filter unseen data and forecast whether a user would enjoy a given resource.

3.8 Linear Classifier (Logistic Regression)

A linear classifier classifies objects based on the value of a linear combination of parameters. The qualities of an object are also known as feature values, and they are often supplied to the computer as a feature vector. These classifiers perform well for real issues like document classification and, more broadly, problems with multiple characteristics, achieving accuracy levels comparable to non-linear classifiers but requiring less training and usage time [40].

The logistic regression model, which, despite its name, is a classification rather than a regression method, is another approach to linear classification. Logistic regression uses linear functions to describe the odds of an observation belonging to each of the K classes, ensuring that the probabilities add to one and remain in the (0, 1) range.

3.9 Support Vector Machine (SVM) Classifier

SVM is a supervised machine learning technique that may be used to solve classification and regression problems [41]. It is, however, mostly employed to solve classification tasks. Each data item is plotted as a point in n-dimensional space (where n is the number of features you have), with the value of each feature being the value of a certain coordinate in the SVM algorithm. Then we accomplish classification by locating the hyper-plane that clearly distinguishes the two classes.

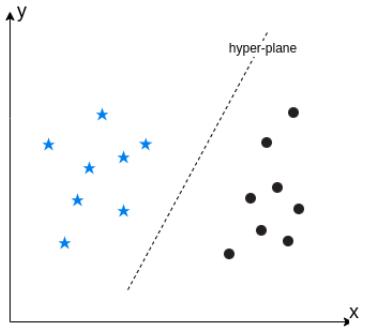

Figure 3.1: Support Vector Machine (SVM)

3.10 Random Forest Classifier

Random forest is an ensemble learning approach for classification, regression, and other problems that generates a large number of decision trees [42]. For classification tasks, the random forest's output is the class chosen by the majority of trees. Random forests outperform decision trees in most cases, but they are less accurate than gradient enhanced trees. Data features, on the other hand, might have an impact on their performance.

CHAPTER 4

DATASET AND GRAPH MODEL

4.1 Dataset

In this thesis work, two different datasets are used. The first one is the Turkish dataset and the second one is the English dataset.

4.1.1 Turkish Dataset

The Turkish dataset has been collected from an online Turkish newspaper site[1]. The news data contains news between January 1, 2015 and December 31, 2016 and 32381 online news data. Also, it consists of approximately 500k words. The words are preprocessed and labeled according to their Named Entities. Categorization of the words is performed in 8 different named entity types which are *PERSON*, *PERCENT*, *LOCATION*, *ORGANIZATION*, *TIME*, *DATE*, *MONEY* and *OTHER* and unique ids are assigned to each of them that is shown in Table 4.1. News are labeled as one of 27 categories: Sustainable World (Sürdürülebilir Dünya), Life (Yaşam), Textile (Tekstil), News from the Home (Yurttan Haberler), Economic Data (Ekonomik Veriler), Business World (İş Dünyası), Industires (Sektörler), Health (Sağlık), Insurance (Sigortacılık), Logistic (Lojistik), World Food (Dünya Gıda), Sport (Spor), Companies (Şirketler), Real Estate (Emlak), Tourism (Turizm), Culture and Art (Kültür-Sanat), Education (Eğitim), Export (İhracat), Technology (Teknoloji), Automotive (Otomotiv), Finance (Finans), Agriculture (Tarım), Economy (Ekonomi), World (Dünya), Energy (Enerji), Agenda (Gündem) or Other (Diğer). Category distribution is given detailed in the Figure 4.1

[1] https://www.dunya.com

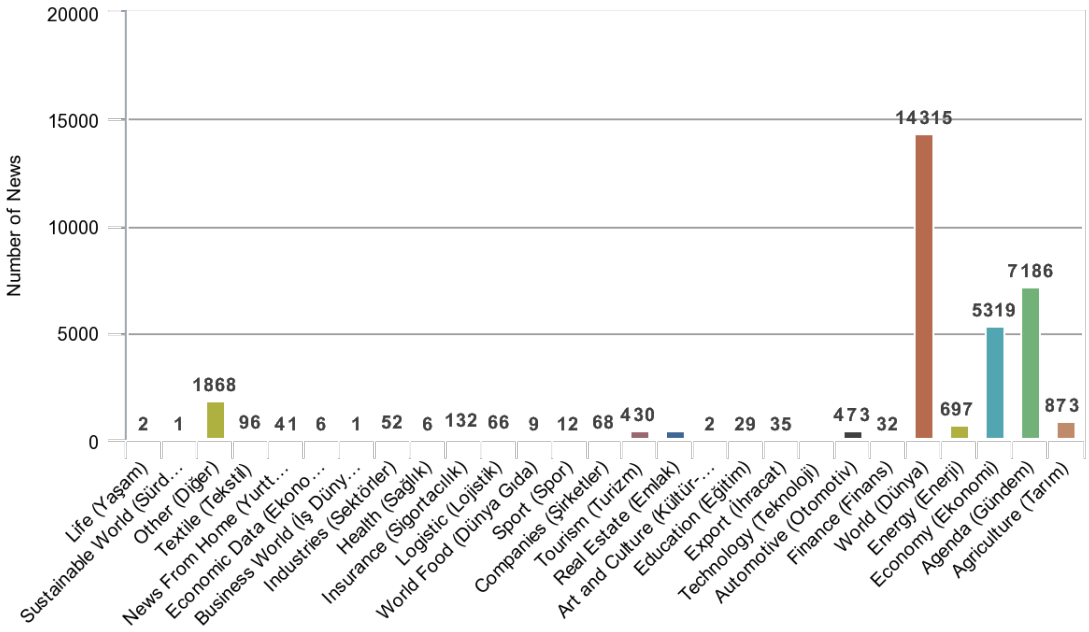

Figure 4.1: Category Distribution of the Turkish Dataset

Table 4.1: Named Entities and their corresponding Node IDs. We used their ids as their labels in the project for the Turkish Dataset

id	Named Entity
0	ORGANIZATION
1	LOCATION
2	PERSON
3	PERCENT
4	MONEY
5	DATE
6	TIME
7	OTHER

Preprocessing

Turkish is morphologically rich language, and words can take different forms and have different meanings. Verbs can have different structural forms and words can take many different suffixes. Because of these reasons, lemmas gives more accu-

rate results instead of surface forms. Zemberek[2] library is used in preproccessing since it provides rich natural language processing modules for Turkish. Preprocessing is handled in stages. Firstly, unnecessary words are removed from the news such as non-unicode characters, URLs and emojis. Secondly, data is tokenized in order to convert sentences to single tokens. Thirdly, for identifying the semantics of the words more precisely disambigution is used. And finally, lemmatization is applied so that forms of the words are grouped and CRF-based NER model that was proposed by Cekinel et al. [32] is used for finding named entities. The Turkish news and social media datasets annotated by Tur et al. [43] and Seker et al. [44] are used to train the NER model. Thus, we have the preprocessed data, matched with named entities. At the end of these steps data is ready to be imported to a graph database.

4.1.2 English Dataset

The English dataset is a public dataset that has been taken from BBC[3]. It contains 2225 news instances. The dataset is splitted into 1490 records for training and 735 for testing. The words in dataset are preprocessed and labeled according to their named entities. Categorization of the words is performed in 17 different types of named entities which are *ORDINAL*, *GPE*, *DATE*, *PERCENT*, *MONEY*, *PERSON*, *ORG*, *EVENT*, *PRODUCT*, *LANGUAGE*, *TIME*, *LAW*, *LOC*, *CARDINAL*, *NORP*, *WORK OF ART*, *FAC* and *QUANTITY* and unique ids are assigned to each of them that is shown in Table 4.2. News are labeled as one of 5 categories: Business, Entertainment, Politics, Sport or Technology. Category distribution is given detailed in the Figure 4.2.

Preprocessing

Spacy[4] English Named Entity Recognizer library is used in order to preprocess the English dataset. This library is very successful to recognize named entities in English. Also, it provides a pipeline while finding named entities. The pipeline contains tokenization, disambigution, lemmatization and stop word removing steps. Thus, data

[2] https://github.com/ahmetaa/zemberek-nlp
[3] https://www.bbc.com/
[4] https://spacy.io/api/entityrecognizer

is preprocessed, and matched with named entities. At the end of these steps data is ready to be imported to a graph database.

Table 4.2: Named Entities and their corresponding Node IDs. We used their ids as their labels in the graph model for the English Dataset

id	Named Entity	id	Named Entity
0	ORDINAL	9	LANGUAGE
1	GPE	10	TIME
2	DATE	11	LAW
3	PERCENT	12	LOC
4	MONEY	13	CARDINAL
5	PERSON	14	NORP
6	ORG	15	WORK OF ART
7	EVENT	16	FAC
8	PRODUCT	17	QUANTITY

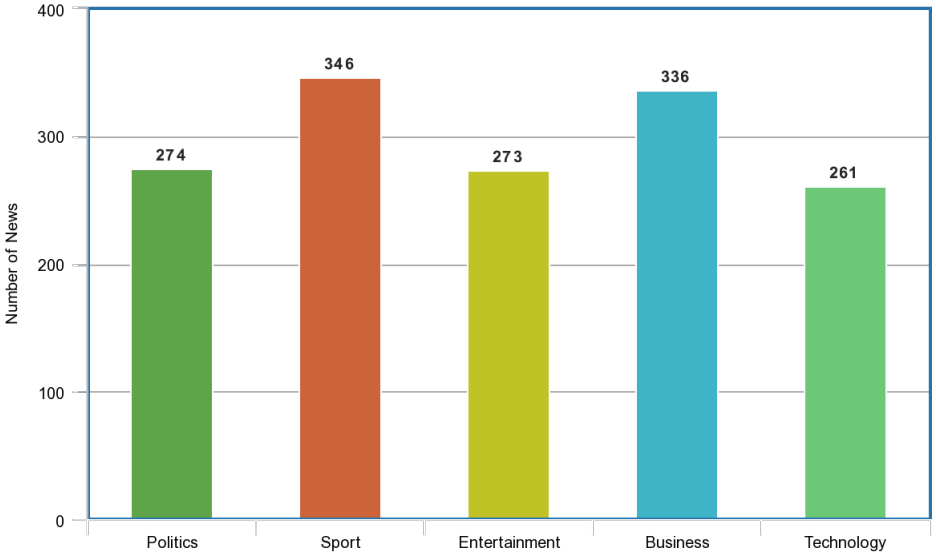

Figure 4.2: Category Distribution of the English Dataset

4.2 Graph Model

A graph G is formally defined as $G = (V, E)$ where V is a set of vertices, E is a set of edges. Two different graph models by using named entities, and TF-IDF scores are constructed. We have used Neo4j Graph Database[5] in order to store these graph models.

4.2.1 Named Entity Graph Model

In this graph model, a news is represented as an unweighted directed graph. In this graph model, named entities in the news are added to graph structure as a node. Sample Named Entity Graph Models for the Turkish and English datasets are given in Figure 4.3 and Figure 4.4, respectively.

Nodes

In the Named Entity Graph Model, news content is stored as graph and graphs contain 3 different nodes:

- **News node** is used to represent news document and contains the corresponding news id that is a unique number. These nodes are connected to *Word* nodes with *CONTAINS* edge.
- **Word node** is used to show each unique word in all news documents. These nodes can be connected to several different *News* nodes. Also, *Word* nodes can be connected to other *Word* nodes with *FOLLOWED_BY* edge.
- **NamedEntity node** is used to represent named entity types of the words in news data. These nodes are connected to *Word* nodes with *IS* edge.

Relationships

In the Named Entity Graph Model, 3 different types of edges are used to connect the nodes:

[5] https://neo4j.com

Figure 4.3: Named Entity Graph Model for the Turkish Dataset

- **CONTAINS** edge is used to connect *News* and *Word* nodes. *News* node is starting point of the edge, *Word* node is end point of the edge. This relationship is labeled as "1" in our model.

- **IS** edge is used to connect *Word* and *NamedEntity* nodes. *Word* node is starting point of the edge, *NamedEntity* node is end point of the edge. This relationship is labeled as "2" in our model.

- **FOLLOWED_BY** edge is used to connect *Word* nodes to each other. The main purpose of this relationship is preserving which word comes after another word in a news content. This relationship is labeled as "3" in our model.

Figure 4.4: Named Entity Graph Model for the English Dataset

4.2.2 TF-IDF Based Graph Model

In this graph model, a news is represented as an unweighted directed graph. Nodes of the graph are the unique words with the high TF-IDF scores where the threshold is determined by user. Sample TF-IDF based Graph Models for the Turkish and English datasets are given in Figure 4.5 and Figure 4.6, respectively.

