DISTRIBUTIONAL INVESTIGATION OF SOME FREQUENT TURKISH DERIVATIONAL AFFIXES FOR EXPLORING THEIR SEMANTICS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS OF
THE MIDDLE EAST TECHNICAL UNIVERSITY
BY

GİZEM NUR ÖZDEMİR

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF COGNITIVE SCIENCE

JULY 2021

Approval of the thesis:

DISTRIBUTIONAL INVESTIGATION OF SOME FREQUENT TURKISH DERIVATIONAL AFFIXES FOR EXPLORING THEIR SEMANTICS

Submitted by by GİZEM NUR ÖZDEMİR in partial fulfillment of the requirements for the degree of **Master of Science in Cognitive Science Department, Middle East Technical University** by,

Prof. Dr. Deniz Zeyrek Bozşahin
Dean, **Graduate School of Informatics** ____________________

Assist. Prof. Dr. Ceyhan Temurcu
Head of Department, **Cognitive Science** ____________________

Prof. Dr. Cem Bozşahin
Supervisor, **Cognitive Science Dept., METU** ____________________

Examining Committee Members:

Assist. Prof. Dr. Barbaros Yet
Cognitive Science Dept., METU ____________________

Prof. Dr. Hüseyin Cem Bozşahin
Cognitive Science Dept., METU ____________________

Assist. Prof. Dr. Burcu Can Buğlalılar
Computational Linguistics Dept., University of Wolverhampton ____________________

Date: **14.07.2021**

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, Last name : Gizem Nur Özdemir

Signature : ________________

ABSTRACT

DISTRIBUTIONAL INVESTIGATION OF SOME FREQUENT TURKISH DERIVATIONAL AFFIXES FOR EXPLORING THEIR SEMANTICS

Özdemir, Gizem Nur
MSc., Department of Cognitive Sciences
Supervisor: Prof. Dr. Cem Bozşahin

July 2021, 64 pages

In agglutinating languages such as Turkish, the process of derivation is mostly performed by adding suffixes at the end of words. Most of the derivational suffixes carry a distinctive semantic content and representing them has an important role in computational tasks, such as question answering. In this thesis, we aim to explore the structure of some frequent Turkish derivational suffixes in distributional vector space by clustering word embedding vectors of them and analyzing their underlying semantic properties. Suffix vectors are obtained by subtracting the vector of the base form of the derived word from the derived word's word vector. We used a pre-trained word embedding model for obtaining word vectors and multiple unsupervised clustering algorithms with different parameters for clustering them. Our assumption is if a derivational suffix category manages to dominate one or more clusters, it is possible to obtain reliable representations of it in the distributional vector space. Our results show that many Turkish derivational suffix categories have this capability. We analyzed the underlying semantic structure of the generated clusters in terms of the thematic roles the suffixes are selecting, the UCCA labels and the UD relations the stem and the derived word can get.

Keywords: distributional semantics, word embeddings, derivational morphology, unsupervised clustering, semantics

ÖZ

TÜRKÇEDEKİ YÜKSEK FREKANSLI YAPIM EKLERİNİN SEMANTİĞİNİN KEŞFİ İÇİN DAĞILIMSAL İNCELENMESİ

Özdemir, Gizem Nur
Yüksek Lisans, Bilişsel Bilimler Bölümü
Tez Yöneticisi: Prof. Dr. Cem Bozşahin

Temmuz 2021, 64 sayfa

Türkçe gibi sondan eklemeli dillerde türetme işlemi çoğunlukla ekin kelimenin sonuna getirilmesi ile gerçekleştirilir. Türevsel soneklerin birçoğu belirgin bir anlamsal içerik taşır ve bunları temsil etmek soru cevaplama gibi uygulamalarda önemli bir role sahiptir. Bu tezde amacımız, Türkçe'nin türevsel morfolojisinin dağılımsal ve semantik yapısını türevsel soneklerin kelime temsillerini kümeleyerek ve bunların altında yatan anlambilimsel özellikleri analiz ederek incelemektir. Sonek vektörleri, türetilmiş sözcüğün kök formunun vektörünün kendi kelime vektöründen çıkarılmasıyla elde edilmiştir. Kelime vektörlerini elde etmek için önceden eğitişmiş bir kelime temsili modeli, bu vektörleri kümelemek için ise birden fazla denetimsiz kümeleme algoritması farklı parametreler ile kullanılmıştır. Varsayımımız, eğer bir türevsel ek kategorisi bir veya daha fazla kümeye hakim olmayı başarırsa, dağılımsal vektör uzayında bu ekin güvenilir temsillerini elde etmenin mümkün olduğudur. Sonuçlarımız, birçok Türkçe türevsel ek kategorisinin bu yeteneğe sahip olduğunu göstermektedir. Oluşturulan kümelerin anlamsal yapısı soneklerin seçtiği tematik roller, köklerin ve türemiş kelimelerin sahip olabileceği UCCA etiketleri ve UD ilişkileri açısından analiz edilmiştir.

Anahtar Sözcükler: dağılımsal anlambilim, kelime temsilleri, türevsel morfoloji, denetimsiz kümeleme, anlambilim

To My Family

ACKNOWLEDGMENTS

First of all, I would like to express my gratitude to my supervisor Prof. Dr. Cem Bozşahin, for his guidance through out the entire process. Without his profound expertise, his invaluable and encouraging comments this thesis would not be possible.

Besides my supervisor, I would like to thank to my examining committee members Asst. Prof. Dr. Barbaros Yet and Asst. Prof. Dr. Burcu Can Buğlalılar for their kind and valuable feedbacks.

I would also like to thank to my friends Şeyma Nur Ertekin, Şura Genç, Mine Güneş, Merve Nur Yılmaz, Tuba Köseoğlu, Zahide Akkoç and Zeynep Kara for their invaluable support and help, they made hard times a lot easier.

And my endless thanks are to my family, my father Kemal Özdemir, my brother Abdulkadir Özdemir, my grandparents Kadir and Rahime Uğur and finally my biggest suppotter, my mother, Aysun Özdemir for their love and support not just throughtout this thesis, but throughout my whole life. Thanks for always being at my side.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CBOW	Continuous Bag Of Words
GMM	Gaussian Mixture Model
MRR	Mean Reciprocal Rank
NLP	Natural Language Processing
NMT	Neural Machine Translation
NP	Noun Phrase
OOV	Out of Vocabulary
POS	Part Of Speech
Propbank	Propositional Bank
SISG	Subword Information Skip-gram
UCCA	Universal Conceptual Cognitive Annotation
UD	Universal Dependency

CHAPTER 1

INTRODUCTION

1.1. Motivation

Natural language is a way of communication between the human species. A natural language can be in the form of a text, it can be verbal or represented as signs. In any form of representation, the way language is used carries a lot of information within words. Natural language processing, NLP for short, is a subfield of multiple areas such as computer science, linguistics and artificial intelligence that concerns analyzing and extracting information from language data as well as enabling machines to process and sometimes generate human language depending on the various tasks of NLP. It is a challenging area since the data it deals with is unstructured, ambiguous and constantly evolving.

Machine learning models require quantified data as input, and NLP tasks need to represent words in the form of vectors. There are several ways to form these representations. Techniques like one-hot encoding are considered inefficient because the vectors created using this technique are sparse and they do not capture the relationship between individual words. Word embeddings comes into scene for solving such problems. This technique can be summarized with the famous quote, of Firth (1957) *you shall know a word by the company it keeps*. In other words, the main principle behind word embeddings is that words that occur in the same contexts tend to have similar meanings (Harris, 1954). Different forms of data can be used while training word embeddings. Continuous representations are a very useful and widely used form of such training data. However, these representations ignore the morphology of words therefore do not provide us direct information on individual morphemes. It is also not likely to cover the entire vocabulary of a language with these models even if they are trained on a very large corpus. This problem is called being out of vocabulary (OOV), and agglutinating languages like Turkish, Finnish and Korean are very likely to suffer from it due to their highly productive morphology.

In this thesis, we focus on derivational suffixes of Turkish by extracting their representations from word embedding vector space and perform unsupervised clustering on obtained vectors. We extract their representations from word embedding vector space and performed unsupervised clustering on obtained vectors. We used only the words that have a single morphological analysis in order to avoid ambiguity. We excluded words getting suffixes that do not have content that are actually structural suffixes or case markers such as -ArAk or -mA. This is because these suffixes are able to combine with many Turkish words and do not have semantic contributions to the derived words. The semantic analysis is performed in these clusters in order to understand the effect of semantics in distributions of derivational morphemes. Other than the pairs with structural affixes, we excluded the pairs that do not contain the most frequent suffixes. There is no annotation that we know of which indicates the morpheme's choice of the stem. This is important for the future prospect of wide-coverage semantic parsing in morphologically rich languages because one cannot expect to find a derivational affix with every possible stem in training sets. For instance, -cI is present in "şekerci" (confectioner), "manifaturacı" (draper), "kurucu" (founder), "kamyoncu" (trucker) and similarly familiar words achieving varying semantic contributions depending on stem's semantic properties, but it is unreasonable to expect to see all the word forms with -cI in training, therefore it is important to understand -cI's selection. In literature, we usually find the whole word's dependencies at best, which gives us not much information about the dependency between stem and the derivational affix. In this thesis, we explore how much we can find these relations from the current word-level annotations during the semantic analysis part of the thesis. We included only the most frequent suffixes with the hopes of encountering them more in word-level annotations. Different semantic resources with different techniques are used during the process.

1.2. Related Work

There are several different studies in terms of the data forms used to create embeddings in addition to using continuous forms of words such as using characters (Chen et al. 2015), morphemes (Cotteral and Schütze, 2019; Luong et al., 2013) or syllables (Choi et al., 2017, Şahin and Steedman, 2018), all of which have their own strengths and weaknesses. For instance, character embeddings are better than word and morpheme level embeddings when problems like OOV and infrequent words are considered, but they cannot give information on the words or morphemes in the distributional vector space, which is a drawback that morpheme and word embeddings do not hold.

There are several studies focusing on morphological relations in word embeddings. Gladkova, Drozd and Matsuoka (2016) worked on derivational and inflectional relations in English separately in addition to lexical relations using analogy pairs. Their conclusion on derivational morphology is that it is way harder to detect than inflectional morphology. Güngör and Yıldız (2017) investigated derivational and inflectional morphology of Turkish again with using analogy pairs. They also concluded that derivational relations are not as successfully detected as inflectional ones using MRR as the evaluation technique. Musil et al. (2016) focused only on derivational morphology of the Czech language, performed unsupervised clustering on the word embeddings obtained by getting the difference of vectors in pairs using both Neural Machine Translation (NMT) and skip-gram models, and evaluated the clusters with multiple evaluation metrics. They concluded that it is possible to capture semantic classes of derivational relations between words by clustering obtained word vectors.

Şenel et al. (2018) investigated the interpretability of the word embeddings and the underlying semantic structure encoded in different dimensions of a word embedding. They introduced SEMCAT, which contains 110 semantic categories, as their semantic resource and performed their semantic analysis using this dataset. They obtained different results in terms of semantic analysis when different methods of creating interpretable word embeddings were used.

1.3. Outline

We divide this thesis into eight main chapters. In the introduction, we state our motivation and present related work. Then we talk about Turkish morphology with the focus on derivational morphemes and semantics in terms of thematic roles, the UCCA framework, and the UD treebank in chapter 2. In chapter 3 we explain word embeddings. In chapter 4, we talk about the different types of unsupervised clustering methods we used. We continue by explaining our dataset in chapter 5, where we explain that we identified the most frequent derivational affixes from distributional data and use them in the rest of the thesis. In chapter 6 we present the clustering results of the data that we explained in chapter 5, using various clustering techniques. In chapter 7 we explain our semantic analysis methods and results on clustering data with different semantic annotation datasets. Here we did not confine ourselves to Turkish annotation only and looked at the English propositional bank along with the Turkish one, with the assumption that translation of Turkish words with our suffixes will be semantically related in other languages as well, so that their clustering may also be worth exploring. We also looked into UCCA framework annotations, and to try even further, we have looked at dependency parsers that parse words with our affixes, trying to find out any grouping among these words and their dependencies. Chapter 8 presents results and evaluations. In chapter 9, we talk about the limitations of our approach. We conclude with conclusions and future work.