Nodes

In TF-IDF based Graph Model, news content is stored as graph and graphs contain 2 different nodes:

- **News node** is used to represent news document and contains the corresponding news id that is a unique number. These nodes are connected to *Word* nodes with *CONTAINS* edge.
- **Word node** is used to show each unique word in all news documents. These nodes can be connected to several different *News* nodes. Also, *Word* nodes can

be connected to other *Word* nodes with *FOLLOWED_BY* edge.

Relationships

In TF-IDF based Graph Model, 2 different types of edges are used to connect the nodes:

- **CONTAINS** edge is used to connect *News* and *Word* nodes. *News* node is starting point of the edge, *Word* node is end point of the edge. This relationship is labeled as "1" in our model.
- **FOLLOWED_BY** edge is used to connect *Word* nodes to each other. The main purpose of this relationship is preserving which word comes after another word in a news content. This relationship is labeled as "2" in our model.

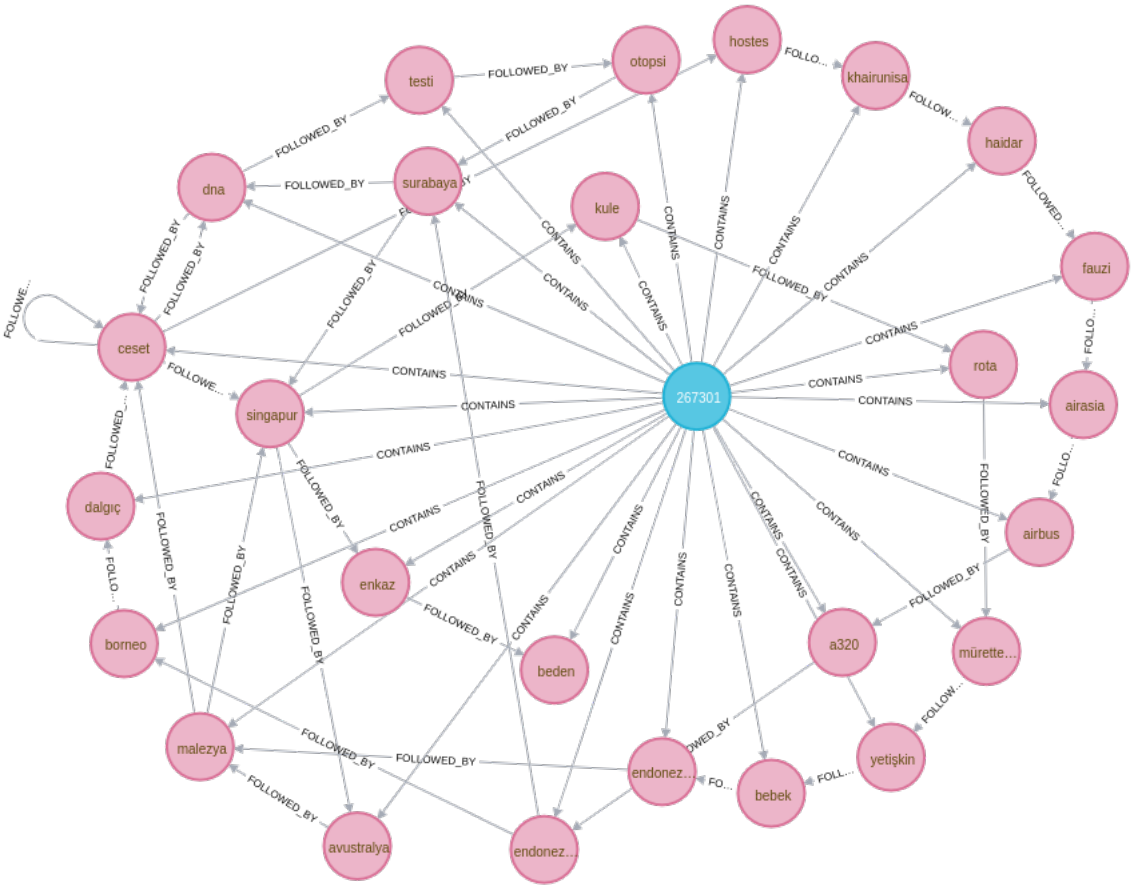

Figure 4.5: TF-IDF based Graph Model for the Turkish Dataset

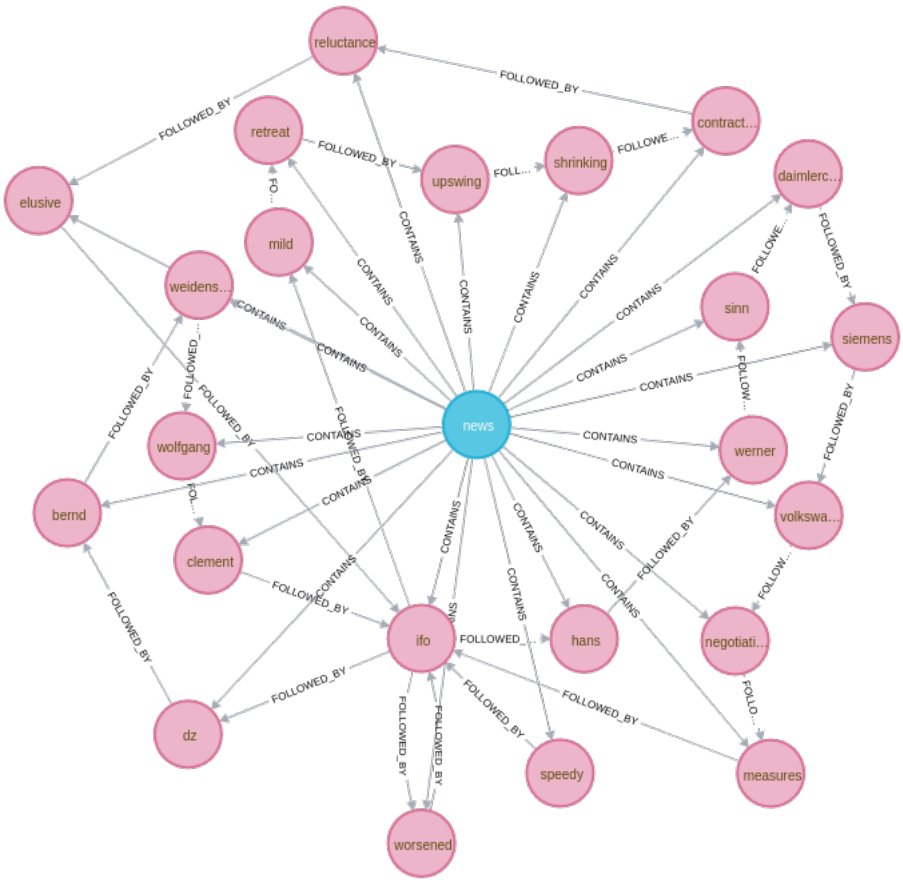

Figure 4.6: TF-IDF based Graph Model for the English Dataset

CHAPTER 5

METHODS

In our study we used two different approaches for news recommendation. In the first approach, the textual data is preprocessed and the graph-of-words structure for each news is generated. Two different graph model as mentioned in Chapter 4 were created: Named Entity Graph Model and TF-IDF based Graph Model. After that, by using different graph embedding algorithms such as graph2vec and node2vec, graph and node level embeddings of the news were exported. Different similarity calculation algorithms such as cosine similarity, Euclidean distance, and Pearson similarity were applied in order to find similar news for news recommendation system (NRS). In the second approach, different text embedding algorithms such as doc2vec, bag-of-words and Specter were used in order to generate textual embeddings of the textual data. After that, different similarity calculation algorithms such as cosine similarity, Euclidean distance, and Pearson similarity were applied in order to find similar news for news recommendation system (NRS). And finally, to decide which method is better we prepared user studies and according to these survey results we decided which method is better.

Additionally, by using the our news recommendation system, we developed a method to classify the news. The news in the datasets are labelled according to their topics. In this task, we predicted the label of the news by using the results of the news recommendation system. The developed pipeline for the news recommendation and classification system is shown in Figure 5.1.

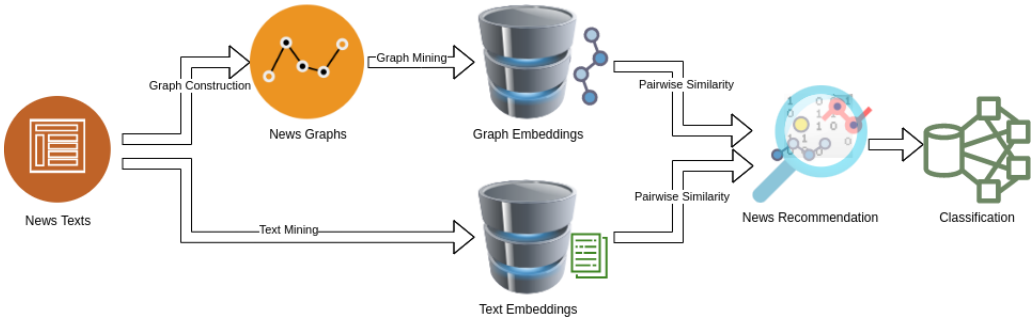

Figure 5.1: The General Pipeline

5.1 Graph Class

The graph class is designed in order to have a common graph structure that can be easily converted to different input formats of each graph/node embedding technique. Python language is used while implementing the graph class. In a graph object, in addition to nodes and edges (vertices and relationships), attributes such as content, and date of the news can be kept as well. Different methods are implemented in our graph class to export appropriate data input for corresponding embedding technique. Graph object are stored in Neo4j Graph Database[1] by using Labeled Property Graph[2] approach.

Implementation Details

First of all, the textual data is preprocessed and an appropriate data to import Neo4j Graph Database is created. After that, the graph data is exported from Neo4j Graph Database in order to create the graph objects based on the related graph model. We have exported information of nodes and relationships of the news in CSV file format that are named as $nodes\_x.csv$ and $relationships\_x.csv$ where x denotes the news id. By using a script the attribute files named $graph\_x.attrs$ where x denotes the news id from raw news data are generated. While performing these operations, the

[1] https://neo4j.com
[2] https://neo4j.com/blog/rdf-triple-store-vs-labeled- property-graph-difference/

Table 5.1: Table representation of $nodes\_0.csv$. Each line represents a node in the Graph 0.

nodes_0.csv	
id	**label**
61	katil
60	binlerce
...	
19	bund
...	
15	sangay
...	
7	7
...	

studies[3] done within the scope of the TEGHUB project[4] are taken as reference. The basic node, relationship and attribute input files are given in Table 5.1, Table 5.2 and Table 5.3, respectively.

Finally, input data files are imported to graph objects based on the related graph model and by using the related methods appropriate data files are created for the graph embedding techniques.

[3] https://github.com/teghub/news$_prediction$
[4] https://teghub.ceng.metu.edu.tr

Table 5.2: Table representation of $relationships\_0.csv$. Each line represents an edge in the Graph 0.

relationships_0.csv			
start_id	**end_id**	**label**	**type**
8	61	1	CONTAINS
8	60	1	CONTAINS
...			
61	7	2	IS
60	7	2	IS
...			
19	1	2	IS
...			
15	1	2	IS
...			
19	20	3	FOLLOWED_BY
20	21	3	FOLLOWED_BY
...			

Table 5.3: Table representation of $graph\_0.attrs$. File contains date, title, first paragraph and content information.

graph_0.attrs
'date': '2015-01-01', **'title'**: "Çin'de kutlama faciaya dönüştü: 35 ölü! | Dünya", **'firstParagraph'**: "Çin'in en büyük kenti Şangay'da, yılbaşı kutlamalarında yaşanan izdihamda 35 kişi hayatını kaybetti", **'content'**: "Çin'in resmi haber ajansı Şinhua, Şangay'da deniz kenarındaki Bund bölgesinde yapılan yeni yıl kutlamalarında izdiham çıktığını bildirdi. İzdihamda 35 kişinin öldüğü, 42 kişinin yaralandığı belirtildi. Yaralılar Şangay Hastanesi'ne kaldırıldı. İzdihamın nedeni araştırılıyor. Nüfusu 24 milyonu bulan Şangay'ın Bund bölgesinde dar sokaklar, restore edilmiş eski binalar, dükkanlar ve turistik yerler bulunuyor. Bölgede yeni yıl gibi önemli olayların kutlamalarına binlerce kişi katılıyor."

5.2 Similarity Measurement (Ranking Aggregation)

The act of integrating numerous ranked lists into a single ranking is known as ranking aggregation (RA). Ranking aggregation's purpose is to find a ranking that merge and best represents all of the input ranks. Information retrieval [45, 46], bioinformatics [47], estimator ranking [48], and management [49] are only a few of the successful applications where ranking aggregation is increasingly employed. The batch mode methods and the instant-runoff mode methods are the two primary types of ranking aggregation methods [50]. The ranking positions of all items in a batch are generated using batch mode methods. Borda count [51] is a batch mode approach that works directly on the items' ranks in the input rankings. The instant-runoff mode methods divide the aggregate of n item rankings into n rounds in a sequential order. Only one item is ranked in each round, and it will not be included in the subsequent rounds' rankings of unranked things. A sample instant-runoff mode method is instant-runoff voting (IRV) [52]. It simply ranks the best (or worst) item in each round, which is determined by the number of votes received in the first place.

Implementation Details

In order to reveal the similar news data, different similarity calculation methods that are Cosine Similarity, Euclidean Distance and Pearson Similarity were used. Ranking aggregation was used in order to combine the results that are generated from different similarity calculation methods. For each similarity calculation methods, a list that contains 10 most similar news ids for each news are exported. And then, a weight is assigned to each news id according to its order in the list. For example, the most similar news id has a weight of 10, the second news id has a weight of 9 and the weights are decreased one by one. Then, for each unique news id, the weights are sum up and sorted them in descending order according to their weights. And finally, a list that contains 10 most similar news ids are generated. This method is referred to as Borda count [51] in the literature.

5.3 Graph Embedding Techniques

5.3.1 Graph2vec

Graph2vec[5] is a neural embedding system that allows users to learn data-driven distributed representations of graphs of any size. Naranayan et al. [17] propose that a whole graph can be viewed as a document, and the rooted subgraphs surrounding each node in the graph can be viewed as words that make up the text, and that document embedding neural networks be used to learn representations of entire graphs. It is based on the Doc2vec [18] framework, which employs word embeddings and a document embedding to train a neural network and learns the latent representation of a document. The graph2vec model provides a single embedding for each graph, we used these vectors for similarity calculation between other graphs. Graph2vec outperforms substructure representation learning techniques in classification and clustering accuracy, and is competitive with state-of-the-art graph kernels [17].

Implementation Details

By using the methods in the graph class, a JSON file is exported for each graph for graph2vec since the input format of graph2vec is JSON file. The basic input file format is as follows:

$$\{"edges" : [[0, 1], [0, 2], [1, 2]],$$
$$"features" : \{"0" : "12", "1" : "2", "2" : "30"\}\}$$

Each element in the list of edges represents a relation between nodes. First item in each element is the starting point, second one is the end point. Features show the nodes and their corresponding labels. Node labels is chosen from node unique id.

Finally, input files are used in order to export graph embeddings by using *graph2vec*. The epoch and dimension size are arranged by reference from an information that was proposed by Yilmaz et al. [1]. We adjusted the epoch size to 50, dimension size to 64 in the implementation. The output of the *graph2vec* is a CSV file that contains

[5] https://github.com/benedekrozemberczki/graph2vec

the embedding vectors of each graph with the length of the dimension.

Similarity Calculation: By using the embedding vectors that are exported from graph2vec similarity calculation is done. We have used 3 different similarity algorithms:

- Cosine Similarity
- Euclidean Distance
- Pearson Similarity

Each algorithm generates a list for each graph that consist of 10 similar graph ids that are sorted from most similar to least similar. After that, the common result are found by using ranking aggregation method with the 3 different results.

5.3.2 Node2vec

Node2vec[6] is a node embedding technique that maps the nodes to low-dimensional space while maximizing the likelihood of preserving network neighborhood of nodes [5]. Node2vec learns node embeddings using short and biased random walks, with the randomness of the walks controlled by parameters. For each node in the graph, node2vec generates a vector in d dimensional space. These vectors are used in order to calculate similarity between graphs.

Implementation Details

By using the methods in the graph class, the graph objects are converted to $Networkx$ graph model, since node2vec accepts this graph model as an input. Dimension, random walk length and number of walks parameters of the node2vec algorithm are optimized with respect to the input data. After all optimizations, we adjusted the dimension to 20, random walk length to 16 and number of walks to 100. The output of

[6] https://github.com/aditya-grover/node2vec

node2vec consists of embedding vectors for each node in the graph with the length of the dimensions.

Similarity Calculation: We have used two similarity calculation methods:

1. Euclidean distance and Jaccard similarity are used in order to find the similarity between graphs. The similarity calculation method is as follows: First of all, graphs are embedded using *node2vec* and embeddings of each node in the graphs are found. Assume that there are two graphs to find the similarity between. After that, the same nodes in these two graphs are found. The most similar 60 percent of the total nodes in the graphs are found for the common nodes. In order to find the most similar nodes, Euclidean distance is used. Once the most similar 60 percent of total node embeddings are found and the node counterparts of these embeddings are kept in a list. This means track of actual nodes which are most similar to common words are kept. This way, a list of relatively similar words for each graph are obtained. As the next step, it is calculated how similar these common nodes are. At this point, Jaccard similarity is used to obtain an overall similarity percentage between the given graphs. In Chapter 6, this method is referred to as *node2vec*.

2. In the second method, embedding for a graph is directly obtained by averaging the node embeddings of the nodes. Then the similarity between two graphs is obtained by measuring the distance between their embeddings through cosine similarity, Euclidean distance and Pearson similarity. Finally, the similarity is determined as the common result of these three similarity calculation methods. In Chapter 6, this similarity calculation method is referred to as *node2vec2*.

5.4 Text Embedding Techniques

5.4.1 Bag of Words

It is a typical difficulty in Natural Language Processing applications to represent textual data in vector space. One of the most extensively used methods for embedding

textual data into vector space is the Bag of Words [53] approach. Rather than structure, this technique is just concerned with the word's appearance in a text. The vector size is smaller than or equal to the vocabulary size. The vectors do not preserve the semantic relationship between the terms.

Implementation Details

All of the nodes in the graphs are traversed over and a vocabulary is generated. By using this vocabulary, *One Hot Encoding* form of the graphs are exported. If an element of the vector is 0, it means that this node does not exist in the corresponding graph. If it is 1, it means this node exists in the corresponding graph. A basic example of One Hot Encoded graph is as follows:

$$[0, 0, 1, 0, 1, ..., 0, 1, 0, 0, 0, 1]$$

Similarity Calculation: For Bag of Words representation, cosine similarity, Euclidean distance and Pearson similarity methods are used, as in *graph2vec*. Therefore, a list for each news that consist of 10 similar news ids that are sorted from most similar to least similar is exported.

5.4.2 Doc2vec

To create fixed-lenght feature vector from text, bag-of-words method comes to mind first. But, bag-of-words have some weaknesses: it loses track of the words' order and ignores the semantics of the words. Each document is represented by a dense vector that has been trained to anticipate words in the document via Doc2vec [18]. Due to its design, the algorithm has the ability to overcome the drawbacks of bag-of-words method. The algorithm[7] is based on word2vec [19] and it creates a fixed-length feature vector for each documents in the corpus.

[7] https://github.com/RaRe-Technologies/gensim/blob/develop/gensim/models/doc2vec.py

Implementation Details

By using the contents of the graphs, the content of the each news are tokenized and given to *doc2vec* in order to train the model. While creating the doc2vec model, we set dimension to 20, window size to 2, and epoch size to 100.

Similarity Calculation: By using the content of the graphs as test file for the Doc2vec model, we have exported the most similar graphs by using cosine similarity. As a result, list for each news that consist of 10 similar news ids that are sorted from most similar to least similar is exported.

5.4.3 Specter

Specter [54] is a novel approach for generating document-level embedding of scientific documents that uses a transformer language model that has been pre-trained on a powerful indication of document-level relatedness: the citation graph. Recent Transformer language models, like as BERT [55], develop powerful textual representations, but they are focused on token and sentence level training objectives and do not take use of inter-document relatedness information, limiting their document-level representation capacity.

Implementation Details

By using the graph model, a JSON file that is in appropriate format for Specter is exported. Each item in file contains the graph id, graph title and graph content. By using the web API that is offered by specter, the embeddings of the graph contents in 768 dimension are exported. Since the Specter gives the results with this dimension, we did not change this value for optimization.

Similarity Calculation: Cosine similarity, Euclidean Distance and Pearson similarity algorithms are applied in order to find the most similar news with corresponding

news, as in *graph2vec*. Therefore, a list for each news that consist of 10 similar news ids that are sorted from most similar to least similar is exported.

CHAPTER 6

EXPERIMENTS

6.1 Experimental Settings

We conducted a set of experiments to analyze the graph based news recommendation and news classification performance against text based representations. In the first experiment, news recommendation with graph based and text based representations are compared. In the second experiment, news classification performance by using the similarity results of best graph based and text based methods in the first experiment. In this analysis, the performance under Named Entity Graph Model and TF-IDF based Graph Model are compared as well. In the third experiment, news classification by using Naive Bayes classifier, Linear regression classifier, Support Vector Machine classifier and Random Forest classifier with different feature vectors are compared. The results are also compared against the solutions in the second experiment.

6.2 Experiments and Results

6.2.1 Experiment 1: News Recommendation Analysis

In the first experiment, the performance of graph based and text based approaches are compared for news recommendation. For text-based methods, doc2vec, Bag-of-words (BOW) and Specter is used and for graph-based methods graph2vec and node2vec is used. By using cosine similarity, Euclidean distance and Pearson similarity algorithms, the most similar ten news for each text-based and graph-based methods are exported. Both Named Entity Graph Model and TF-IDF based Graph

Model for the Turkish and English datasets are used. Examples of news recommendation system for the Turkish and English datasets with Named Entity Graph Model are given in Table 6.1 and 6.2. Also, examples of news recommendation system for the Turkish and English datasets with TF-IDF based Graph Model are given in Table 6.3 and Table 6.4. First news is the one of the news from dataset and others are recommended news by different methods.

Table 6.1: An example of News Recommendation System for English dataset with Named Entity Graph Model

News: ex-boss launches defence lawyers defending former worldcom chief bernie ebbers against a battery of fraud charges have called a company whistleblower as their first witness. ...
Proposed News by doc2vec: director ends evidence the former chief financial officer at us telecoms firm worldcom has finished giving evidence at the trial of his ex-boss bernie ebbers. scott sullivan admitted to jurors he was willing to commit fraud to meet wall street earnings projections. ...
Proposed News by graph2vec: wal-mart is sued over rude lyrics the parents of a 13-year-old girl are suing us supermarket giant wal-mart over a cd by rock group evanescence that contains swear words. ...
Proposed News by node2vec: ampbell: e-mail row silly fuss ex-no 10 media chief alastair campbell is at the centre of a new political row over an e-mail containing a four-letter outburst aimed at bbc journalists. ...
Proposed News by bag-of-words: wal-mart is sued over rude lyrics the parents of a 13-year-old girl are suing us supermarket giant wal-mart over a cd by rock group evanescence that contains swear words. the lawsuit filed in washington county alleges wal-mart deceived customers by not putting warning labels on the cover. ...
Proposed News by Specter: ebbers denies worldcom fraud former worldcom chief bernie ebbers has denied claims that he knew accountants were doctoring the books at the firm. speaking in court mr ebbers rejected allegations he pressured ex-chief financial officer scott sullivan to falsify company financial statements. ...

Table 6.2: An example of News Recommendation System for Turkish dataset with Named Entity Graph Model

News: Dünyada akaryakıtın en ucuz olduğu ülkelerden Türkmenistan'da 2015'in ilk gününde akaryakıta yüzde 60 civarında zam geldi. Daha önce kurşunsuz benzinin litresi 62 tengeden (22 sent) satılırken, yeni yılın ilk gününde kurşunsuz benzinin litresi 1 manat (35 sent) oldu. ...
Proposed News by doc2vec: Elektronik devi Apple kazanç raporunu açıkladı. Rekor "amiral gemi" iPhone satışları Apple'ın tüm beklenti rakamlarını yerle bir etmesini sağladı. Apple'ın 2015 mali yılının ilk çeyreğine ait finansal sonuçlarına göre, şirket 31 Aralık 2014'te biten çeyrekte 74,6 milyar dolar gelir ve 18 milyar dolar net kar elde etti. ...
Proposed News by graph2vec: Petrol fiyatlarındaki sert düşüş benzini de aşağı çekmeye devam ediyor. Dün 45 dolar seviyelerine kadar düşen Brent petrolün varil fiyatından sonra benzine 6 kuruş indirim yapıldı. ...
Proposed News by node2vec: ABD'nin kalbi Beyaz Saray'ın bahçesine düşerek büyük paniğe neden olan hava aracının sarhoş bir Amerikalı istihbarat görevlisine ait olduğu ortaya çıktı. ...
Proposed News by node2vec2: Mısır'ın son dönemdeki en büyük projelerinden Yeni Süveyş Kanalı'nın bir kısmının çöktüğü bildirildi. Emniyet kaynaklarından alınan bilgiye göre, Yeni Süveyş Kanalı'nın doğusunda yapılan çökeltim havuzlarından biri kısmen çöktü. ...
Proposed News by bag-of-words: Elektronik devi Apple kazanç raporunu açıkladı. Rekor "amiral gemi" iPhone satışları Apple'ın tüm beklenti rakamlarını yerle bir etmesini sağladı. Apple'ın 2015 mali yılının ilk çeyreğine ait finansal sonuçlarına göre, şirket 31 Aralık 2014'te biten çeyrekte 74,6 milyar dolar gelir ve 18 milyar dolar net kar elde etti. ...
Proposed News by Specter: Dünyada 65 ülke arasında yapılan araştırmaya göre İngilizce konuşulan ülkeler içinde en iyi eğitimin verildiği Kanada'ya Türkiye'den göç oranı arttı. 1998-2013 tarihleri arasında 6 milyona yakın göçmenlik başvurusuna onay veren Kanada, bu tarihler arasında Türkiye'den de 18 binin üzerinde toplam başvuruyu kabul etti. ...