CHAPTER 2

MORPHOLOGY AND SEMANTICS

2.1. Morphology

2.1.1 Introduction

Morphology in linguistics is the study of forms of words. Words are formed by combining one or more morphemes, which are the smallest meaning bearing units of a natural language. There are three main processes of word-formation in linguistics which are derivation, inflection and word-formation. Inflections create different forms of the same lexeme by performing operations like adding case markers, changing the tense of the word, or making the word plural. Derivation, on the other hand, creates new lexemes from the existing ones. Therefore, the morphological structure of a word can contain a single root, multiple prefixes, suffixes, or infixes.

Languages can be classified according to their word-building processes. Languages that fall into analytic and isolating classes do not have a rich morphology and their words may consist of a single morpheme. Mandarin Chinese is an example of a language that falls into that class. Fusional languages can have more than a single morpheme, with a morpheme having the ability to contain multiple grammatical information within it. An example is Spanish to that category. Polysynthetic languages can have multiple stems in a word in addition to multiple morphemes.

In agglutinating languages, words can be combinations of morphemes like fusional ones with a single stem, but unlike fusional languages, their morphemes have a one to one corresponding relationship with grammatical features.

In our study, we focus on Turkish which is an agglutinating language with word structures formed by productive affixations of inflectional and derivational morphemes. (Oflazer et al., 1994).

2.1.2 Turkish Inflectional and Derivational Morphology

Turkish word-formation process relies heavily on suffixation with derivational and inflectional morphemes. According to Oflazer (2003), in addition to the stem, there are on average two to three morphemes per word. There can be 2^3 continuous inflectional morphemes per stem, for number, possession, and case (Cakici et al., 2018). The number of derivational morphemes per word is 1.7 on average (Oflazer, 2003) and the total number of derivational morphemes is approximately 110 (Oflazer et al., 1994).

Theoretically, it is possible to produce words having indefinite lengths by affixation in Turkish. Below is an example from Oflazer et al. (1994), showing the productive word-formation nature of Turkish.

OSMANLILAŞTIRAMAYABİLECEKLERİMİZDENMİŞSİNİZ

Which is broken down into its morphemes in the same source as follows:

OSMAN-LI-LAŞ-TIR-AMA-YABİL-ECEK-LER-İMİZ-DEN-MİŞ-SİNİZ

-'s above indicate morpheme boundaries. The translation of the phrase to English is "(behaving) as if you were of those whom we might consider not converting into an Ottoman." (Oflazer et al., 1994). This extreme example might indicate that it is possible to produce novel word forms in Turkish by performing affixation.

Bozşahin (2018) has made a tentative offer of around 24 concepts to encode the semantics of 104 derivations. In his study, concepts are formed and coded base on the POS tags of base and lemma, following the function of the suffix or a representation of an allomorph. For instance, VVI_ANEG is the code for the inflections having a verb as root, making the verb a negative where I means inflection and VV means verb; NJD_CA is a noun (N) as input to the morpheme, making it an adjective (J) by adding "-ca" suffixal derivation (D).

Words in Turkish can also have ambiguous morphological analyses. For instance, the Turkish word "masalı" can be morphologically interpreted in three ways; two of them having the root "masal" (tale) and getting accusative or possessive inflectional affixes, one of them having the root "masa" (table), and getting derivational affix "-lı" which adds the word the meaning of "with" (Yüret and Türe, 2006). Therefore classifying a suffix as derivational in Turkish is heavily depends on the context of the word.

2.2 Thematic Roles

Below are the two example sentences one of which is grammatical, and the other is not.

(2.1) The girl is reading a book.

(2.2) She sleeps a sheep.

Our intuition of such inferences comes from the syntactic and the semantic restrictions of the verb of a sentence. In the case of our examples, in (2.1) our verb "read" takes two NP's as its arguments, one being the subject and the other direct object. In (2.2), the syntactic properties of the verb "sleep" confirms our intuition of the sentence not being grammatical since it takes a single NP as its subject therefore giving a second NP as its argument makes the sentence ungrammatical. Now consider the following:

(2.3) The girl writes a book.

(2.4) The book writes a girl.

Both are grammatical, with different semantic implications. Here, the semantic properties of the verb cause this difference. In our example, *write* requires an *agent* that can perform the writing action, and the object being written needs to be writable.

All the verbs in a natural language have a semantic relationship with their arguments. In linguistics, *thematic roles* explain that relationship. They were first introduced by Gruber (1965), and other researchers contributed to the theory of the thematic roles (Fillmore, 1967; Jackendoff, 1987; Dowty, 1991). Below we briefly illustrated common thematic roles.

(2.5) Agent: The performer of the action

Theme: The entity that changes its location by the action

Instrument: The entity that the action is performed with

Goal: The target of the entity that is moving towards to

Source: The entity that the target started to move from

Location: Location of the action that is taken place

In our thesis, we will use the thematic roles for analyzing the semantic content of derivational suffixes after the clustering on derivational word vectors are performed.

2.3 Universal Conceptual Cognitive Annotation (UCCA) Framework

UCCA (Abend and Rappoport, 2013) is a multi-layered framework which is applicable cross-linguistically and used for encoding semantic annotation. The framework is designed in such a way that the annotators without a linguistic background can also be trained easily. Golden annotations of UCCA are available online for English, French, and German.

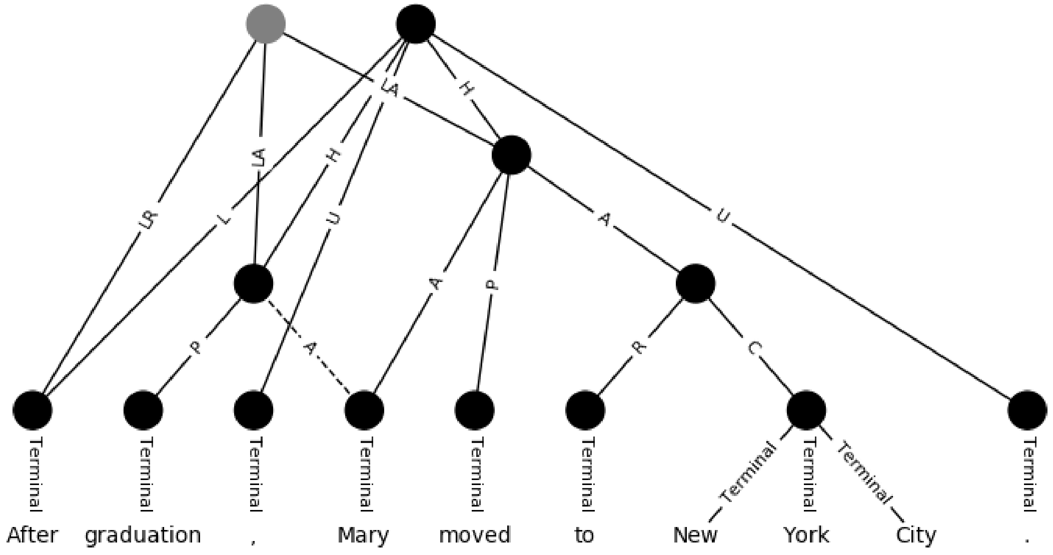


Figure 2.1 An Example of UCCA annotation for the sentence "After graduation, Mary moved to New York City."

UCCA is represented with directed acyclic graphs (DAGs). Its nodes are called units, which can consist of one or multiple tokens. Figure 2.1 illustrates an example of the UCCA annotation. In the internal representation of a unit, it has unbound edges and its semantic categories. The main type of units are Scenes, which forms the foundational layer of the framework. A Scene express either a Process (P) or a State (S) along with the participants (A) in the Scene (Abend et al., 2020). In most cases, a participant is an entity or a location. The modifiers of the scenes are called Adverbials (D).

There are other sets of categories that do not evoke a scene. The time at which the Scene has occurred is represented as Time (T). Elaborators (E) describes the entity called the Center (C). Phrases like *and*, that connect multiple C's are called Connectors (N). Relations (R), relates multiple scenes without making new ones. Phrases that are not participants or that do not indicate relations are called Functions (F) in the UCCA framework.

Phrases like *while, however* that connect the Scenes are called Linkers (L). The Scenes that are connected by a Linker are called a Parallel Scene (H). A linker is not necessary for linking the Parallel Scenes.

Some other labels of the UCCA framework that took place in this thesis are Ground (G), which relates the speech with the spoken Scene, Quantity (Q), and predicate (P). We used the UCCA framework for analyzing the underlying semantic structure of the derivational suffixes in distributional vector space after clustering them.

2.4 Universal Dependency (UD) Treebank

Universal Dependency is a framework that is available across many languages which is accessible online. It consists Stanford dependencies (De Marneffe et al., 2014), Google universal POS tags (Petrov et al., 2012), and the morphosyntactic tag set presented by Zeman (2008). The main goal of the framework is to provide a consistent cross-linguistic annotation guideline.

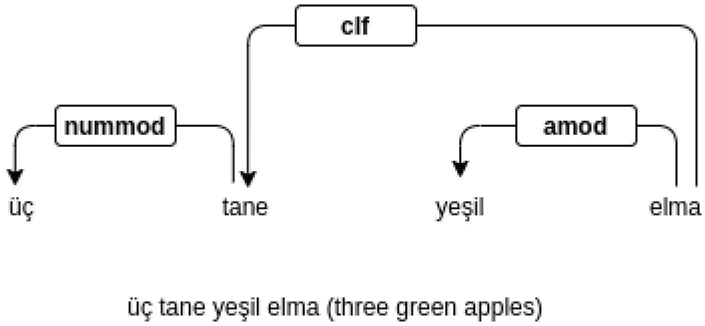
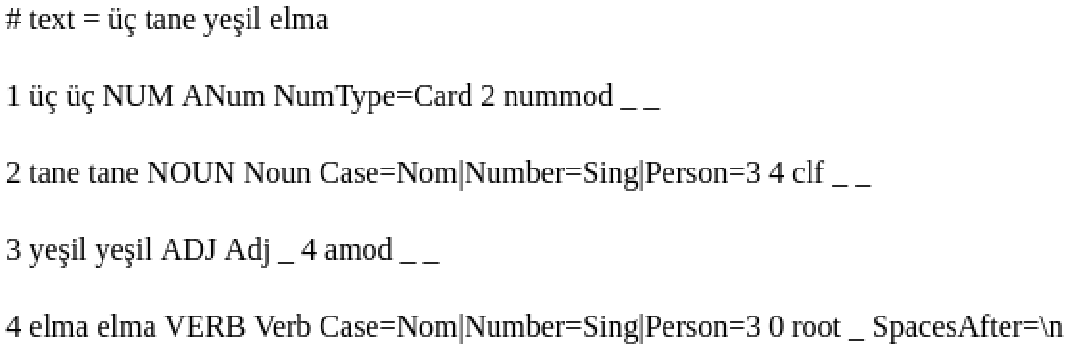


Figure 2.2: Example Graphical Representation of UD tagging

Figure 2.2 illustrates an example graphical dependency representation of the Turkish sentence *üç tane yeşil elma* (three green apples). In our thesis, we used the UD Turkish Boun Treebank (Türk et al., 2020) as our semantic data source. It consists of 9757 annotated sentences from various topics and is accessible online. It uses the CoNNL-U format, which includes the id, the form, the lemma, the universal part-of-speech tag, the language-specific POS tag, the list of morphological features, the id of the head of the word, the universal dependency relation of the word, enhanced dependency graph,and any other annotation respectively. The fields are separated by tabs in the files using this format.



```
# text = üç tane yeşil elma

1 üç üç NUM ANum NumType=Card 2 nummod _ _

2 tane tane NOUN Noun Case=Nom|Number=Sing|Person=3 4 clf _ _

3 yeşil yeşil ADJ Adj _ 4 amod _ _

4 elma elma VERB Verb Case=Nom|Number=Sing|Person=3 0 root _ SpacesAfter=\n
```

Figure 2.3: CoNNL-U format of the sentence üç *tane yeşil elma*

Figure 2.3 shows the annotation of the sentence üç *tane yeşil elma* (three green apples).