Table 6.3: An example of News Recommendation System for English dataset with TF-IDF based Graph Model

News: german business confidence slides german business confidence fell in february knocking hopes of a speedy recovery in europe s largest economy. munich-based research institute ifo said that its confidence index fell to 95.5 in february from 97.5 in january its first decline in three months. the study found that the outlook in both the manufacturing and retail sectors had worsened. ...
Proposed News by doc2vec: uk economy facing major risks the uk manufacturing sector will continue to face serious challenges over the next two years the british chamber of commerce (bcc) has said. the group s quarterly survey of companies found exports had picked up in the last three months of 2004 to their best levels in eight years. ...
Proposed News by graph2vec: blair said he would stand down tony blair promised gordon brown he would stand down before the next election a new book about the chancellor claims. but the prime minister changed his mind following intervention from allies in the cabinet according to the book. ...
Proposed News by node2vec: blair told to double overseas aid tony blair is being urged to use all his negotiating powers to end poor countries debt and double aid. some 45 million children will die needlessly before 2015 and aid budgets are half their 1960 levels oxfam says in a report paying the price. ...
Proposed News by bag-of-words: german growth goes into reverse germany s economy shrank 0.2% in the last three months of 2004 upsetting hopes of a sustained recovery. the figures confounded hopes of a 0.2% expansion in the fourth quarter in europe s biggest economy. ...
Proposed News by Specter: german growth goes into reverse germany s economy shrank 0.2% in the last three months of 2004 upsetting hopes of a sustained recovery. the figures confounded hopes of a 0.2% expansion in the fourth quarter in europe s biggest economy. ...

Table 6.4: An example of News Recommendation System for Turkish dataset with TF-IDF based Graph Model

News: Çin'in resmi haber ajansı Şinhua, Şangay'da deniz kenarındaki Bund bölgesinde yapılan yeni yıl kutlamalarında izdiham çıktığını bildirdi. İzdihamda 35 kişinin öldüğü, 42 kişinin yaralandığı belirtildi. ...
Proposed News by doc2vec: Chicago Üniversitesi'nden paleontologlar Neil Shubin ve Edward Daeschler tarafından Ellesmere Adası'nda bulunduktan sonra üzerinde bilimsel çalışmalar başlatılan balık fosiline, Tiktaalik Roseae ismi verilmişti. ...
Proposed News by graph2vec: Reuters, Belçika ajansı Belga'yı kaynak göstererek, Belçika'da bir silahlı saldırı paniği yaşandığını duyurdu. Olayın, başkent Brüksel'in 50 kilometre güneyindeki Chatelineau kasabasında bir süpermarkette yaşandığı belirtildi. ...
Proposed News by node2vec: Türkiye Cumhuriyet Merkez Bankası (TCMB), gıda ürünlerinde istikrarlı arz ve fiyat oluşumunu desteklemek üzere kurulması öngörülen Erken Uyarı Sistemine ilişkin çalışmalarda gelinen aşamanın değerlendirildiğini açıkladı. ...
Proposed News by bag-of-words: Çin'in doğusundaki Şanghay kentinde yeni yıl kutlamalarında çıkan izdihamda hayatını kaybeden 36 kişinin ailelerine yüz otuzar bin dolar tazminat ödeneceği bildirildi. ...
Proposed News by Specter: Malezya'daki Sabak Bernam'dan Endonezya'daki Sumatra limanına hareket eden kaçak göçmenleri taşıyan teknenin batması sonucu 13'ü kadın 14 kişi öldü. ...

In order to construct the ground truth for news recommendation, a survey study is conducted. Six different similarity calculation methods are applied on the news text. Three of them is graph based representations (graph2vec, node2vec, node2vec2), and three of them are text based representations (doc2vec, Specter, BOW). The similarity calculation method, *node2vec2*, is only used for Named Entity Graph Model for Turkish dataset. As it resulted with limited performance in the first survey study, it is not included in the other survey studies with Named Entity Graph model for English dataset and TF-IDF based Graph Model for Turkish and English datasets.

In user study, 5 different surveys were prepared. Each survey contains 20 different questions. Question content is news, choices are most similar other news contents that are recommended by our methods. Each question has maximum 6 choices. Some questions have less than 6 choices since some methods recommend the same news. After that, each survey is filled by 3 users, hence on the total 15 participants took part in the each survey study. For each user, the order of questions and choices are shuffled, in order to prevent any bias. In the surveys, users were told to only choice similar news and users marked the most similar one or more choices. In order to determine the ground truth we assign a score of 1 point to the method if its recommendation is marked by a user. The surveys are conducted for Turkish and English datasets with Named Entity Graph Model and TF-IDF based Graph Model separately. The gathered aggregation counts of each survey and the results are given in Table 6.5.

Table 6.5: Aggregation Count of User Studies for Turkish and English Datasets with Named Entity Graph Model and TF-IDF based Graph Model

		doc2vec	graph2vec	node2vec	BOW	node2vec2	Specter
Named Entity Graph Model	**Turkish Dataset**	135 (%45.00)	166 (%55.33)	25 (%8.33)	128 (%42.67)	13 (%4.34)	143 (%47.67)
	English Dataset	203 (%67.67)	72 (%24.00)	24 (%8.00)	59 (%19.67)	-	191 (%63.67)
TF-IDF based Graph Model	**Turkish Dataset**	126 (%42.00)	164 (%54.67)	23 (%7.67)	126 (%42.00)	-	149 (%49.67)
	English Dataset	169 (%56.34)	78 (%26.00)	50 (%16.67)	151 (%50.34)	-	176 (%58.67)

In order to determine the ground truth with majority voting (rather than aggregating all scores), a scoring mechanism is applied as follows: For a given question, the respond having the highest count is given 1 point. In case of tie, all winners receive 1 point. The results under this majority voting scoring schema are given in Table 6.6.

Table 6.6: Aggregation Count of User Studies by using majority voting for Turkish and English Datasets with Named Entity Graph Model and TF-IDF based Graph Model

		doc2vec	graph2vec	node2vec	BOW	node2vec2	Specter
Named Entity Graph Model	**Turkish Dataset**	33 (%33.00)	46 (%46.00)	4 (%4.00)	35 (%35.00)	0 (%0.00)	42 (%42.00)
	English Dataset	57 (%57.00)	18 (%18.00)	1 (%1.00)	11 (%11.00)	-	47 (%47.00)
TF-IDF based Graph Model	**Turkish Dataset**	29 (%29.00)	49 (%49.00)	1 (%1.00)	35 (%35.00)	-	33 (%33.00)
	English Dataset	42 (%42.00)	23 (%23.00)	5 (%5.00)	37 (%37.00)	-	45 (%45.00)

According to results of the surveys, graph2vec gives more accurate results than other methods for Named Entity Graph Model and TF-IDF based Graph Model for Turkish dataset. Also, graph2vec has the highest aggregation count according to results for majority voting. But doc2vec gives more accurate results than other methods for Named Entity Graph Model for English dataset and Specter gives more accurate results than other methods for TF-IDF based Graph Model for English dataset. Also, doc2vec adn Specter have the highest aggregation count according to results for majority voting. We think that the reason for this contradiction is that the named entity recognition tools are different. The named entity recognition tool we used for the English dataset works for more entity names. So our graph models for English dataset are more messy and contain more nodes and edges. Because of this reason graph based similarity calculation methods gives worse results than text based similarity calculation methods for English dataset.

6.2.2 Experiment 2: News Classification by using News Similarity

In the second experiment, we analyze the performance of similarity based news classification. In this analysis, on the basis of the results of the first experiment, graph2vec is used as the representative for graph-based method and doc2vec, Specter are used as text-based methods. After similarity calculation between graphs, we obtain a list for each news that contains the 10 most similar news ids that are sorted from most similar to least similar. These lists are used in order to classify the news. Also, A hybrid model is constructed by using ranking aggregation method with the graph2vec's doc2vec's and Specter's similarity results. Using different text-based and graph-based embedding methods, we attempted to improve the outcomes of the hybrid model we constructed for news classification.

For the classification, most similar 1, 3, 5, 7, and 10 news ids and their classes are taken. The class label is determined as the majority of the class labels among the most similar news. Finally, we have compared the ground truth and predicted class and obtained an accuracy value. We have calculated the accuracy for each class and for overall. In order to calculate overall accuracy, weighted average is used that is, each category participates in the average as much as the number of news. In accuracy

calculations the label *Other* is ignored, since dataset includes some of these news instances with other labels. Additionally, number of the news are decreased in TF-IDF based Graph Model for the Turkish dataset after applying the TF-IDF filter, so classification accuracy of text-based methods are not same for Named Entity Graph Model and TF-IDF based Graph Model for the Turkish dataset. The results are given in Table 6.7, Table 6.8, Table 6.9 and Table 6.10 .

Table 6.7: Classification Accuracy of graph2vec, doc2vec, Specter and Hybrid Model for the Turkish Dataset with Named Entity Graph Model

Category	**1**				**3**				**5**				**7**				**10**			
	graph2vec	***doc2vec***	***Specter***	***Hybrid Model***	***graph2vec***	***doc2vec***	***Specter***	***Hybrid Model***	***graph2vec***	***doc2vec***	***Specter***	***Hybrid Model***	***graph2vec***	***doc2vec***	***Specter***	***Hybrid Model***	***graph2vec***	***doc2vec***	***Specter***	***Hybrid Model***
Sust. World	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Life	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Textile	0.00	14.58	26.04	9.38	0.00	13.54	20.83	9.38	0.00	8.33	16.67	8.33	0.00	5.21	12.50	9.38	0.00	2.08	10.42	9.38
News from Home	2.44	0.00	4.88	2.44	0.00	2.44	4.88	4.88	0.00	2.44	2.44	2.44	0.00	0.00	2.44	0.00	0.00	0.00	0.00	0.00
Economic Data	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Business world	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Industries	0.00	9.62	7.69	5.77	0.00	7.69	5.77	5.77	0.00	7.69	3.85	3.85	0.00	7.69	1.92	3.85	0.00	9.62	1.92	1.92
Health	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Insurance	0.00	18.94	40.91	23.48	0.00	31.82	45.45	30.30	0.00	35.61	46.21	26.52	0.00	33.33	45.45	28.79	0.00	29.55	40.15	28.79
Logistic	0.00	1.52	4.55	4.55	0.00	0.00	4.55	3.03	0.00	1.52	3.03	1.52	0.00	1.52	3.03	0.00	0.00	1.52	0.00	0.00
World Food	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Sport	0.00	16.67	16.67	16.67	0.00	16.67	8.33	16.67	0.00	0.00	8.33	8.33	0.00	0.00	0.00	8.33	0.00	0.00	0.00	0.00
Companies	1.47	2.94	8.82	5.88	1.47	2.94	8.82	5.88	1.47	1.47	4.41	1.47	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Tourism	1.16	37.44	48.60	24.65	1.16	39.77	58.37	34.42	0.23	40.93	60.23	36.51	0.00	43.72	59.30	38.37	0.00	41.86	56.28	40.23
Real estate	0.93	43.52	41.44	25.46	0.69	49.54	47.69	35.19	0.00	53.47	49.07	35.19	0.00	52.55	46.99	39.58	0.00	52.55	43.06	40.50
Art and culture	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Education	0.00	10.34	31.03	17.24	0.00	3.45	27.59	10.34	0.00	3.45	13.79	6.90	0.00	6.90	10.34	0.00	0.00	3.45	3.45	0.00
Export	0.00	2.86	8.57	11.43	0.00	0.00	11.43	5.71	0.00	0.00	5.71	0.00	0.00	0.00	5.71	0.00	0.00	0.00	2.86	0.00
Technology	0.00	12.69	12.18	7.11	0.51	11.17	10.66	5.58	0.51	9.64	7.11	5.08	0.51	9.64	5.58	5.08	0.00	10.15	5.58	3.55
Automotive	1.69	51.37	50.95	35.31	1.05	56.45	57.93	42.49	0.21	58.14	58.35	44.19	0.00	58.99	56.03	45.67	0.00	57.51	52.43	47.57
Finance	0.00	0.00	9.38	0.00	0.00	0.00	9.38	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
World	46.87	85.64	82.55	60.60	56.24	90.88	86.32	85.91	65.19	92.50	87.96	91.28	70.05	93.16	88.59	93.55	74.70	93.62	89.01	94.92
Energy	1.43	38.02	34.00	20.23	1.15	42.32	39.17	28.12	0.14	42.32	41.18	29.84	0.00	41.46	40.75	30.85	0.00	41.61	39.17	29.70
Economy	18.26	49.50	46.98	34.65	16.77	60.52	57.94	47.32	13.38	64.60	60.61	52.64	11.15	66.78	62.72	56.44	8.48	68.72	63.94	59.13
Agenda	29.46	76.30	77.16	49.76	32.02	78.19	80.30	75.79	33.31	78.53	81.00	79.17	33.61	78.49	81.41	80.04	33.91	77.75	81.55	81.02
Agriculture	2.52	48.91	49.48	27.38	1.60	60.60	55.56	39.63	0.57	65.18	56.13	45.13	0.46	68.84	55.21	48.45	0.23	68.73	53.95	50.97
Average	30.42	67.13	65.76	46.12	34.84	72.30	70.50	66.03	38.44	73.96	71.86	70.22	40.27	74.70	72.42	72.25	41.95	74.99	72.58	73.61

Table 6.8: Classification Accuracy of graph2vec, doc2vec, Specter and Hybrid Model for the English Dataset with Named Entity Graph Model

Category	**1**				**3**				**5**				**7**				**10**			
	graph2vec	***doc2vec***	***Specter***	***Hybrid Model***	***graph2vec***	***doc2vec***	***Specter***	***Hybrid Model***	***graph2vec***	***doc2vec***	***Specter***	***Hybrid Model***	***graph2vec***	***doc2vec***	***Specter***	***Hybrid Model***	***graph2vec***	***doc2vec***	***Specter***	***Hybrid Model***
Politics	24.09	93.43	92.34	80.29	22.62	93.80	94.89	93.07	21.53	94.89	93.06	94.16	19.71	94.89	92.70	93.80	18.61	94.53	93.07	93.80
Sport	28.03	99.71	97.11	81.21	28.03	99.71	95.95	98.55	28.03	99.71	96.82	99.13	29.77	99.71	96.82	99.42	30.06	99.71	96.82	99.42
Entertainment	16.12	92.31	86.45	71.79	17.95	94.51	89.74	89.01	20.51	95.24	90.48	90.84	19.41	95.24	90.84	92.67	16.48	95.97	88.28	94.14
Business	26.19	92.86	88.69	76.49	29.46	93.45	91.67	90.18	32.14	93.15	91.37	91.96	31.85	93.15	91.07	92.26	30.06	93.75	90.77	93.45
Technology	24.52	96.55	86.97	85.06	22.61	96.93	89.27	89.66	21.84	95.02	90.42	91.95	19.92	95.02	90.04	94.25	19.54	95.02	88.51	95.02
Average	24.09	95.10	90.60	78.93	24.56	95.77	92.48	92.35	25.30	95.70	92.62	93.83	24.77	95.70	92.48	94.63	23.62	95.91	91.74	95.30

Table 6.9: Classification Accuracy of graph2vec, doc2vec, Specter and Hybrid Model for the Turkish Dataset with TF-IDF based Graph Model

Category	1				3				5				7				10			
	graph2vec	*doc2vec*	*Specter*	*Hybrid Model*	*graph2vec*	*doc2vec*	*Specter*	*Hybrid Model*	*graph2vec*	*doc2vec*	*Specter*	*Hybrid Model*	*graph2vec*	*doc2vec*	*Specter*	*Hybrid Model*	*graph2vec*	*doc2vec*	*Specter*	*Hybrid Model*
Sust. World	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Life	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Textile	20.83	10.42	27.08	29.17	17.71	12.50	20.83	26.04	14.58	10.42	17.71	19.79	12.50	6.25	13.54	12.50	12.50	4.17	8.33	8.33
News from Home	0.00	2.44	4.88	0.00	0.00	4.88	4.88	0.00	0.00	0.00	2.44	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Economic Data	0.00	0.00	0.00	16.67	0.00	0.00	0.00	16.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Business world	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Industries	1.92	5.77	7.69	5.77	1.92	5.77	5.77	7.69	0.00	7.69	3.85	5.77	0.00	7.69	0.00	1.92	0.00	7.69	0.00	1.92
Health	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Insurance	38.81	32.84	41.79	40.30	43.28	40.30	47.76	50.00	45.52	29.85	46.27	52.99	42.54	29.85	46.27	48.51	41.04	22.39	41.79	48.51
Logistic	9.09	7.58	4.55	7.58	9.09	7.58	4.55	6.06	10.61	0.00	3.03	1.52	4.55	1.52	3.03	0.00	1.52	0.00	0.00	1.52
World Food	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10
Sport	33.33	16.67	16.67	33.33	25.00	16.67	8.33	33.33	25.00	0.00	8.33	33.33	25.00	0.00	0.00	16.67	0.00	0.00	0.00	16.67
Companies	4.29	1.43	8.57	10.00	1.43	0.00	10.00	5.71	0.00	0.00	2.86	2.86	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Tourism	39.91	39.20	47.65	48.59	44.60	47.89	58.45	53.05	46.48	47.89	60.80	55.63	48.59	47.42	59.86	56.34	46.71	44.37	57.75	55.87
Real estate	46.39	38.23	39.39	47.79	52.44	45.45	44.76	51.28	56.64	44.06	48.02	55.24	54.55	42.19	45.45	55.24	54.78	43.12	43.82	54.55
Art and culture	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Education	10.00	13.33	30.00	20.00	6.67	13.33	23.33	16.67	6.67	10.00	3.33	3.33	6.67	0.00	3.33	3.33	6.67	0.00	3.33	0.00
Export	11.11	5.56	8.33	11.11	5.55	0.00	8.33	5.56	0.00	0.00	5.56	0.00	0.00	0.00	2.78	0.00	0.00	0.00	2.78	0.00
Technology	11.22	11.71	10.73	17.56	7.80	10.24	9.76	11.22	7.80	10.24	7.80	9.27	8.29	9.76	5.85	7.80	6.83	7.32	4.88	4.39
Automotive	53.80	47.94	51.41	55.31	59.44	52.06	58.57	59.87	61.82	55.53	59.87	62.69	64.86	57.05	56.18	64.21	64.86	56.18	54.45	65.08
Finance	0.00	0.00	9.09	0.00	0.00	0.00	9.09	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
World	86.40	85.87	81.66	87.47	91.16	91.07	85.47	92.78	92.55	92.58	87.17	94.67	93.13	93.68	87.76	95.18	93.65	94.12	88.26	95.70
Energy	34.75	35.88	34.60	37.85	41.38	42.09	39.12	45.20	40.96	44.63	42.23	43.93	40.96	42.94	41.67	44.49	40.11	41.67	39.69	41.81
Economy	48.85	49.79	46.97	50.68	58.48	61.26	57.06	62.29	61.13	65.90	60.53	66.43	62.11	67.29	62.82	68.29	62.91	69.12	64.31	70.28
Agenda	75.56	77.78	77.07	78.99	77.91	79.96	80.40	82.45	78.15	80.05	81.49	82.69	78.08	79.66	82.12	83.05	77.67	79.28	82.52	83.24
Agriculture	55.11	48.98	48.07	56.82	66.14	62.39	54.66	66.59	70.34	66.93	55.68	70.45	71.48	69.55	54.43	72.05	73.41	72.39	53.64	73.41
Average	67.20	67.22	65.03	69.12	72.03	72.67	69.70	74.71	73.34	74.24	71.37	76.41	73.78	74.86	71.95	77.03	74.03	75.22	72.31	77.57

Table 6.10: Classification Accuracy of graph2vec, doc2vec, Specter and Hybrid Model for the English Dataset with TF-IDF based Graph Model

Category	1				3				5				7				10			
	graph2vec	*doc2vec*	*Specter*	*Hybrid Model*	*graph2vec*	*doc2vec*	*Specter*	*Hybrid Model*	*graph2vec*	*doc2vec*	*Specter*	*Hybrid Model*	*graph2vec*	*doc2vec*	*Specter*	*Hybrid Model*	*graph2vec*	*doc2vec*	*Specter*	*Hybrid Model*
Politics	18.61	93.43	92.34	77.74	16.42	93.80	94.89	92.70	14.96	94.89	93.06	94.89	14.23	94.89	92.70	94.53	13.50	94.53	93.07	95.99
Sport	30.64	99.71	97.11	78.61	31.21	99.71	95.95	96.82	28.03	99.71	96.82	98.55	28.32	99.71	96.82	98.27	27.46	99.71	96.82	98.84
Entertainment	20.15	92.31	86.45	69.23	15.75	94.51	89.74	88.64	14.65	95.24	90.48	90.84	14.65	95.24	90.84	93.41	16.85	95.97	88.28	94.87
Business	17.56	92.86	88.69	78.87	15.77	93.45	91.67	91.07	12.20	93.15	91.37	91.67	13.99	93.15	91.07	93.15	17.26	93.75	90.77	92.56
Technology	36.78	96.55	86.97	86.21	34.10	96.93	89.27	93.10	41.38	95.02	90.42	94.25	41.76	95.02	90.04	95.79	46.36	95.02	88.51	96.93
Average	24.63	95.10	90.60	78.12	22.68	95.77	92.48	92.62	21.95	95.70	92.62	94.16	22.35	95.70	92.48	95.10	23.96	95.91	91.74	95.84

According to results, doc2vec, Specter and hybrid model give significantly more accurate results compared to graph2vec for news classification task for the Turkish and English datasets with Named Entity Graph Model and the English dataset with TF-IDF based Graph Model. But TF-IDF based Graph Model for the Turkish dataset, graph2vec, doc2vec, Specter and hybrid model gives almost same results for news classification. Even, the hybrid model gives the most accurate result in TF-IDF based Graph Model for the Turkish dataset.

In the Named Entity Graph Model, we have only used entities as nodes in the graph structure and entities are certain. This graph model is suitable for news recommendation since in the recommendation system, we propose the related news with content. For example, same locations or organizations can be used in different two news and these two nodes can be similar with their content. But their categories can be different. Because of this reason, graph based embedding methods gives better results for news recommendation and worse results for news classification. In TF-IDF based Graph Model, we use the terms that are frequently used in the news as nodes in graph structure. This graph model is more suitable for new classification, since repetitive words make it easier to classify the news. For the sake of the self containment of our explanation, category matching accuracy of user surveys that are done in section 6.2.1 are exported and the accuracy results are given in Table 6.11. While calculating accuracies, the results of the surveys by using majority voting technique are used.

Table 6.11: Category Matching Accuracy of User Surveys for the Turkish and English Datasets with Named Entity Graph Model and TF-IDF based Graph Model

		graph2vec	**doc2vec**	**Specter**	**Average**
Named Entity Graph Model	**Turkish Dataset**	%63.830	%75.000	%75.000	%73.118
	English Dataset	%50.000	%98.305	%89.362	%88.660
TF-IDF based Graph Model	**Turkish Dataset**	%65.306	%79.310	%73.684	%75.510
	English Dataset	%47.368	%96.970	%95.000	%93.827

According to results in Table 6.11, even though graph2vec has the highest aggregation count for news recommendation with the Turkish dataset, category matching accuracy is lower. graph2vec has the lowest aggregation count for news recommendation and the lowest category matching accuracy with the English dataset, too. On the basis of the presented results, it is seen that, graph2vec is not successfull compared to doc2vec and Specter for news classification.

6.2.3 Experiment 3: News Classification with Different Classifiers

Because a labelled dataset including text items and their labels is used to train a classifier, text classification is an example of supervised machine learning problem. Three primary components make up an end-to-end text categorization pipeline:

- **Data Preparation:** In this experiment, the Turkish dataset is used for news classification. Three different data that contains the textual content and label were prepared. In the first data, only title of the news and their labels were used. We mentioned about that scenario as 'title' below. In the second data, the whole content of the news and their labels were used. We mentioned about that scenario as 'content' below. And in the final data, the lemmatized news content and their labels were used. We mentioned about that scenario as 'lemmatized' below.

  In the experiment, data splits into two parts randomly. First part is training part and it contains the %90 of the whole data. Second part is test part and it contains the %10 of the whole data.

- **Feature Engineering:** Textual data is transformed into feature vectors and new features are created using the existing dataset. Different ideas were implemented in order to obtain features from our dataset. First of all, count vectors were created as features. The term 'Count Vector' refers to a matrix representation of the dataset in which each row represents a document from the corpus, each column represents a term from the corpus, and each cell represents the frequency count of a specific word in a certain document. Secondly, TF-IDF vectors were created as features. The TF-IDF score indicates how important a phrase is in the document and throughout the corpus. Different levels of input tokens were created: Word Level TF-IDF, N-gram Level TF-IDF, and Character Level TF-IDF. In Word Level TF-IDF, matrix represents TF-IDF scores of every term in different documents. In N-gram Level TF-IDF, matrix represents TF-IDF scores of N-grams that are the combination of N terms together. In Character Level TF-IDF, matrix represents TF-IDF scores of character level n-grams in the corpus.