In our thesis, we used the universal dependency relations of the derived word and the stem of our derivational pairs. Below are some universal dependency relations that took place in our analysis of our clusters.

(2.7) acl: clausal modifier of noun
advcl: adverbial clause modifier
advmod: adverbial modifier
amod: adjectival modifiers
appos: appositional modifier
case: case marking
ccomp: clausal complement
clf: classifier
compound: compound
conj: conjuction
csubj: clausal subject
dislocated: dislocated elements
flat: flat multi-word expression
obj: object
obl: oblique nominal
nmod: nominal modifiers
nsubj: nominal subject
root: root

CHAPTER 3

WORD EMBEDDINGS

There is a large amount of text data generated every day and making the computers analyze them is a keen interest for businesses as well as academic researchers. Communicating with computers via natural language also took interest of people from different backgrounds like AI researchers, engineers and, philosophers since the 1950s. Using machine learning methods for achieving such goals is very popular and common today more than ever thanks to stronger and cheaper hardware in addition to easy to maintain ML libraries available.

For any purpose under the mentioned topics, the first step is to represent the collected text data in a form of which machine learning models can process, which requires them to be transformed into vectors of quantities. There are numerous methods for creating vector representations of words. Below we will explain two of them before we explain word embeddings, which is the method we used in this thesis.

3.1 One-Hot Encoding

This method creates a vocabulary-size vector for each word in the corpus, then it encodes a single unique index as 1 for each word, the remaining indices are encoded as zero. If we consider the sentence, "Building a thinking machine with machine learning is fun" as our corpus, we might end up with the vector representations illustrated in (3.1).

(3.1)	building	1 0 0 0 0 0 0 0
	a	0 1 0 0 0 0 0 0
	thinking	0 0 1 0 0 0 0 0
	machine	**0 0 0 1 0 0 0 0**
	with	0 0 0 0 1 0 0 0

machine	**0 0 0 1 0 0 0 0**
learning	0 0 0 0 0 1 0 0
is	0 0 0 0 0 0 1 0
fun	0 0 0 0 0 0 0 1

In real world applications and academic researchers, the input data is not as small as our illustrated example and using this method creates vectors having most indices as zeros, therefore this method is insufficient for the representation of large datasets.

3.2 Integer Encoding

This method encodes each word in a corpus as a unique integer. If we consider the sentence (3.2) as our example, with integer coding a possible representation of the sentence might be a vector like [1, 2, 3, 4, 5, 4, 6, 7, 8]. Integer encoding does not suffer from the vector sparsity problem as one-hot encoding does, but it is not able to capture the relation between the words in the corpus.

3.3 Word Embeddings

Word embeddings is a way of representing words as dense vectors while using their distributions in data, which allows the vectors to investigate the semantic relation between the words by taking account of Firth's (1957) principle. The history of using distributional hypotheses for generating word representations goes back to early 2000s. Bengio et al. introduced neural probabilistic language model in 2003 which learns the distributional word representations by using feed forward neural network. Toutanova et al. (2003), showed that using the words both on the left and right side of the center word gives better results than using either the words that are on the left side or right side of the center word. The number of words that are near the center word that needs to be considered while creating word vectors is called the window size of the language model.

In 2013, Mikolov et al. introduced word2vec models with two vector formation methods, which are CBOW and Skip-gram models, using 5 as window size and 300 as vector dimension. Below we explain these concepts in more detail.

3.3.1 Continuous Bag of Words (CBOW) and Skip-gram

CBOW models try to predict the word's vector by looking at its surrounding words, while Skip-gram tries to predict the context by looking at the distributional representation of the center word. Both architectures use neural networks for creating word representations. Unlike the common usage of neural network models, word vectors are formed by the learned weights of the projection layer instead of the output of the neural networks.

In the Skip-gram model, the network gives the probability of being the randomly chosen word within the window size of the input word for each word in the vocabulary. For instance, if the input word is “capital”, we would expect to have high probabilities predicted for the words like “Ankara” and “London”, whereas less relevant words like “wardrobe” and “sharpener” would likely to have lower probabilities.

The first thing to do for obtaining word vectors from a neural network is to create a vocabulary from the training corpus. Since a neural network is also a machine learning model and machine learning models require its inputs to be in numerical form, a numerical representation of each word in the vocabulary is needed. For that purpose, all the words in the vocabulary are represented as unique one-hot encoded vectors each having the size of the vocabulary.

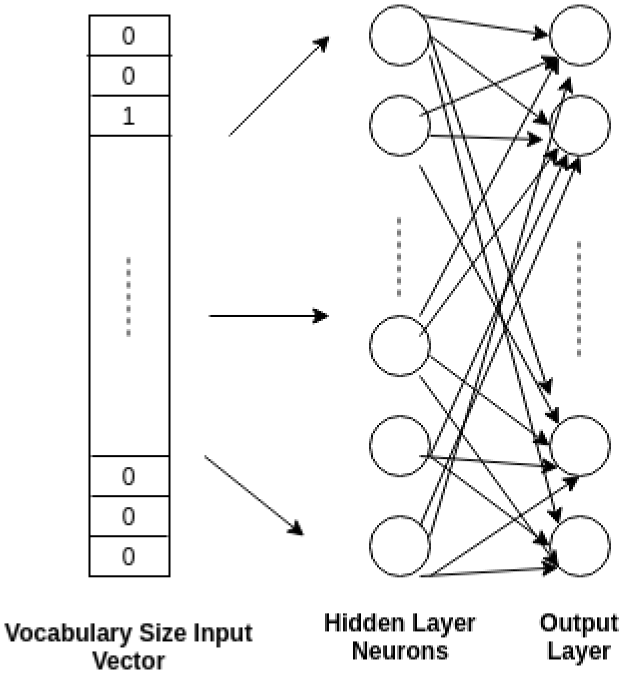


Figure 3.1: Example Representation of One Layer Neural Network

The number of neurons that the projection layer has is a hyperparameter that needs to be adjusted. The output vector is a probability distribution giving each word’s probability of being nearby of the center word, which is not the main interest of word vectors, but the weight matrix of the projection layer. The weight matrix of the projection layer is initialized with random values other than zero and they are adjusted in every iteration for minimizing the loss function. If two words appear in similar contexts, the weight matrices they have would be also similar, consequently, they would have similar word embeddings. This similarity between word vectors can be measured with different measurement techniques, one of them which we used in this thesis is cosine similarity. Cosine similarity measures the cosine angle between the vectors.

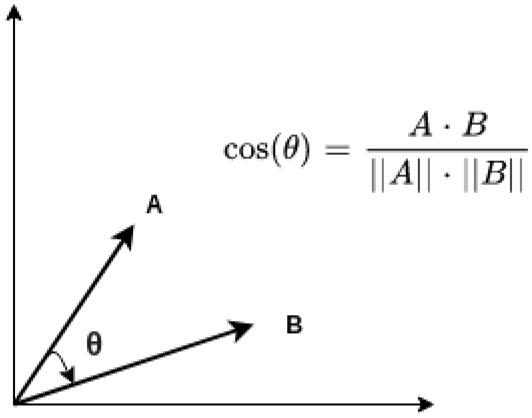


Figure 3.2: Representation of Cosine Similarity

Cosine similarity can measure the similarity of vectors in multidimensional space. It is the cosine of the angle between the vectors that takes values between 0 and 1. Figure 3.2 illustrates the cosine similarity graphically in addition to its calculation formula.

3.3.2 FastText

FastText (Bojanowski et al., 2017) is used as the training algorithm of the word vectors in this thesis. It is an open-source library developed by Facebook. FastText overcomes the disadvantage of word2vec that is ignoring the structure of the word while assigning their vectors. For example, word vectors for the words “study”, “studies” and “studied” are assigned distinctively from each other when word2vec is used as the method. It is quite possible that the model would capture there is some kind of relationship between these words and assign them similar word vectors after training, but that is less possible when words with rear affixes are concerned. FastText solves this problem with an algorithm called *Subword Information Skip-gram* (SISG) which uses character n-grams as its core structure. Character n-grams are sequences of n characters that are extracted from a word (Majumder, 2002). SISG adds “<” character to the beginning and “>” character to the end of the word, and afterwards the word is broken up into its n-grams. As an example, 3-grams for the word *study* would be “<st”, “stu”, “tud”, “udy”, “dy>” using SISG. After forming the n-grams of a word, they will be used in the training of the word vector. The sum of all the trained n-gram vectors is assigned as the word vector of the full word. Using this method will result the words having similar orthographic structure to have similar word vectors by means of them having one or multiple number of same n-grams that are used while training.

3.3.3 Normalization of Word Embeddings

Normalization enables the instances of data to have unit forms. According to Wilson and Schakel (2015), applications that are using word embeddings for exploring the relations between them, like similarity task, are showing better performance when word vectors are normalized. In this thesis, we performed clustering tasks with various algorithms using both normalized and original versions of word vectors and reported the results of both approaches individually.

CHAPTER 4

METHODS OF CLUSTERING

4.1 Clustering

Machine learning algorithms can be classified under two main categories, which are supervised and unsupervised learning. In supervised learning training data is labeled and the algorithms learn to map the input data to their respective output and predict the output of any new coming input data. Email spam detection is one example of such algorithms. In unsupervised learning algorithms, the training data is not labeled, and the algorithms try to gain more information about the structure of the data instead of predicting the output of a given input. Clustering algorithms fall into the category of unsupervised learning algorithms. As the name suggests, these algorithms try to cluster the data points based on how similar they are. There are different clustering methods and similarity metrics, all of which have their own strengths and weaknesses due to their internal structure, some are better than others when different shapes of clusters are considered.

In this thesis, we focused on clustering algorithms of two main categories, centroid-based (K-means, Gaussian Mixture Models) and hierarchical (agglomerative) clustering due to their ability to capture similarity on different shapes of data.

4.1.1 K-means Clustering

K-means clustering clusters each data point in a dataset into a predetermined k number of clusters iteratively, based on the similarity of their features.

According to Kodinariya et al. (2013), the k-means clustering algorithm partitions the data into k clusters using the following steps:

(1) Place k points into vector space represented by the clustering objects as initial centroids.

(2) Assign each data point to the cluster that has the closest centroid to the data point.

(3) When all data points are assigned to a cluster, recalculate the positions of the centroids.

(4) Repeat step 2 and 3 respectively until the centroids can no longer change.

There are different ways to perform step (1), picking the initial centroids. A good method to use is to assign the centroids by hand if it is possible to know the centroids even approximately. A common way is random centroid assignment, which does not considered to be the best way since it is not very likely to choose proper centroids randomly. If the random centroids are not well assigned, the number of iteration could increase and this will decrease the performance of the algorithm. Poorly assigned centroids might also cause the clusters to be not formed well. In addition to that, due to the randomness of initial centroids, the formed clusters might vary in different runs. The algorithm we used in this thesis for initializing our centroids is called k-means++. This algorithm uses the following 4 steps for centroid initialization.

(1) Select the first centroid randomly among the data points.

(2) For each point of data, calculate its distance from the centroid that was chosen before, and that is closest to the data point.

(3) Select the next centroid that is proportional with the distance calculated in second the step.

(4) Repeat the previous steps until k centroids are selected.

K-means++ can be considered as a smarter way of centroid selection since it enables to select the centroids that are the farthest from each other.

There are different measurement metrics that are used to calculate the distance between data points and centroids for performing Step (2) of the k-means algorithm. The metrics we used in this thesis are cosine and Euclidean distance metrics. We explained the cosine similarity metric in Chapter 3 under Continuous Bag of Words (CBOW) and Skip-gram subtitle. Euclidean distance is calculated as follows:

(4.1) $\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$

Here x and y are two vectors having n dimensions. K-means clustering algorithm tries to minimize this sum of squared distances for each data point in a cluster from its centroid. The algorithm then sums up all squared distances for each k clusters.

4.1.2 Agglomerative Clustering

Agglomerative clustering is a type of clustering algorithm that falls under the category of hierarchical clustering. This algorithm starts by assigning each data point to a separate cluster and merge them into larger clusters (Madhulatha, 2012). The merging process happens due to the similarities or dissimilarities between data points, in each iteration of merging most similar data points according to the selected similarity metric are merged in together. This kind of clustering can be represented as a dendrogram.

As in K-means, agglomerative clustering also needs a distance metric for the process of merging separate clusters. We used Euclidean distance as our metric in this thesis for calculating the distance between data points while merging them into clusters.

In addition to that, a method is needed for calculating the distance between the clusters, which is called the linkage method. Ward's method is used as the linkage type of agglomerative clustering in the scope of this thesis. This method calculates the distance between two clusters by looking at the increase in the sum of its squares when they are merged.

(4.2) $\sum \frac{\mathrm{dist}(p_i - p_j)^2}{|(c_1)| - |(c_2)|}$

(4.2) shows how to find Ward's distance between two clusters where p_i and $p_j$represents the data points in the first and second clusters respectively and $c_1$and c_2 are the representations of their respective centroids. At the beginning of the clustering process sum of squares starts as zero since all data points belong to their respective clusters and it grows with iterations as the merging of clusters happens. What Ward's method tries to achieve is to keep the growth of this number in the smallest amount (Madhulatha, 2012).

The process of performing agglomerative clustering can be summed up in the following steps.

(1) Assign each data point to a separate cluster.

(2) Choose a distance metric and create a distance matrix.

(3) Choose a linkage method for merging the clusters that are having the smallest distance in the matrix.

(4) Merge the clusters.

(5) Update the distance matrix with newly formed clusters.

(6) Repeat the process until you reach the determined number of clusters.

4.1.3 Gaussian Mixture Models (GMMs)

GMMs approach the problem of clustering in a probabilistic manner. In k-means, data points were clustered by the distance of the nearest centroid to them. In GMMs, the data is again clustered into initially determined k clusters, but rather than looking at only the distance to centroids, parameters like variance and mean are also considered. In other words, the models think of the data as the weighted sum of k Gaussian distributions. (4.3) represents the approach as a formula.

(4.3) $p(x) = \sum_{k=1}^{k} a_k \, g(x \vee u_k, \sigma_k)$

Here k is the number of clusters, function g represents the clusters in data having $a_k$as the assigned weight of the Gaussian, σ_k as co-variance, and u_k as mean. We say that each Gaussian distribution of k is a separate cluster in our data.

One advantage of GMMs over k-means is it is more flexible when the shape of the clustering data is concerned. GMMs can capture the shape of ellipses while k-means is more successful when the clusters of data are circularly shaped. This is because k-means does not consider mean and variance like GMMs do. Another advantage of GMMs over k-means is that k-means can only tell whether a data point belongs to a cluster or not, whereas GMMs are able to tell the probability of a data point is included to a cluster. This probabilistic kind of clustering is called soft clustering, while clustering like k-means is called hard clustering.

CHAPTER 5

DATA

5.1 Derivational Relations

We used the open-source data presented by Özbek (2012) for extracting derivational relations containing 88705 distinct Turkish words. Derivational relations are extracted from the dataset by performing morphological analysis using the open-source Java library Zemberek (Akın and Akın, 2007), which is specifically developed for NLP tasks for Turkish. Zemberek is able to tell whether a word is derived from another word, with the base of the derived word. It assigns the same tag to derivations having different orthographies but belongs to the same derivational class. For instance, *-lık, -lik, -luk, -lük* affixes are different forms of the same affix, and all of them are classified as *Ness* in Zemberek. This information is not used during clustering, but we use it while evaluating created clusters. Using a library while forming such datasets can reduce the preparation time and the need for qualified human annotators.

In order to avoid a further process of disambiguation, we chose the words which have one morphological analysis instead of disambiguating the words. This is done as such instead of disambiguating the words in the dataset because the obtained words are from a dictionary-like database therefore they do not appear in a context. The word embedding model is also pre-trained on the Common Crawl and the Wikipedia datasets therefore we do not have its context either. We also removed all affixes having fewer than 20 instances for eliminating the suffixes with lower frequency. As we mentioned in Chapter 1.1, we also removed the derivational relations that do not have semantic content. That left us with 8503 unique derivational relations with 10 distinct suffix categories out of 33870 derivational relations with 37 suffix categories, all of which are relatively frequent, and bears semantic content within them Table 5.1 shows all derivational suffixes extracted from the data before removing processes.

Table 5.1: All Derivational Classes Extracted from Dataset

Suffix Category	Suffixes	Frequencies
Inf1	mak, mek	6733
Inf2	ma, me	6284
Ness	lık, lik, luk, lük	4799
Agt	cı, cu, çi, ci, çı, ıcı, çu, yıcı, ici	2548
With	lı, li, lu, lü	2454
Caus	tır, t, dır, tir, dir, dur, dür, tür, tur	1439
Pass	ıl, il, ul, ül, nıl, nil, nul, nül, ın, in	1409
Without	sız, siz, suz, süz	1239
Acquire	lan, len	993
Become	laş, leş	986
Inf3	ış, iş, uş, üş, yış, yiş, yuş, yüş	935
PresPart	yan, yen, an, en	767
NarrPart	mış, miş, muş, müş	634
AorPart	ar, er, ır, ir, ur, ür, r	430
Ly	ca, ce	408
AsIf	ca, ça, casına, ce, çe, cesine	379
JustLike	sı, si, su, sü, ımsı, imsi, umsu, ümsü, msı, msi, msu, msü	261
Related	sal, sel	218

Dim	cık, cik, cuk, cük	199
ByDoingSo	arak, erek, yarak, yerek	177
FutPart	acak, ecek, yacak, yecek	71
WithoutHavingDoneSo	meksizin, maksızın	71
NotState	mazlık, mezlik	70
PastPart	dık, dik, duk, dük, tık, tik, tuk, tük	66
ActOf	maca, mece	58
Able	abil, ebil, yabil, yebil	51
While	ken	32
FeelLike	ası, esi, yası, yesi	31
SinceDoingSo	al, el	30
Rel	ki, kü	30
When	ınca, ince, unca, ünce, yınca, yince, yunca, yünce	18
AfterDoingSo	ıp, ip, up, üp, yıp, yip, yup, yüp	13
EverSince	agel, egel	8
Adamantly	asıya, esiye	7
Repeat	adur, edur, yadur, yedur	6
Stay	akal, ekal, yakal, yekal	5
Hastily	ıver, iver, uver, üver	5
AsLongAs	dıkça, dikçe, dukça, dükçe, tıkça, tikçe, tukça, tükçe	3
Almost	ayaz, eyaz	3

Table 5.2 Extracted Derivational Relations and Frequencies After Processing

Suffix Category	Examples	Frequencies
Ness	girişimci – girişimcilik, hoşnut- hoşnutluk	3034
With	manzara – manzaralı, okul - okullu	1641
Agt	türkçe – türkçeci, bağış - bağışçı	1459
Without	kurgu – kurgusuz, sorun - sorunsuz	894
Become	kurum – kurumlaş, ebedi - ebedileş	575
Inf3	aktar – aktarış, sığdır - sığdırış	560
Related	işlev – işlevsel, neden - nedensel	125
Dim	tanıtım – tanıtımcık, iplik - iplikçik	100
Acquire	tav- tavlan, yetki – yetkilen	93
Ly	enli – enlice, boylu - boyluca	22

Categories that are eliminated due to their structural nature are Inf1, Caus, Pass, PresPart, Inf2, ByDoingSo, Able, and JustLike. Categories that have low frequencies, less than 20, are directly eliminated. Other frequency differences between Table 5.1 and Table 5.2 are because of the excluded words that are having ambiguous nature, for instance from Table 5.1 we can see that there are 4799 word pairs having Ness as their suffix category in total and this number drops to 3034 in Table 5.2, this tells us that there are 1765 pairs that contain ambiguous word or words, therefore they are excluded from the data we used for clustering.

5.2 Pre-trained Word Embeddings

We used the pre-trained FastText model presented by Grave et al. (2018). The models of the study are publicly available for 157 languages including Turkish. It uses Common Crawl and Wikipedia datasets as training data using the CBOW method with 300 dimensions, character n-grams having the length of 5 and 5 as the window size.

CHAPTER 6

CLUSTERING DERIVATIONAL PAIRS

6.1 Introduction

We used three main unsupervised clustering algorithms with their own respective parameters and 10 clusters for each of the suffix category. Clustering and vector normalization processes are performed using Python as the programming language and Scikit-Learn (Pedregosa et al., 2011) as the machine learning library. Scikit-Learn is a free and easy to use library developed for Python developers which is available online.

We evaluated the distribution of each category in each cluster and detected whether a suffix category can dominate a cluster or not. We concluded that if an affix is capable to dominate a cluster, and distributed in fewer clusters, it has more a predictable meaning, therefore its semantic content affects its vectors, and it has the capability of having reliable word vectors. Below we presented our results for each clustering algorithm in matrices. In the matrices, the columns represent the suffix categories, and each row represents a cluster.

6.2 Clustering with K-Means

We used two distance metrics while clustering with k-means algorithm, which are cosine and Euclidean distance metrics. We initialized centroids using k-means++ algorithm. We illustrated the distributions for each metric while original and normalized word vectors are used in the tables below. The results are confirming one of our hypotheses which is affixes that are having higher frequencies to have clusters that are separated more clearly than affixes having low frequencies. From the results of both matrices, we see that Ness, With, Inf3, Agt, and Without, which are having higher frequencies than Acquire, Ly and Dim are dominating one or more clusters.

Table 6.1 Clustering Results of K-means clustering with Euclidean Distance and Normalized Word Vectors

	Ness	With	Agt	Without	Inf3	Become	Related	Dim	Acquire	Ly
Cluster1	**0.99**	0.003	0	0	0	0	0.003	0	0	0.003
Cluster2	**0.98**	0.001	0.001	0.004	0.001	0	0	0.004	0	0
Cluster3	**0.95**	0.01	0.01	0	0	0.02	0	0	0	0.01
Cluster4	0.004	**0.59**	0.07	**0.31**	0	0	0.02	0	0	0
Cluster5	0.08	**0.56**	0.1	0.19	0.01	0	0.02	0.03	0.01	0
Cluster6	0.18	**0.40**	0.2	0.08	0.03	0	0.04	0.02	0.02	0
Cluster7	0.14	**0.35**	**0.33**	0.14	0.01	0	0.01	0.02	0	0
Cluster8	0.11	**0.35**	0.09	**0.3**	0	0	0.09	0.03	0.02	0
Cluster9	0.01	0.18	**0.79**	0.01	0	0	0	0.01	0	0
Cluster10	0.04	0.04	0.16	0.01	**0.73**	0	0.01	0.01	0	0

Table 6.2 Clustering Results of for K-means clustering with Euclidean Distance

	Ness	With	Agt	Without	Inf3	Become	Related	Dim	Acquire	Ly
Cluster1	**0.93**	0.01	0.01	0.00	0.02	0.01	0.01	0.01	0.00	0.00
Cluster2	0.01	**0.57**	0.07	**0.31**	0.00	0.00	0.02	0.00	0.01	0.00
Cluster3	0.09	**0.37**	0.15	**0.30**	0.01	0.00	0.04	0.02	0.02	0.00
Cluster4	0.13	**0.34**	**0.24**	**0.20**	0.02	0.00	0.01	0.03	0.02	0.00
Cluster5	**0.33**	**0.33**	0.00	**0.33**	0.00	0.00	0.00	0.00	0.00	0.00
Cluster6	0.09	**0.34**	**0.44**	0.08	0.00	0.00	0.03	0.02	0.00	0.00
Cluster7	0.00	**0.33**	**0.33**	**0.33**	0.00	0.00	0.00	0.00	0.00	0.00
Cluster8	0.00	0.00	**0.25**	0.00	**0.25**	0.00	0.00	**0.50**	0.00	0.00
Cluster9	0.18	0.18	**0.20**	0.12	0.18	0.00	0.04	0.08	0.00	0.00
Cluster10	0.11	0.09	0.19	0.04	**0.51**	0.01	0.02	0.01	0.02	0.00