- **Model Training:** The final stage in the text classification framework is to use the features produced in the previous phase to train a classifier. Machine learning models come in a variety of shapes and sizes, and they may all be utilized to train a final model. In our experiment, we used four different models: Naive Bayes Classifier, Linear Classifier, SVM Classifier, and Random Forest Classifier.

By using the models, feature vectors, and data that are mentioned above, we have calculated the accuracy of the each model with different feature vectors and data by using the 10-fold cross validation method. The final results are given Table 6.12.

Table 6.12: Accuracy of Naive Bayes Classifier, Linear Classifer, SVM Classifier, Random Forest Classifier with different feature vectors and data

		title	**content**	**lemmatized**
Naive Bayes Classifier	**Count Vectors**	53.506	58.912	54.174
	Word Level TF-IDF	53.932	59.076	58.232
	N-Gram Vectors	49.410	57.569	57.869
	Char Level Vectors	50.491	56.979	55.328
Linear Classifier	**Count Vectors**	52.851	59.174	56.252
	Word Level TF-IDF	43.473	60.387	59.419
	N-Gram Vectors	49.476	59.109	58.792
	Char Level Vectors	52.195	60.092	58.562
SVM Classifier	**Count Vectors**	54.751	58.355	58.726
	Word Level TF-IDF	54.554	58.814	59.353
	N-Gram Vectors	50.098	58.224	59.320
	Char Level Vectors	54.948	58.814	59.419
Random Forest Classifier	**Count Vectors**	52.163	56.717	58.265
	Word Level TF-IDF	51.343	57.831	59.023
	N-Gram Vectors	48.263	55.996	57.473
	Char Level Vectors	52.326	57.241	58.198

According to results, by using the title of the news are used for classification SVM classifier with char level vector as feature vector has the highest accuracy with the %54.948 . By using the content of the news are used for classification Linear classifier with char level vector as feature vector has the highest accuracy with the %60.092 . By using the lemmatized content of the news are used for classification SVM classifier with char level vector as feature vector and Linear classifier with word level TF-IDF as feature vector have the highest accuracies with the %59.419 . If we look at all the results the highest accuracy is %60.092. This value is smaller than the results in section 6.2.2, that is, news classification by using news recommendation system give more accurate results for news classification.

CHAPTER 7

CONCLUSIONS

In this thesis study, it is hypothesized that graph based text representation will yield more accurate results than text-based methods for text mining tasks. In particular, the study focuses on news texts and compares graph and text embedding methods for news recommendation and news classification tasks. The experiments are conducted on two datasets including Turkish and English news collections.

By using two different datasets (the Turkish and English datasets) we created graph models from news text. In our study, we proposed 2 different graph models: First one is *Named Entity Graph Model* and second one is *TF-IDF based Graph Model*. These two graph models are compared by doing experiments. For the news recommendation task, both graph models are successful but for news classification task TF-IDF based Graph Model is more accurate.

Additionally, we extracted feature vector representation of news by using graph and text embedding techniques. After that, cosine similarity, Euclidean distance and Pearson similarity algorithms were applied in order to find similarity between the news by using feature vectors. In order to measure the accuracy of news recommendation task, user surveys were conducted and answer of the surveys were collected from the users. For news classification task, similarity calculation results were used. Also, a hybrid model that is created by using ranking aggregation of graph2vec's, doc2vec's and Specter's similarity results is proposed for news classification.

Experimental results show that graph based methods are more successful than text based methods in news recommendation for the Turkish dataset with both Named Entity Graph Model and TF-IDF based Graph Model. However, for the English dataset,

text based representation methods provide more accurate results. This difference can be due to the use of different named entity recognizer tools for English and Turkish texts. Additionally, the differences in the nature of the languages can affect the named entity recognition performance. For the news classification task, text based methods are more accurate for both the Turkish and English datasets with Named Entity Graph Model. On the other hand, for the Turkish dataset with the TF-IDF based Graph Model, graph based and text based methods gives similar performance for news classification task. Also, the proposed hybrid model has the highest accuracy for the TF-IDF based Graph Model on the Turkish dataset.

As the future work, this study can be extended with new graph models in order to keep more accurate information about news. Besides, more efficient similarity calculation methods can be used and developed as time efficiency of similarity calculation phase becomes a bottleneck under large data collections. Also, different ranking aggregation methods can be used in order to combine different similarity calculation methods. Another future work dimension is to increase the size of the ground truth set by extending the user surveys.

REFERENCES

[1] B. YILMAZ, H. Genc, M. Agriman, B. K. Demirdover, M. Erdemir, G. Simsek, and P. Karagoz, "Recent trends in the use of graph neural network models for natural language processing," in *Deep Learning Techniques and Optimization Strategies in Big Data Analytics*, pp. 274–289, IGI Global, 2020.

[2] F. Rousseau, E. Kiagias, and M. Vazirgiannis, "Text categorization as a graph classification problem," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1702–1712, 2015.

[3] X. Yan and J. Han, "gspan: Graph-based substructure pattern mining," in *2002 IEEE International Conference on Data Mining, 2002. Proceedings.*, pp. 721–724, IEEE, 2002.

[4] H. Genc and B. Yilmaz, "Text-based event detection: Deciphering date information using graph embeddings," in *International Conference on Big Data Analytics and Knowledge Discovery*, pp. 266–278, Springer, 2019.

[5] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855–864, 2016.

[6] Z. Huang, W. Chung, and H. Chen, "A graph model for e-commerce recommender systems," *Journal of the American Society for information science and technology*, vol. 55, no. 3, pp. 259–274, 2004.

[7] A. Schenker, M. Last, H. Bunke, and A. Kandel, "Clustering of web documents using a graph model," in *Web Document Analysis: Challenges and Opportunities*, pp. 3–18, World Scientific, 2003.

[8] H. Cai, V. W. Zheng, and K. C.-C. Chang, "A comprehensive survey of graph

embedding: Problems, techniques, and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 9, pp. 1616–1637, 2018.

[9] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: Methods and applications," *arXiv preprint arXiv:1709.05584*, 2017.

[10] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering.," in *Nips*, vol. 14, pp. 585–591, 2001.

[11] A. Ahmed, N. Shervashidze, S. Narayanamurthy, V. Josifovski, and A. J. Smola, "Distributed large-scale natural graph factorization," in *Proceedings of the 22nd international conference on World Wide Web*, pp. 37–48, 2013.

[12] S. Cao, W. Lu, and Q. Xu, "Grarep: Learning graph representations with global structural information," in *Proceedings of the 24th ACM international on conference on information and knowledge management*, pp. 891–900, 2015.

[13] M. Ou, P. Cui, J. Pei, Z. Zhang, and W. Zhu, "Asymmetric transitivity preserving graph embedding," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1105–1114, 2016.

[14] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, 1964.

[15] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 701–710, 2014.

[16] B. Adhikari, Y. Zhang, N. Ramakrishnan, and B. A. Prakash, "Sub2vec: Feature learning for subgraphs," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 170–182, Springer, 2018.

[17] A. Narayanan, M. Chandramohan, R. Venkatesan, L. Chen, Y. Liu, and S. Jaiswal, "graph2vec: Learning distributed representations of graphs," *arXiv preprint arXiv:1707.05005*, 2017.

[18] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International conference on machine learning*, pp. 1188–1196, 2014.

[19] K. W. Church, "Word2vec," *Natural Language Engineering*, vol. 23, no. 1, pp. 155–162, 2017.

[20] H. Tan, J. Guo, and Y. Li, "E-learning recommendation system," in *2008 International conference on computer science and software engineering*, vol. 5, pp. 430–433, IEEE, 2008.

[21] A. Pongpech, M. E. Orlowska, and S. W. Sadiq, "Personalized courses recommendation functionality for flex-el," in *Seventh IEEE International Conference on Advanced Learning Technologies (ICALT 2007)*, pp. 631–633, IEEE, 2007.

[22] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Analysis of recommendation algorithms for e-commerce," in *Proceedings of the 2nd ACM Conference on Electronic Commerce*, pp. 158–167, 2000.

[23] W. S. Lee, "Online clustering for collaborative filtering," 2000.

[24] J. B. Schafer, J. A. Konstan, and J. Riedl, "E-commerce recommendation applications," *Data mining and knowledge discovery*, vol. 5, no. 1, pp. 115–153, 2001.

[25] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Application of dimensionality reduction in recommender system-a case study," tech. rep., Minnesota Univ Minneapolis Dept of Computer Science, 2000.

[26] J. L. Herlocker, *Understanding and improving automated collaborative filtering systems*. University of Minnesota, 2000.

[27] C. C. Aggarwal, J. L. Wolf, K.-L. Wu, and P. S. Yu, "Horting hatches an egg: A new graph-theoretic approach to collaborative filtering," in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 201–212, 1999.

[28] C. Kim and J. Kim, "A recommendation algorithm using multi-level association rules," in *Proceedings IEEE/WIC International Conference on Web Intelligence (WI 2003)*, pp. 524–527, IEEE, 2003.

[29] L. Li, D. Wang, T. Li, D. Knox, and B. Padmanabhan, "Scene: a scalable two-stage personalized news recommendation system," in *Proceedings of the 34th*

international ACM SIGIR conference on Research and development in Information Retrieval, pp. 125–134, 2011.

[30] J. Liu, P. Dolan, and E. R. Pedersen, "Personalized news recommendation based on click behavior," in *Proceedings of the 15th international conference on Intelligent user interfaces*, pp. 31–40, 2010.

[31] B. Mohit, "Named entity recognition," in *Natural language processing of semitic languages*, pp. 221–245, Springer, 2014.

[32] R. F. Cekınel, M. Ağriman, P. Karagöz, and B. Yilmaz, "Named entity recognition with conditional random fields on turkish news dataset: Revisiting the features," in *2019 27th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, IEEE, 2019.

[33] J. Ramos *et al.*, "Using tf-idf to determine word relevance in document queries," in *Proceedings of the first instructional conference on machine learning*, vol. 242, pp. 29–48, Citeseer, 2003.

[34] H. V. Nguyen and L. Bai, "Cosine similarity metric learning for face verification," in *Asian conference on computer vision*, pp. 709–720, Springer, 2010.

[35] Q. Li, Y. Zheng, X. Xie, Y. Chen, W. Liu, and W.-Y. Ma, "Mining user similarity based on location history," in *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, pp. 1–10, 2008.

[36] P.-E. Danielsson, "Euclidean distance mapping," *Computer Graphics and image processing*, vol. 14, no. 3, pp. 227–248, 1980.

[37] S. Niwattanakul, J. Singthongchai, E. Naenudorn, and S. Wanapu, "Using of jaccard coefficient for keywords similarity," in *Proceedings of the international multiconference of engineers and computer scientists*, vol. 1, pp. 380–384, 2013.

[38] I. Rish *et al.*, "An empirical study of the naive bayes classifier," in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, pp. 41–46, 2001.

[39] M. Ghazanfar and A. Prugel-Bennett, "An improved switching hybrid recommender system using naive bayes classifier and collaborative filtering," 2010.

[40] D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant, *Applied logistic regression*, vol. 398. John Wiley & Sons, 2013.

[41] W. S. Noble, "What is a support vector machine?," *Nature biotechnology*, vol. 24, no. 12, pp. 1565–1567, 2006.

[42] M. Pal, "Random forest classifier for remote sensing classification," *International journal of remote sensing*, vol. 26, no. 1, pp. 217–222, 2005.

[43] G. Tür, D. Hakkani-Tür, and K. Oflazer, "A statistical information extraction system for turkish," *Natural Language Engineering*, vol. 9, no. 2, pp. 181–210, 2003.

[44] G. A. Seker and G. Eryigit, "Extending a crf-based named entity recognition model for turkish well formed text and user generated content.," *Semantic Web*, vol. 8, no. 5, pp. 625–642, 2017.

[45] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, "Rank aggregation methods for the web," in *Proceedings of the 10th international conference on World Wide Web*, pp. 613–622, 2001.

[46] G.-w. You and S.-w. Hwang, "Search structures and algorithms for personalized ranking," *Information Sciences*, vol. 178, no. 20, pp. 3925–3942, 2008.

[47] A. Mandal, V. Agrahari, V. Khurana, D. Pal, and A. K. Mitra, "Transporter effects on cell permeability in drug delivery," *Expert opinion on drug delivery*, vol. 14, no. 3, pp. 385–401, 2017.

[48] H. Yin, J. Lan, and X. R. Li, "Measures for ranking estimation performance based on single or multiple performance metrics," in *Proceedings of the 16th International Conference on Information Fusion*, pp. 453–460, IEEE, 2013.