Table 6.3 Clustering Results of K-means clustering with Cosine Distance

	Ness	With	Agt	Without	Inf3	Become	Related	Dim	Acquire	Ly
Cluster1	**0.98**	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cluster2	**0.96**	0.01	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.01
Cluster3	**0.39**	0.20	0.16	0.06	0.02	0.00	0.08	0.05	0.03	0.00
Cluster4	0.15	**0.43**	0.19	0.09	0.06	0.00	0.03	0.04	0.00	0.01
Cluster5	0.01	**0.60**	0.07	**0.31**	0.00	0.00	0.02	0.00	0.00	0.00
Cluster6	0.08	**0.57**	0.08	0.19	0.01	0.00	0.03	0.03	0.01	0.00
Cluster7	0.12	**0.36**	**0.35**	0.12	0.00	0.00	0.01	0.02	0.00	0.00
Cluster8	0.06	**0.39**	0.07	**0.33**	0.00	0.00	0.08	0.03	0.04	0.00
Cluster9	0.05	0.21	**0.69**	0.03	0.00	0.00	0.01	0.00	0.00	0.00
Cluster10	0.03	0.07	0.16	0.02	**0.70**	0.00	0.01	0.01	0.01	0.00

Table 6.4 Clustering Results of K-means clustering with Cosine Distance and Normalized Word Vectors

	Ness	With	Agt	Without	Acquire	Inf3	Become	Related	Dim	Ly
Cluster1	**0.97**	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00
Cluster2	**0.96**	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.01	0.00
Cluster3	**0.93**	0.01	0.02	0.00	0.00	0.00	0.02	0.00	0.00	0.01
Cluster4	0.01	**0.56**	0.06	**0.34**	0.01	0.00	0.00	0.02	0.00	0.00
Cluster5	0.05	**0.50**	**0.23**	**0.18**	0.00	0.00	0.00	0.01	0.03	0.00
Cluster6	0.16	**0.44**	**0.20**	0.08	0.01	0.04	0.00	0.04	0.02	0.00
Cluster7	0.16	**0.38**	**0.29**	0.12	0.00	0.01	0.00	0.01	0.02	0.00
Cluster8	0.16	**0.33**	0.11	**0.20**	0.03	0.01	0.00	0.10	0.05	0.00
Cluster9	0.02	0.19	**0.74**	0.03	0.00	0.00	0.00	0.01	0.00	0.00
Cluster10	0.06	0.05	0.15	0.01	0.01	**0.69**	0.00	0.02	0.01	0.00

6.3 Clustering with Agglomerative Clustering

We used Euclidean as distance metric and ward as linkage method while performing clustering using Agglomerative clustering. Table 6.5 and Table 6.6 illustrates the clustering results for 10 clusters using original and normalized word vectors respectively.

Table 6.5 Clustering Results of Agglomerative Clustering

	Ness	With	Agt	Without	Inf3	Become	Related	Dim	Acquire	Ly
Cluster1	**0.93**	0.03	0.00	0.02	0.01	0.00	0.00	0.00	0.00	0.00
Cluster2	**0.90**	0.03	0.03	0.01	0.01	0.01	0.00	0.01	0.00	0.00
Cluster3	0.15	**0.35**	**0.33**	0.11	0.01	0.00	0.00	0.02	0.00	0.00
Cluster4	0.06	**0.49**	0.09	**0.30**	0.00	0.00	0.02	0.01	0.02	0.00
Cluster5	0.17	0.17	0.16	0.07	**0.38**	0.00	0.01	0.02	0.01	0.00
Cluster6	**0.20**	0.17	0.17	0.13	**0.23**	0.00	0.00	0.10	0.00	0.00
Cluster7	0.14	**0.29**	**0.25**	**0.22**	**0.05**	0.00	0.00	0.02	0.00	0.00
Cluster8	0.10	0.10	**0.30**	0.20	0.00	0.00	0.00	0.20	0.00	0.00
Cluster9	0.00	0.00	**0.33**	0.00	**0.33**	0.00	0.00	**0.33**	0.00	0.00
Cluster10	**0.20**	**0.20**	**0.20**	**0.20**	**0.20**	0.00	0.00	0.00	0.00	0.00

Table 6.6 Clustering Results of Agglomerative Clustering with Normalized Word Vectors

	Ness	With	Agt	Without	Inf3	Become	Related	Dim	Acquire	Ly
Cluster1	**0.99**	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cluster2	**0.92**	0.02	0.02	0.00	0.01	0.01	0.00	0.00	0.00	0.00
Cluster3	**0.92**	0.03	0.00	0.02	0.01	0.00	0.01	0.01	0.00	0.00
Cluster4	0.04	**0.58**	0.10	**0.24**	0.00	0.00	0.02	0.01	0.01	0.00
Cluster5	0.05	**0.47**	0.08	**0.33**	0.00	0.00	0.04	0.01	0.01	0.00
Cluster6	0.12	**0.42**	0.11	0.05	**0.27**	0.00	0.01	0.02	0.00	0.00
Cluster7	0.18	**0.34**	**0.28**	0.11	0.03	0.00	0.03	0.03	0.01	0.00
Cluster8	0.12	**0.36**	**0.34**	0.12	0.02	0.00	0.02	0.01	0.00	0.00
Cluster9	0.05	**0.20**	**0.58**	0.11	0.01	0.00	0.03	0.02	0.00	0.00
Cluster10	0.15	**0.27**	**0.41**	0.13	0.01	0.00	0.01	0.03	0.00	0.00

6.4 Clustering with GMMs

Clustering is performed using GMM algorithm with normalized and non-normalized word vectors, results of the clustering are reported for both in Table 6.7 and Table 6.8.

Table 6.7 Clustering Results Using GMMs

	Ness	With	Agt	Without	Become	Inf3	Related	Dim	Acquire	Ly
Cluster1	**0.71**	0.12	0.09	0.04	0.00	0.01	0.01	0.01	0.00	0.00
Cluster2	0.09	**0.54**	0.15	0.14	0.00	0.00	0.06	0.01	0.00	0.00
Cluster3	0.06	**0.58**	0.04	**0.29**	0.00	0.00	0.01	0.00	0.02	0.00
Cluster4	0.02	**0.49**	0.08	**0.37**	0.00	0.00	0.01	0.00	0.02	0.00
Cluster5	0.11	**0.34**	0.14	**0.29**	0.00	0.03	0.03	0.06	0.00	0.00
Cluster6	0.14	**0.25**	**0.31**	**0.22**	0.00	0.01	0.03	0.02	0.02	0.00
Cluster7	0.18	0.19	**0.23**	0.14	0.00	0.16	0.04	0.07	0.00	0.00
Cluster8	0.00	0.00	**0.25**	0.00	0.00	**0.25**	0.00	**0.50**	0.00	0.00
Cluster9	0.11	0.03	0.16	0.01	0.01	**0.65**	0.01	0.01	0.02	0.00
Cluster10	0.09	0.16	**0.63**	0.07	0.00	0.00	0.00	0.02	0.01	0.00

Table 6.8 Clustering Results Using GMMs with normalized word vectors

	Ness	With	Agt	Without	Inf3	Become	Related	Dim	Acquire	Ly
Cluster1	**0.99**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cluster2	**0.36**	0.19	0.13	0.09	0.02	0.01	0.09	0.07	0.03	0.01
Cluster3	0.01	**0.94**	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cluster4	0.00	**0.59**	0.05	**0.34**	0.00	0.00	0.01	0.00	0.00	0.00
Cluster5	0.02	**0.56**	0.04	**0.33**	0.00	0.00	0.03	0.00	0.01	0.00
Cluster6	0.14	**0.46**	0.22	0.10	0.02	0.00	0.03	0.02	0.01	0.00
Cluster7	0.01	**0.45**	**0.31**	**0.23**	0.01	0.00	0.00	0.00	0.00	0.00
Cluster8	0.13	**0.39**	**0.29**	0.16	0.01	0.00	0.01	0.02	0.00	0.00
Cluster9	0.02	0.07	**0.89**	0.02	0.00	0.00	0.00	0.00	0.00	0.00
Cluster10	0.05	0.04	0.16	0.01	**0.73**	0.00	0.01	0.01	0.00	0.00

CHAPTER 7

SEMANTIC ANALYSIS OF CLUSTERING RESULTS

7.1 Overview

In Chapter 6 we clustered the data presented in Chapter 5 using different clustering algorithms with different configurations. In many cases, the suffix categories do not dominate a single cluster but are distributed among several. Our opinion is that derivation is not just a syntactical process, but semantics is also effective. In Turkish, the derivational affixes other than the structural ones we listed in Chapter 5, do not combine with every word regardless of them sharing the same lexical category. We think that this could be the result of the different semantic properties of the stem that the suffix is combining with holds. Our research question is if this is the case, whether the word vectors that are created using the FastText method can capture this difference. We already saw that some affix categories are distributed among several clusters. Below we investigate the semantic reason for that distribution. We used two different methods with different semantic resources while we perform our analysis. In section 7.2 we present our method and analysis results using the semantic resources Turkish and English propositional bank. In section 7.3, we present our analysis using the semantic resource as the UCCA framework. Following that we perform our analysis using the UD Turkish BOUN treebank in section 7.4.

7.2 Method of Semantic Investigation with Propositional Bank as Semantic Resource

We used a two-stage method for analyzing the semantic content of clusters. First, we extracted the possible thematic roles of the stem part of a derivational affix for exploring the semantic properties of the stem and if we can see a grouping explaining the selectivity of the suffixes. We used both the English Propositional Bank (Palmer et al.,2005) and the Turkish Propositional Bank (Şahin and Adalı, 2018), as our main sources for this first stage, assuming that at the level of thematic roles predicates do not show as much variation as surface predicates of a language.

Propositional bank is a project that contains the predicate-argument structure and semantic role labels of a verb. These role labels are taken as proxies for thematic roles, as also done by research on abstract meaning representation (AMR) (O'Gorman et al., 2018, Dorr, 1998).

To do the inverted exploration, we need the stems of Turkish verbs for which we can match thematic roles of counterparts in English proposition bank. Because of the fact that the propbank contains only the words that are syntactically classified as verbs, we included only the pairs which have verbs as their root word.

We performed basic syntactic analysis using the library Zemberek, analyzed only the roots of the pairs and excluded the ones which are not verbs. Some words can be both verbs or nouns in Turkish. For example, in the pair "boyalı-boya" (painted-paint) the word "boya" (paint) can be used both as the action of painting and the material paint as it is in English depending on the context it is used in. When we specifically look into this derivational pair, we see that the derived word "boyalı" (painted) is actually derived from the noun form of the word. For eliminating such pairs, we manually reviewed the pruned dataset. We used free python module googletrans for translating the verbs into English, which uses Google Translate Library API for performing translation. After removing the pairs that do not have their stem in the propbank, we were left with 75 derivational pairs for semantic analysis, 58 of them derived by getting a suffix from the suffix category Inf3.

In Figure 7.1, we illustrated the thematic roles of stems extracted from the English propbank with the remaining pairs. The rows of the table represent the derivational pairs and the columns are the representations of the thematic roles. The analysis is performed on the clusters that are clustered using agglomerative clustering with word vectors that are not normalized We chose this clustering method because most of the pairs are derived by getting a suffix that belongs to the Inf3 category, and as Table (6.5) illustrates, this suffix category is distributed in multiple clusters when this method of clustering is used. There is no visible thematic role relationship in between a suffix category and the stem of the derivational pair that will cause them to distribute into different clusters as Figure (7.1) shows.

	agent	theme	result	attribute	topic	recipient	experiencer	stimulus	predicate	patient	extent	location	product	material	initial_location	destination	instrument	source	beneficiary	asset	cause	co-agent	co-theme	pivot	co-patient	trajectory
1																										
enchant(büyüleyici-büyüle)																										
adopt(benimseyiş-benimse)	x	x	x	x																							
scream(haykırış-haykır)	x				x	x																				
believe(inanış-inan)	x	x		x			x	x	x																		
grow(büyüyüş-büyü)	x	x								x	x	x	x	x													
rise(yükseliş-yüksel)	x	x								x	x	x			x	x										
transport(taşıyıcı-taşı)	x	x													x	x										
spy(gözetleyici-gözetle)							x	x																		
define(tanımlayış-tanımla)	x	x		x																					
handle(işleyiş-işle)	x	x															x										
know(tanıyış-tanı)	x	x		x			x	x	x																		
2																										
transfer(aktarış-aktar)	x	x				x									x	x										
trust(güvenli-güven)	x	x				x	x	x																		
invoke(yakarış-yakar)																										
eject(çıkarış-çıkar)	x																	x								
3																										
rescue(kurtarış-kurtar)	x	x																x								
stop(duruş-dur)	x	x														x	x									
praise(övücü-öv)	x	x																							
burst(patlayış-patla)	x	x								x		x													
burst(fırlayış-fırla)	x	x								x		x													
walk(yürüyüş-yürü)		x										x													
distribute(dağıtıcı-dağıt)	x	x				x																			
feel(hissediş-hisset)	x	x		x			x	x	x			x													
win(kazanış-kazan)	x	x																x	x	x					
learn(öğreniş-öğren)	x	x			x													x								
wait(bekleyiş-bekle)		x																							
think(düşünsel-düşün)	x	x		x			x																		
suppose(sanlı-san)	x	x		x																					
say(söyleyiş-söyle)	x				x	x															x				
plunge(dalış-dal)		x		x						x	x					x									
make(yapış-yap)	x	x	x				x	x						x	x		x			x						
caress(okşayış-okşa)	x						x																		
give(veriş-ver)	x	x				x																x			
teach(eğitici-eğit)	x				x	x																				
translate(çeviriş-çevir)	x		x							x				x											
shine(parlayış-parla)	x	x										x													
print(basış-bas)	x	x																							
hear(duyuş-duy)	x	x					x	x										x								
call(arayış-ara)	x	x	x		x	x														x						
manage(yönetici-yönet)	x	x																							
listen(dinleyiş-dinle)	x	x					x	x																		
wrap(sarış-sar)	x	x														x									
dissipate(dağılış-dağıl)	x									x															
protect(koruyucu-koru)	x	x																						x			
escape(kurtuluş-kurtul)		x																							
cry(ağlayış-ağla)	x	x			x	x	x																		
fall(düşüş-düş)		x								x	x														
send(gönderiş-gönder)	x	x													x	x										
want(isteyiş-iste)	x	x	x																				x		
watch(izleyiş-izle)	x	x					x	x																		
knit(örüş-ör)	x									x			x	x											
play(oynayış-oyna)	x	x																							
fatigue(yorucu-yor)							x	x																		
rescue(kurtarıcı-kurtar)	x	x																x								
increase(artış-art)	x									x	x														
think(düşünüş-düşün)	x	x		x			x																		
tremble(titreyiş-titre)		x					x																		
listen(dinleyici-dinle)	x	x					x	x																		
diverge(sapış-sap)		x																						x			
live(yaşayış-yaşa)	x	x																							
4																										
mention(anış-an)	x				x	x																				
drink(içiş-iç)	x									x								x								
mention(anlı-an)	x				x	x																				
fly(uçucu-uç)	x	x													x										
stupid(akış-ak)																										
die(ölüş-öl)										x															
open(açış-aç)	x	x				x				x							x										
5																										
align(diziş-diz)																										
break(kırıcı-kır)	x	x					x			x							x								x	
laugh(güllü-gül)	x	x																							
stretch(geriş-ger)	x	x								x															
6																										
pass(geçiş-geç)	x	x				x									x	x				x						
7																										
shiver(ürperiş-ürper)	x						x																		
climb(tırmanış-tırman)		x		x						x	x	x														x
8																										
9																										
10																										
descend(iniş-in)		x										x														x

Figure 7.1 Thematic roles of the stems extracted from the English propbank

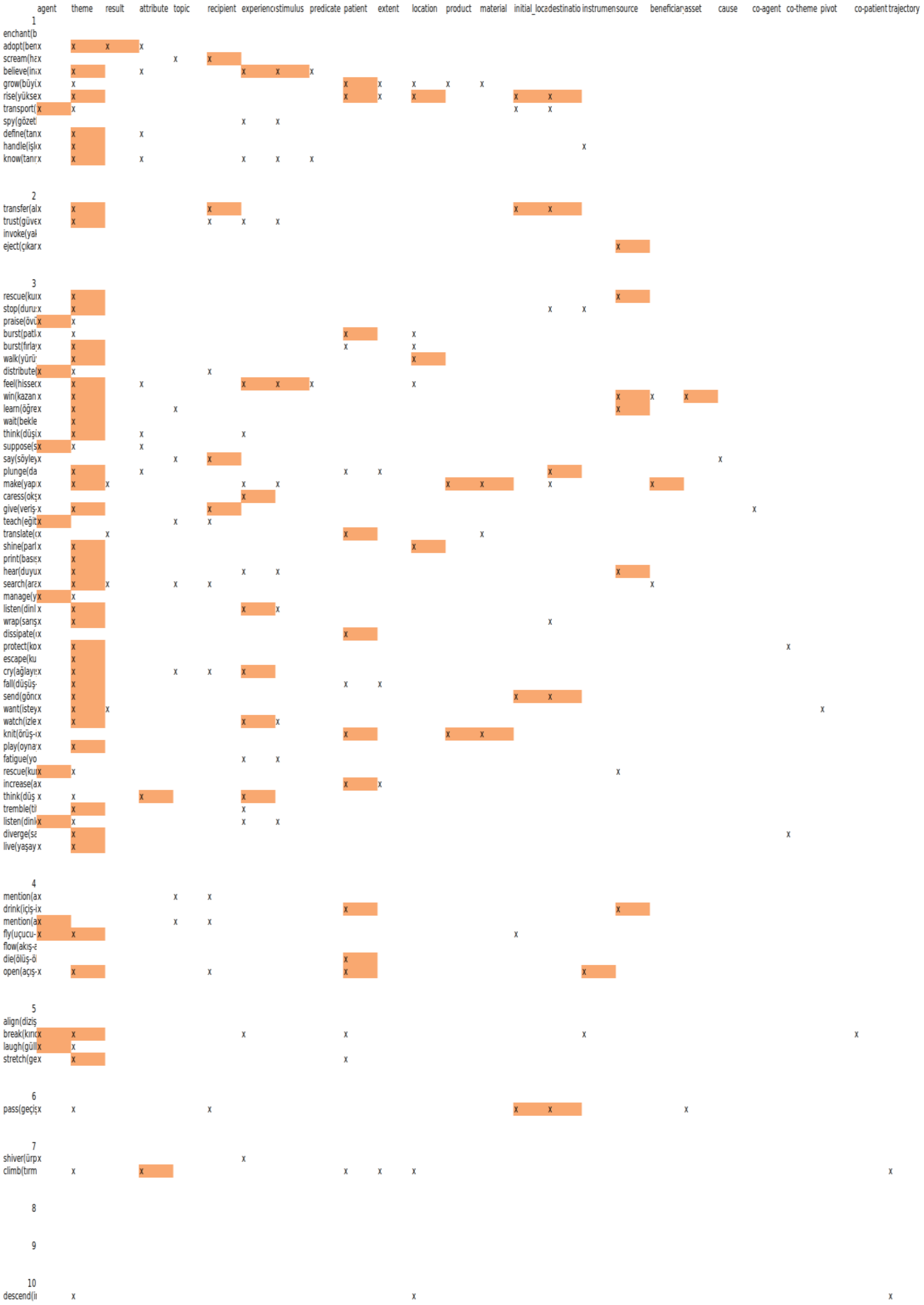
For the second and the last stage of the analysis we wanted to find the selectivity of the suffixes, that is, their preference of stems based on thematic roles. From the first stage as we explained in the previous paragraph, we find the thematic roles of a stem can have. In this stage, we find which roles a derived word can reference to within the possible thematic roles of its stem. For example,

(7.1) benimseyiş-benimse

Adamın davranışı benimseyişi (the man's adoption of the behavior)

As illustrated in (7.1), possible usage of the derived word in a pair is “adamın davranışı benimseyişi” (the man's adoption of the behavior). Here the word “benimseyiş” is selecting the word “davranış”. In Figure (7.1) we see that the stem of the pair “benimse” can have either “agent”, “theme”, “result” or “attribute” as its argument’s thematic role.

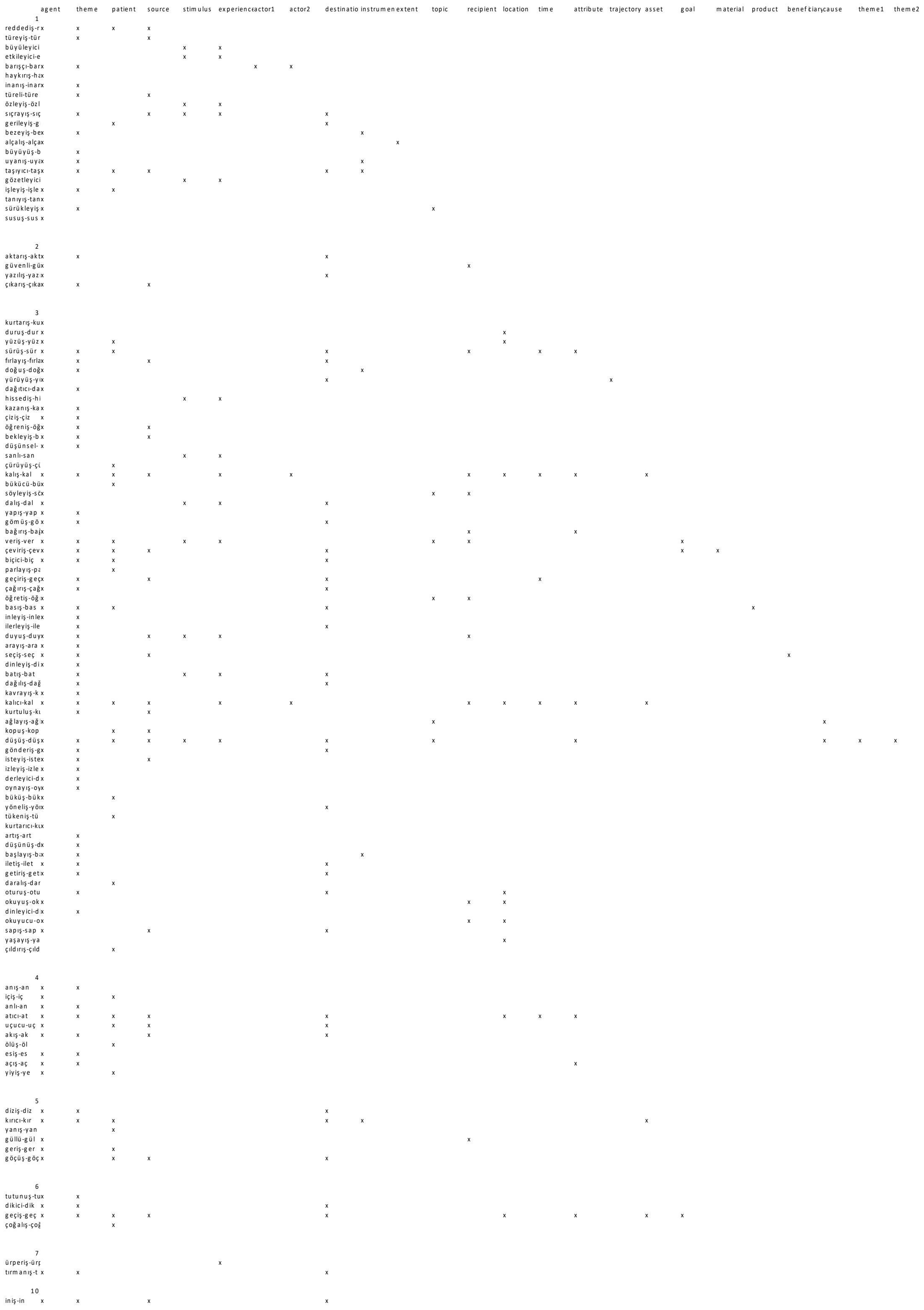
When we look at “davranış” in our example, we see that it can either be a theme or a result, but it cannot be agent or attribute. Therefore, the suffix “ış” (inf3) is selecting the thematic roles of result and theme. The approach is like the one we adopt when we find the vectors of the suffixes in distributional space. As we stated in the previous sections, we extracted the derived word’s vector from stem’s for finding the vector of the suffix. Similarly, in order to find the thematic roles that the suffix is selecting, we find the thematic roles the derived word can reference to from the roles the stem can have. With this method, we are constructing an inverted list for every suffix in a pair. Figure (7.2) highlights the roles suffixes are selecting on top of the roles of stems extracted from the English propbank for the clusters formed with the same algorithm we used in Figure (7.1), that is agglomerative clustering with word vectors which are not normalized.