[49] Y.-L. Chen and L.-C. Cheng, "A novel collaborative filtering approach for recommending ranked items," *Expert systems with applications*, vol. 34, no. 4, pp. 2396–2405, 2008.

[50] J. Ding, D. Han, J. Dezert, and Y. Yang, "A new hierarchical ranking aggregation method," *Information Sciences*, vol. 453, pp. 168–185, 2018.

[51] J. d. Borda, "Mémoire sur les élections au scrutin," *Histoire de l'Academie Royale des Sciences pour 1781 (Paris, 1784)*, 1784.

[52] C. M. Burnett and V. Kogan, "Ballot (and voter)"exhaustion" under instant runoff voting: An examination of four ranked-choice elections," *Electoral Studies*, vol. 37, pp. 41–49, 2015.

[53] Y. Goldberg, "Neural network methods for natural language processing," *Synthesis lectures on human language technologies*, vol. 10, no. 1, pp. 1–309, 2017.

[54] A. Cohan, S. Feldman, I. Beltagy, D. Downey, and D. S. Weld, "Specter: Document-level representation learning using citation-informed transformers," *arXiv preprint arXiv:2004.07180*, 2020.

[55] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

APPENDIX A

SAMPLE SURVEY QUESTIONS

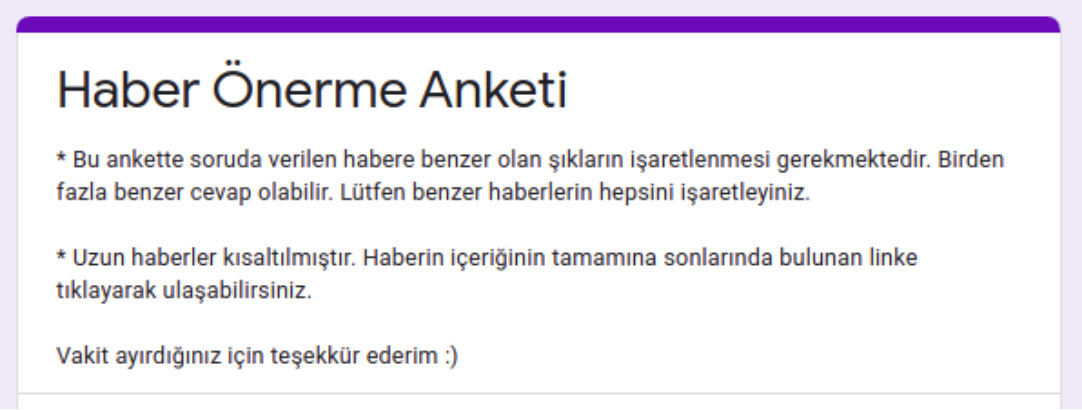

Figure A.1: Information section of the user study

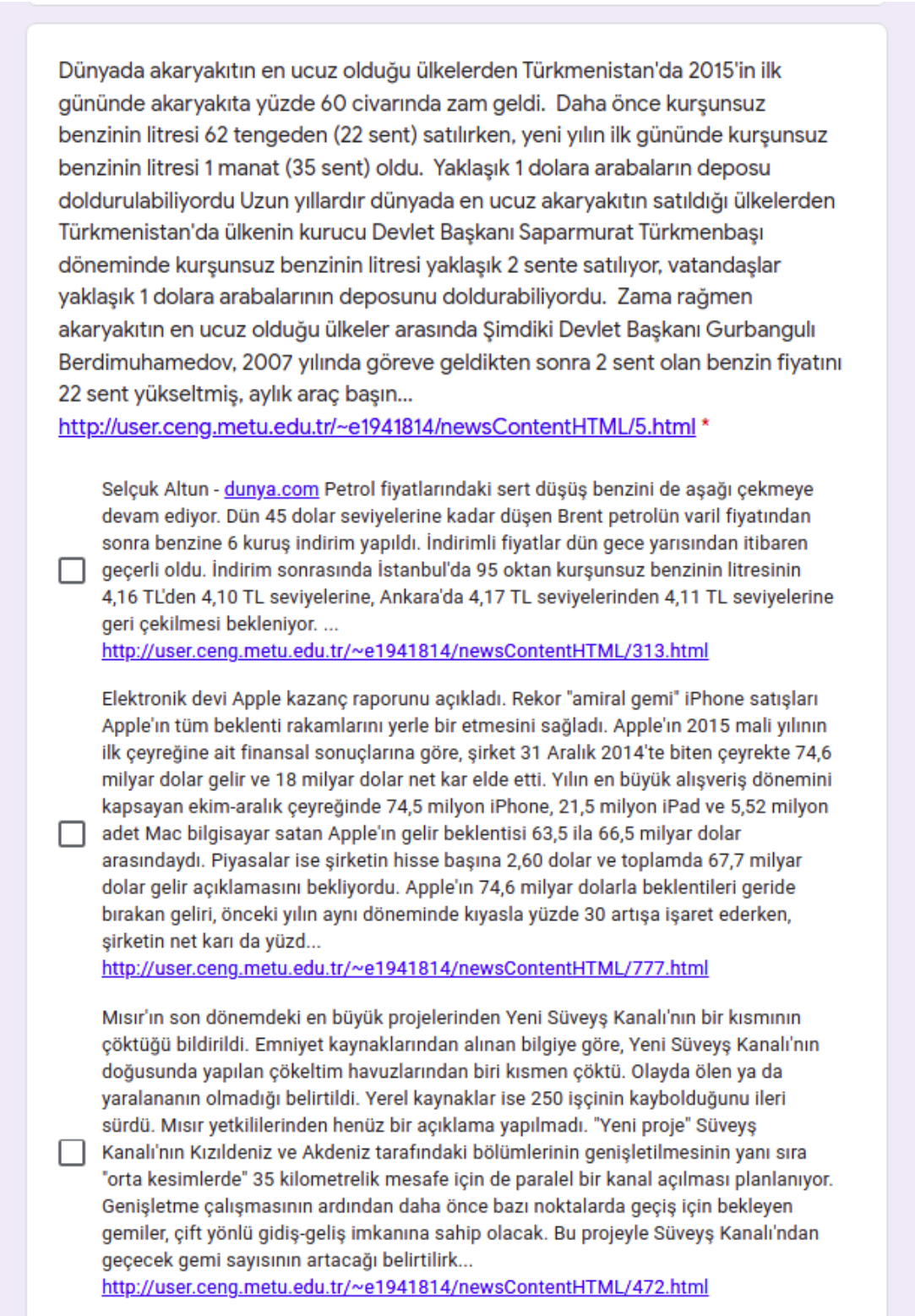

Figure A.2: A question with choices from user study for Turkish dataset

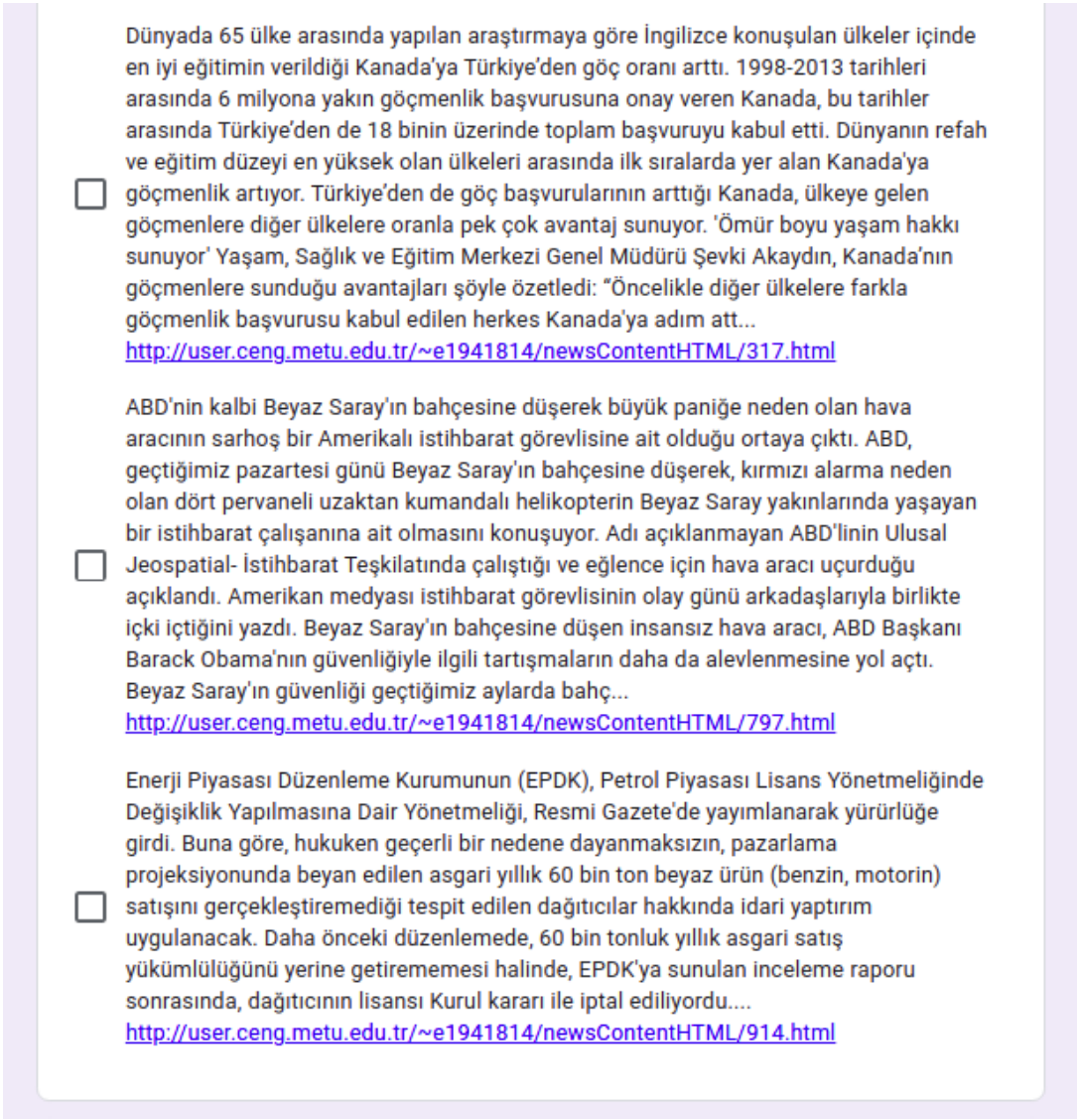

Figure A.3: Continuation of choices of the question in Figure A.2

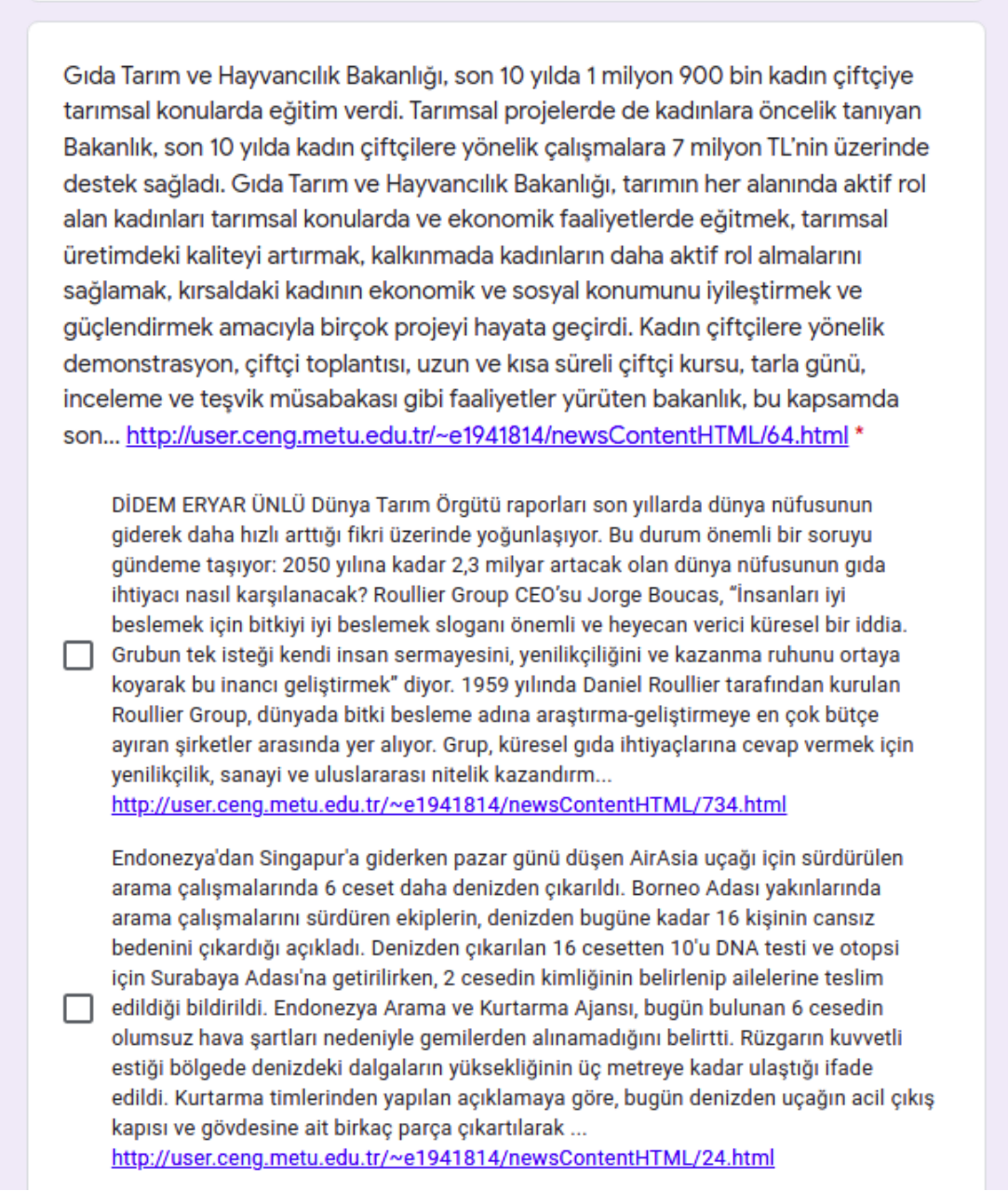

Figure A.4: A question with choices from user study for Turkish dataset

YENER KARADENİZ BAKÜ - Coca Cola, bu yıl ilki Azerbaycan'ın Bakü kentinde düzenlenecek olan 2015 Avrupa Olimpiyat Oyunları'nın 7 ana sponsorundan biri oldu. Haziran'da açılışı yapılacak olan oyunlar için Azerbaycan Hükümeti, 950 milyon Manat (Yaklaşık 2.8 milyar TL) yatırım yaptı. Sponsorluk anlaşması imza töreni öncesi Türk basın mensupları ile bir araya gelen Coca Cola Türkiye, Kafkasya ve Orta Asya Bölümü Başkanı Galya Frayman Molinas, 129'uncu yılını dolduran Coca Cola'nın 200 ülkede faaliyet gösterdiğini anlattı. Bu ülkelerde 500 marka ve 3 bin 500 ürün ile hizmet verdiklerini dile getiren Molinas, yine bu markalardan 17'sinin her birinin perakende değerinin 1 milyar doları aştığını söyledi. Galya Frayman Molinas'ın verdiği bilgilere gore 200 ülkede 700 bin çalışanı ile faaliyette bu...