	agent	theme	result	attribute	topic	recipient	experienc	stimulus	predicate	patient	extent	location	product	material	initial_loca	destinatio	instrumen	source	beneficiar	asset	cause	co-agent	co-theme	pivot	co-patient	trajectory
1																										
enchant(b																										
adopt(ben	x	**x**	**x**	x																						
scream(ha	x				x	**x**																				
believe(in	x	**x**		x			**x**	**x**	x																	
grow(büy	x	x								**x**	x	x	x	x												
rise(yükse	x	**x**								**x**	x	**x**			**x**	**x**										
transport(	**x**	x													x	x										
spy(gözet							x	x																		
define(tan	x	**x**		x																						
handle(işl	x	**x**															x									
know(tanı	x	**x**		x			x	x	x																	
2																										
transfer(al	x	**x**				**x**									**x**	**x**										
trust(güve	x	**x**				x	x	x																		
invoke(yal																										
eject(çıkar	x																	**x**								
3																										
rescue(kur	x	**x**																**x**								
stop(duru	x	**x**														x	x									
praise(övü	**x**	x																								
burst(patl	x	x								**x**		x														
burst(fırla	x	**x**								x		x														
walk(yürü		**x**										**x**														
distribute(	**x**	x				x																				
feel(hisse	x	**x**		x			**x**	**x**	x			x														
win(kazan	x	**x**																**x**	x	**x**						
learn(öğre	x	**x**			x													**x**								
wait(bekle		**x**																								
think(düşü	x	**x**		x			x																			
suppose(s	**x**	x		x																						
say(söyle	x				x	**x**															x					
plunge(da		**x**		x						x	x					**x**										
make(yap	x	**x**	x				x	x					**x**	**x**		x			**x**							
caress(okş	x						**x**																			
give(veriş	x	**x**				**x**																x				
teach(eğit	**x**				x	x																				
translate(	x		x							**x**				x												
shine(parl	x	**x**										**x**														
print(bası	x	**x**																								
hear(duyu	x	**x**					x	x										**x**								
search(ara	x	**x**	x		x	x													x							
manage(y	**x**	x																								
listen(dinl	x	**x**					**x**	x																		
wrap(sarış	x	**x**														x										
dissipate(	x									**x**																
protect(ko	x	**x**																					x			
escape(ku		**x**																								
cry(ağlay	x	**x**			x	x	**x**																			
fall(düşüş-		**x**								x	x															
send(gön	x	**x**													**x**	**x**										
want(iste	x	**x**	x																					x		
watch(izle	x	**x**					**x**	x																		
knit(örüş-	x									**x**			**x**	**x**												
play(oyna	x	**x**																								
fatigue(yo							x	x																		
rescue(kur	**x**	x																x								
increase(a	x									**x**	x															
think(düş	x	x		**x**			**x**																			
tremble(ti		**x**					x																			
listen(dinl	**x**	x					x	x																		
diverge(sa		**x**																					x			
live(yaşay	x	**x**																								
4																										
mention(a	x				x	x																				
drink(içiş-	x									**x**								**x**								
mention(a	**x**				x	x																				
fly(uçucu-	**x**	**x**													x											
flow(akış-a																										
die(ölüş-öl										**x**																
open(açış-	x	**x**				x				**x**							**x**									
5																										
align(diziş																										
break(kırı	**x**	**x**					x			x							x								x	
laugh(gül	**x**	x																								
stretch(ge	x	**x**								x																
6																										
pass(geçiş	x	x				x									**x**	**x**				x						
7																										
shiver(ürp	x						x																			
climb(tırm		x		**x**						x	x	x														x
8																										
9																										
10																										
descend(i		x										x														x

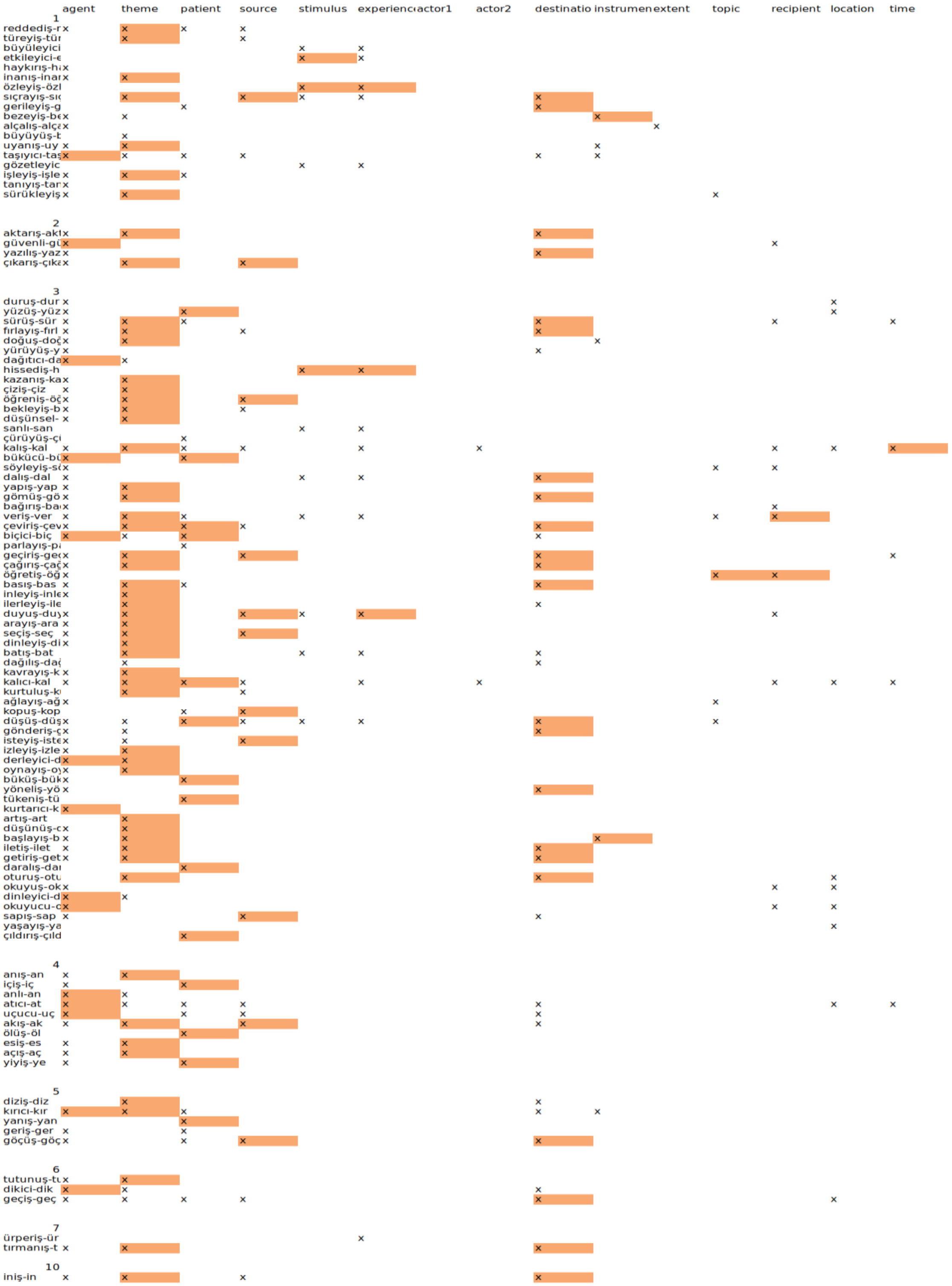
Figure 7.2 Highlighted thematic roles selected by suffixes from English propbank

We perform the same process with the same data, except the translation of the pair's part, using the Turkish propositional bank as our semantic resource. We had 109 derivational pairs matched with the Turkish propositional bank. Figure 7.3 shows the thematic roles of the stems. Similar to the Figure 7.1, there is no observable relationship between the roles of the stems and the distribution of the suffix categories in clusters. Figure 7.4 constructed using the inverted list method from the base of Figure 7.3.



	agent	theme	patient	source	stimulus	experience	actor1	actor2	destinatio	instrumen	extent	topic	recipient	location	time	attribute	trajectory	asset	goal	material	product	beneficiary	cause	theme1	theme2
1																									
reddediş-r	x	x	x	x																					
türeyiş-tür		x		x																					
büyüleyici					x	x																			
etkileyici-e					x	x																			
barışçı-bar	x	x					x	x																	
haykırış-ha	x																								
inanış-inar	x	x																							
türeli-türe		x		x																					
özleyiş-özl					x	x																			
sıçrayış-sıç		x		x	x	x			x																
gerileyiş-g			x						x																
bezeyiş-be	x	x								x															
alçalış-alça	x										x														
büyüyüş-b		x																							
uyanış-uya	x	x								x															
taşıyıcı-taş	x	x	x	x					x	x															
gözetleyici					x	x																			
işleyiş-işle	x	x	x																						
tanıyış-tan	x																								
sürükleyiş	x	x										x													
susuş-sus	x																								
2																									
aktarış-akt	x	x							x																
güvenli-gü	x												x												
yazılış-yaz	x								x																
çıkarış-çıka	x	x		x																					
3																									
kurtarış-ku	x																								
duruş-dur	x													x											
yüzüş-yüz	x		x											x											
sürüş-sür	x	x	x						x				x		x	x									
fırlayış-fırla	x	x		x					x																
doğuş-doğ	x	x								x															
yürüyüş-yı	x								x								x								
dağıtıcı-da	x	x																							
hissediş-hi					x	x																			
kazanış-ka	x	x																							
çiziş-çiz	x	x																							
öğreniş-öğ	x	x		x																					
bekleyiş-b	x	x		x																					
düşünsel-	x	x																							
sanlı-san					x	x																			
çürüyüş-çü			x																						
kalış-kal	x	x	x	x		x		x					x	x	x	x		x							
bükücü-bü	x		x																						
söyleyiş-sö	x											x	x												
dalış-dal	x				x	x			x																
yapış-yap	x	x																							
gömüş-gö	x	x							x																
bağırış-bağ	x												x			x									
veriş-ver	x	x	x		x	x						x	x						x						
çeviriş-çev	x	x	x	x					x										x	x					
biçici-biç	x	x	x						x																
parlayış-pa			x																						
geçiriş-geç	x	x		x					x						x										
çağırış-çağ	x	x							x																
öğretiş-öğ	x											x	x												
basış-bas	x	x	x						x												x				
inleyiş-inle	x	x																							
ilerleyiş-ile		x							x																
duyuş-duy	x	x		x	x	x							x												
arayış-ara	x	x																							
seçiş-seç	x	x		x																		x			
dinleyiş-di	x	x																							
batış-bat		x			x	x			x																
dağılış-dağ		x							x																
kavrayış-k	x	x																							
kalıcı-kal	x	x	x	x		x		x					x	x	x	x		x							
kurtuluş-ku		x		x																					
ağlayış-ağ	x											x											x		
kopuş-kop			x	x																					
düşüş-düş	x	x	x	x	x	x			x			x				x							x	x	x
gönderiş-g	x	x							x																
isteyiş-iste	x	x		x																					
izleyiş-izle	x	x																							
derleyici-d	x	x																							
oynayış-oy	x	x																							
büküş-bük	x		x																						
yöneliş-yö	x								x																
tükeniş-tü			x																						
kurtarıcı-ku	x																								
artış-art		x																							
düşünüş-d	x	x																							
başlayış-b	x	x								x															
iletiş-ilet	x	x							x																
getiriş-get	x	x							x																
daralış-dar			x																						
oturuş-otu		x							x					x											
okuyuş-ok	x												x	x											
dinleyici-di	x	x																							
okuyucu-o	x												x	x											
sapış-sap	x			x					x																
yaşayış-ya														x											
çıldırış-çıld			x																						
4																									
anış-an	x	x																							
içiş-iç	x		x																						
anlı-an	x	x																							
atıcı-at	x	x	x	x					x					x	x	x									
uçucu-uç	x		x	x					x																
akış-ak	x	x		x					x																
ölüş-öl			x																						
esiş-es	x	x																							
açış-aç	x	x														x									
yiyiş-ye	x		x																						
5																									
diziş-diz	x	x							x																
kırıcı-kır	x	x	x						x	x								x							
yanış-yan			x																						
güllü-gül	x												x												
geriş-ger	x		x																						
göçüş-göç	x		x	x					x																
6																									
tutunuş-tu	x	x																							
dikici-dik	x	x							x																
geçiş-geç	x	x	x	x					x					x		x		x	x						
çoğalış-çoğ			x																						
7																									
ürperiş-ürp						x																			
tırmanış-t	x	x							x																
10																									
iniş-in	x	x		x					x																