http://user.ceng.metu.edu.tr/~e1941814/newsContentHTML/471.html

Gıda Tarım ve Hayvancılık Bakanlığı, Avrupa Birliği Katılım Öncesi Yardım Aracı Kırsal Kalkınma Programı (IPARD) kapsamında son 5 yılda 600'den fazla kadın girişimciye destek vererek, söz konusu projeler için 80 milyon liralık ödeme gerçekleştirdi. Bakanlıktan yapılan yazılı açıklamaya göre, IPARD kapsamında kadın girişimcilere verilen hibe desteklerle kadınların çalışma hayatına girmesini sağladı. Bakanlık, başvuru sahibinin kadın olması, işletmede çalışanların en az yüzde 50'sinin kadınlardan oluşması ya da yatırımın bir kadın girişimci tarafından uygulanması gibi özellikler taşıyan projelere öncelik verdi. Bu kapsamda son 5 yılda 600'den fazla kadın girişimcinin projesine destek verilerek, 80 milyon lira ödeme gerçekleştirildi. Tarım ve Kırsal Kalkınmayı Destekleme Kurumu'nun (TKDK)...
http://user.ceng.metu.edu.tr/~e1941814/newsContentHTML/239.html

Almanya'da Dresden Savcılığının, İslam ve göçmen karşıtı Batı'nın İslamlaşmasına Karşı Vatansever Avrupalılar (PEGIDA) derneğinin Başkanı Lutz Bachmann hakkında, "halkı kışkırtma" iddiasıyla soruşturma başlattığı bildirildi. Dresden Başsavcısı Jan Hille, yaptığı açıklamada, sosyal paylaşım sitesi Facebook'tan paylaştığı fotoğraflar ve ifadelerden dolayı Bachmann hakkında "halkı kışkırtma" iddiasıyla soruşturma başlatıldığını belirtti. Hille, söz konusu paylaşımların gerçek olup olmadığının inceleneceğini kaydetti. Öte yandan Bachmann'ın bugün PEGIDA'daki tüm görevlerinden istifa ettiği bildirildi. Bild gazetesine açıklama yapan Bachmann, "Evet yönetimden de ayrılıyorum" ifadesini kullandı. PEGIDA Sözcüsü Kathrin Oertel, Spiegel dergisine yaptığı açıklamada, PEGIDA organizasyonunda yer...
http://user.ceng.metu.edu.tr/~e1941814/newsContentHTML/583.html

Figure A.5: Continuation of choices of the question in Figure A.4

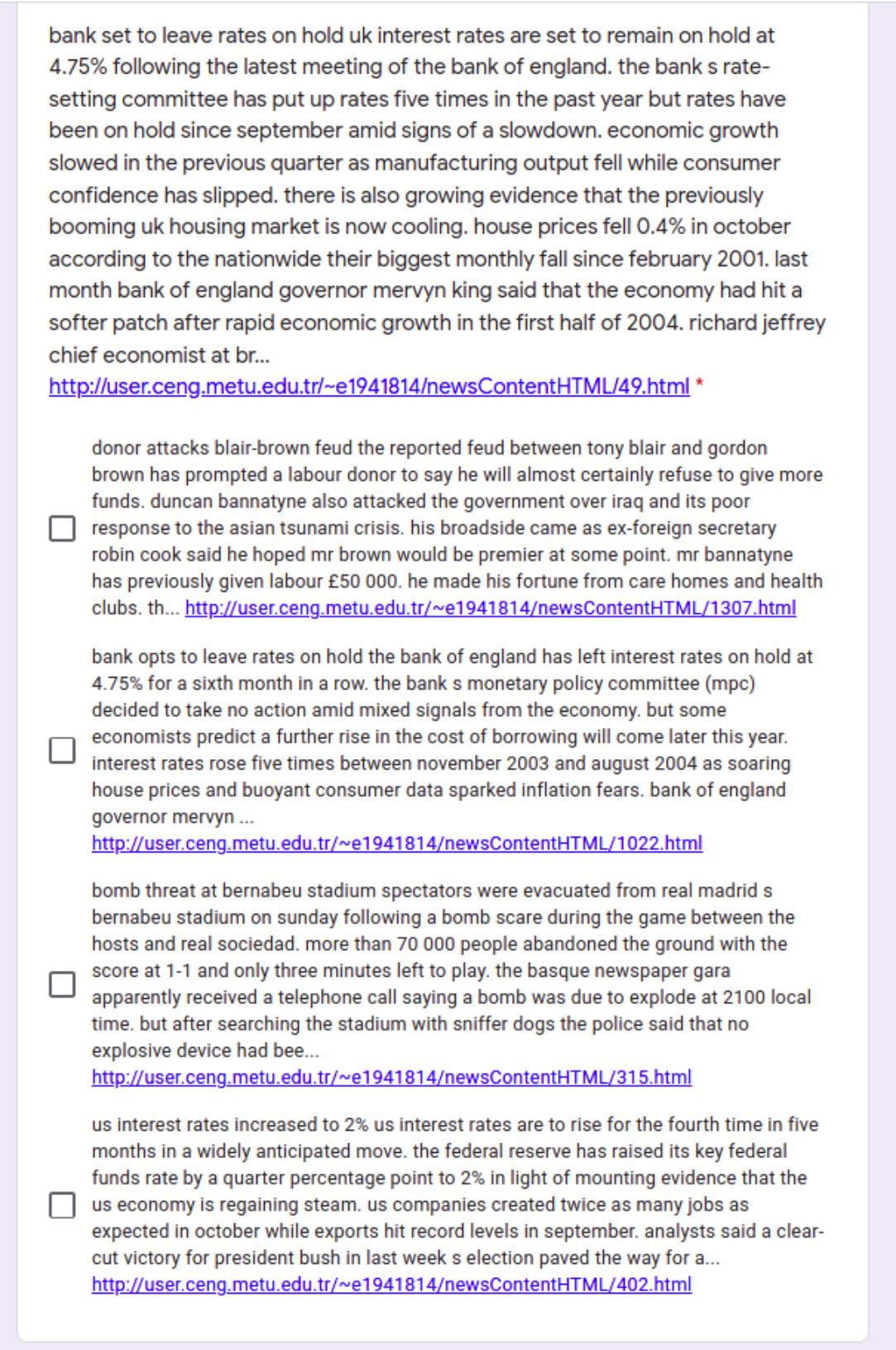

Figure A.6: A question with choices from user study for English dataset

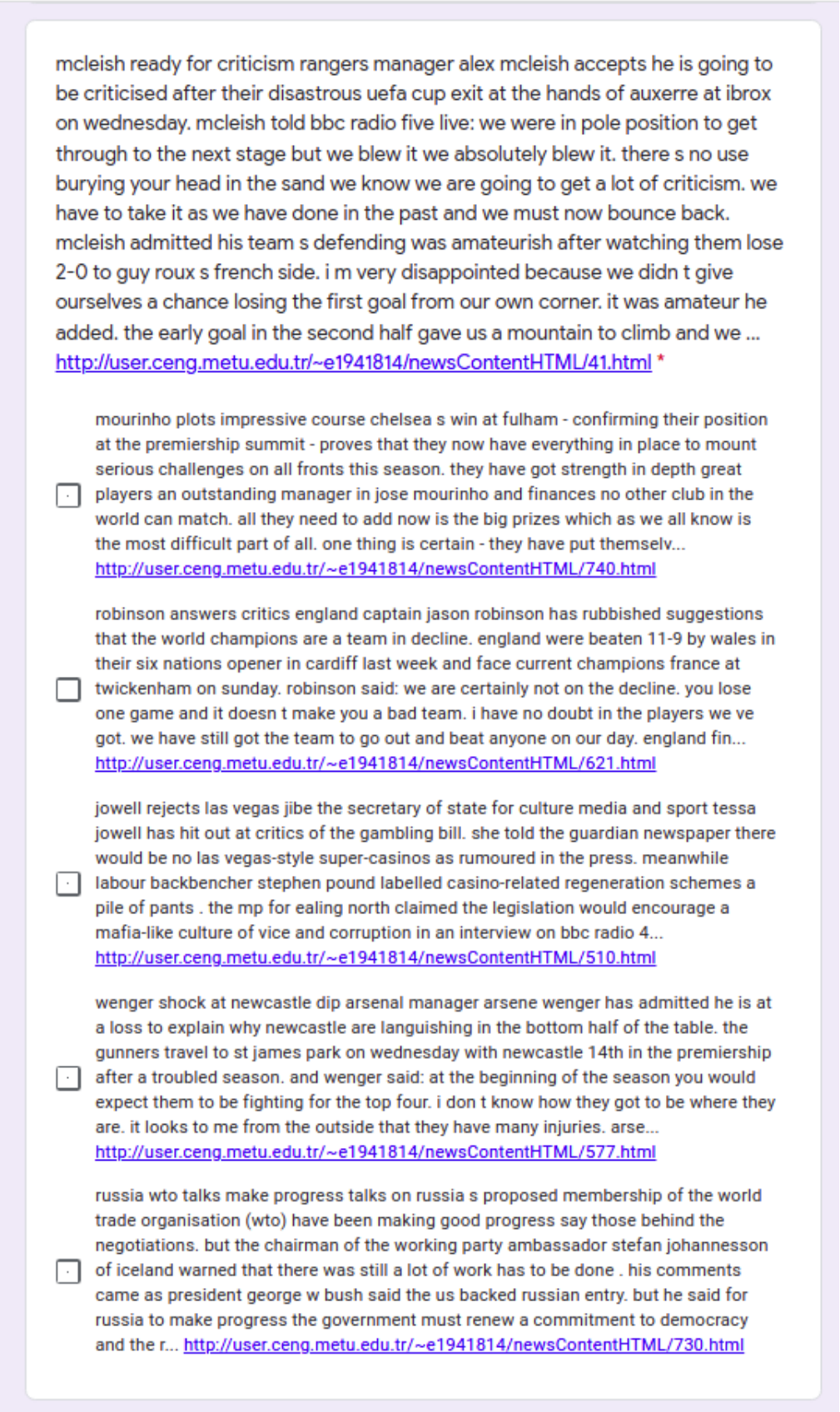

Figure A.7: A question with choices from user study for English dataset