Figure 7.3 Thematic roles of the stems extracted from the Turkish propbank



	agent	theme	patient	source	stimulus	experienc	actor1	actor2	destinatio	instrumen	extent	topic	recipient	location	time
1															
reddediş-r	x	**x**	x	x											
türeyiş-tür		**x**		x											
büyüleyici					x	x									
etkileyici-e					**x**	x									
haykırış-hı	x														
inanış-inar	x	**x**													
özleyiş-özl					**x**	**x**									
sıçrayış-sıç		**x**		**x**	x	x			**x**						
gerileyiş-g			x						**x**						
bezeyiş-be	x	x								**x**					
alçalış-alça	x										x				
büyüyüş-b		x													
uyanış-uy	x	**x**								x					
taşıyıcı-taş	**x**	x	x	x					x	x					
gözetleyic					x	x									
işleyiş-işle	x	**x**	x												
tanıyış-tar	x														
sürükleyiş	x	**x**										x			
2															
aktarış-akt	x	**x**							**x**						
güvenli-gü	**x**												x		
yazılış-yaz	x								**x**						
çıkarış-çıka	x	**x**		**x**											
3															
duruş-dur	x													x	
yüzüş-yüz	x		**x**											x	
sürüş-sür	x	**x**	x						**x**				x		x
fırlayış-fırl	x	**x**		x					**x**						
doğuş-doğ	x	**x**								x					
yürüyüş-y	x								x						
dağıtıcı-da	**x**	x													
hissediş-h					**x**	**x**									
kazanış-ka	x	**x**													
çiziş-çiz	x	**x**													
öğreniş-öğ	x	**x**		**x**											
bekleyiş-b	x	**x**		x											
düşünsel-	x	**x**													
sanlı-san					x	x									
çürüyüş-çı			x												
kalış-kal	x	**x**	x	x		x		x					x	x	**x**
bükücü-bü	**x**		**x**												
söyleyiş-sö	x											x	x		
dalış-dal	x				x	x			**x**						
yapış-yap	x	**x**													
gömüş-gö	x	**x**							**x**						
bağırış-ba	x												x		
veriş-ver	x	**x**	x		x	x						x	**x**		
çeviriş-çev	x	**x**	**x**	x					**x**						
biçici-biç	**x**	x	**x**						x						
parlayış-pa			x												
geçiriş-geç	x	**x**		**x**					**x**						x
çağırış-çağ	x	**x**							**x**						
öğretiş-öğ	x											**x**	**x**		
basış-bas	x	**x**	x						**x**						
inleyiş-inle	x	**x**													
ilerleyiş-ile		**x**							x						
duyuş-duy	x	**x**		**x**	x	**x**							x		
arayış-ara	x	**x**													
seçiş-seç	x	**x**		**x**											
dinleyiş-di	x	**x**													
batış-bat		**x**			x	x			x						
dağılış-dağ		x							x						
kavrayış-k	x	**x**													
kalıcı-kal	x	**x**	**x**	x		x		x					x	x	x
kurtuluş-k		**x**		x											
ağlayış-ağ	x											x			
kopuş-kop			x	**x**											
düşüş-düş	x	x	**x**	x	x	x			**x**			x			
gönderiş-g	x	x							**x**						
isteyiş-iste	x	x		**x**											
izleyiş-izle	x	**x**													
derleyici-d	**x**	**x**													
oynayış-oy	x	**x**													
büküş-bük	x		**x**												
yöneliş-yö	x								**x**						
tükeniş-tü			**x**												
kurtarıcı-k	**x**														
artış-art		**x**													
düşünüş-d	x	**x**													
başlayış-b	x	**x**								**x**					
iletiş-ilet	x	**x**							**x**						
getiriş-get	x	**x**							**x**						
daralış-da			**x**												
oturuş-otu		**x**							**x**					x	
okuyuş-ok	x												x	x	
dinleyici-d	**x**	x													
okuyucu-o	**x**												x	x	
sapış-sap	x			**x**					x						
yaşayış-ya														x	
çıldırış-çıld			**x**												
4															
anış-an	x	**x**													
içiş-iç	x		**x**												
anlı-an	**x**	x													
atıcı-at	**x**	x	x	x					x					x	x
uçucu-uç	**x**		x	x					x						
akış-ak	x	**x**		**x**					x						
ölüş-öl			**x**												
esiş-es	x	**x**													
açış-aç	x	**x**													
yiyiş-ye	x		**x**												
5															
diziş-diz		**x**							x						
kırıcı-kır	**x**	**x**	x						x	x					
yanış-yan			**x**												
geriş-ger	x		x												
göçüş-göç	x		x	**x**					**x**						
6															
tutunuş-tu	x	**x**													
dikici-dik	**x**	x							x						
geçiş-geç	x	x	x	x					**x**					x	
7															
ürperiş-ür						x									
tırmanış-t	x	**x**							**x**						
10															
iniş-in	x	**x**		x					**x**						

Figure 7.4 Highlighted thematic roles selected by suffixes from Turkish propbank

From Figure 7.2, some semantic patterns can be observed (see Table 5.1 for suffix cluster specifications):

- The role experiencer selected by the suffixes Inf3 are gathered in the same cluster.

- The roles source, initial_location and destination are all selected by Inf3 suffixes, but they are spread across different clusters, they do not seem to be affecting how clusters are formed.

- The affixes that can have the thematic role agent are mostly belong to the Agt suffix category, regardless of the cluster they are distributed in.

Figure 7.4 supports that the role destination is selected by the Inf3 suffixes and the thematic role agent is mostly selected by the Agt suffixes. Other observations we listed above are not visible in this figure.

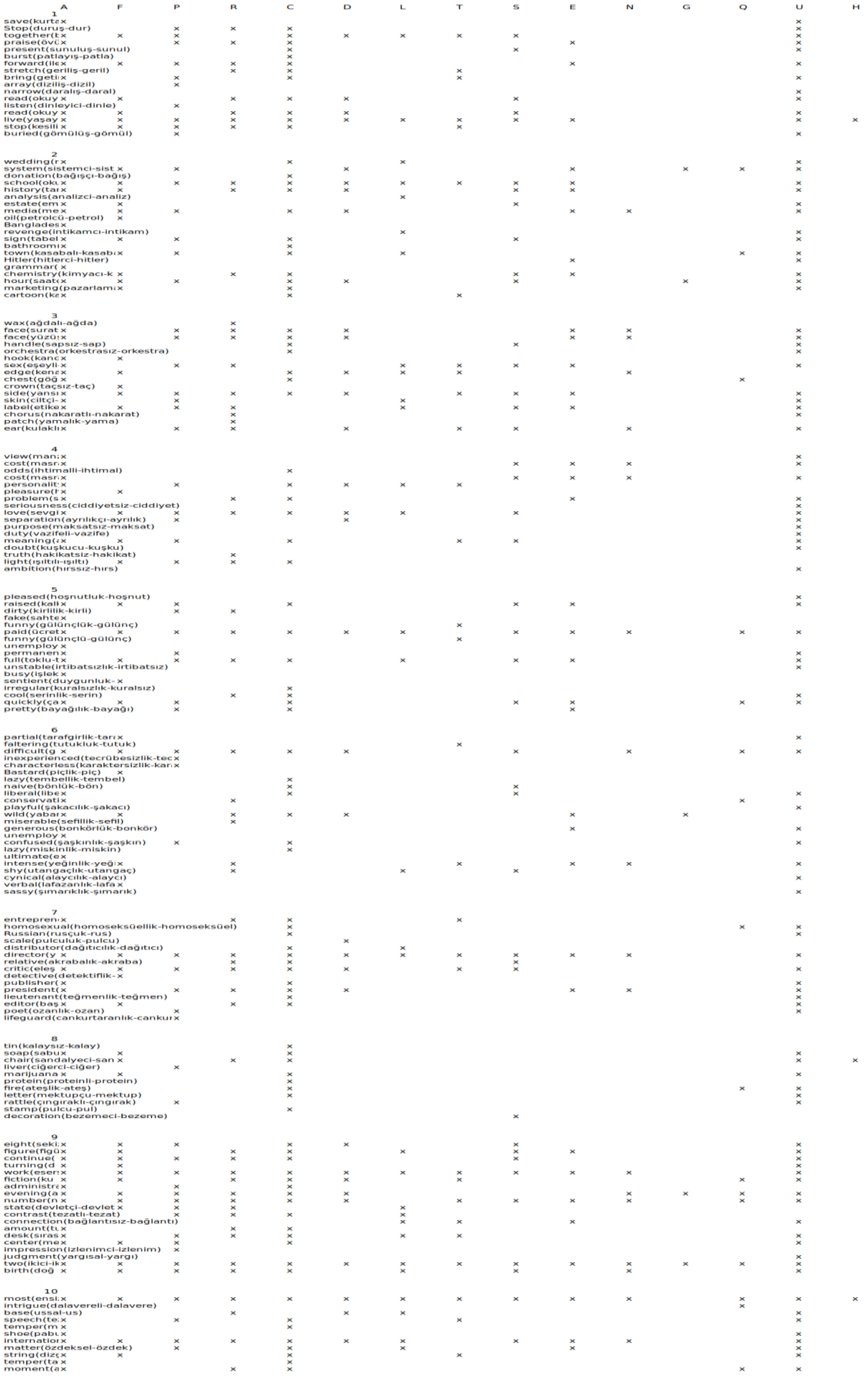
There is no visible indicator that shows the selected thematic roles of the suffixes are affecting the similarity between the obtained word vectors of suffixes other than the first item we listed above. Therefore, based on the analysis we performed, it isn’t possible to say similar word vectors of the suffixes having the same suffix category also have similar thematic roles, nor selected thematic roles affect the word vector of a suffix.

7.3 Method of Semantic Investigation with the UCCA framework as Semantic Resource

We used Abend and Rappoport (2013) in English as our semantic resource which is publicly available online. It contains 158690 total number of tokens annotated. The UCCA framework annotates the tokens regardless of their grammatical category, which removes the restriction we faced while using the propositional bank, which is analyzing the pairs that are only derived from verbs. We used the golden annotated dataset in English because currently, the golden annotated dataset for Turkish is not available. We adopted the similar assumption that we did in Chapter 7.2, that the roles do not vary much between languages. For translating the pairs into English, we used Google Translate Library API as we did in Chapter 7.2. We used the clusters formed using the k-means algorithm with Euclidean distance metric and normalized word vectors.

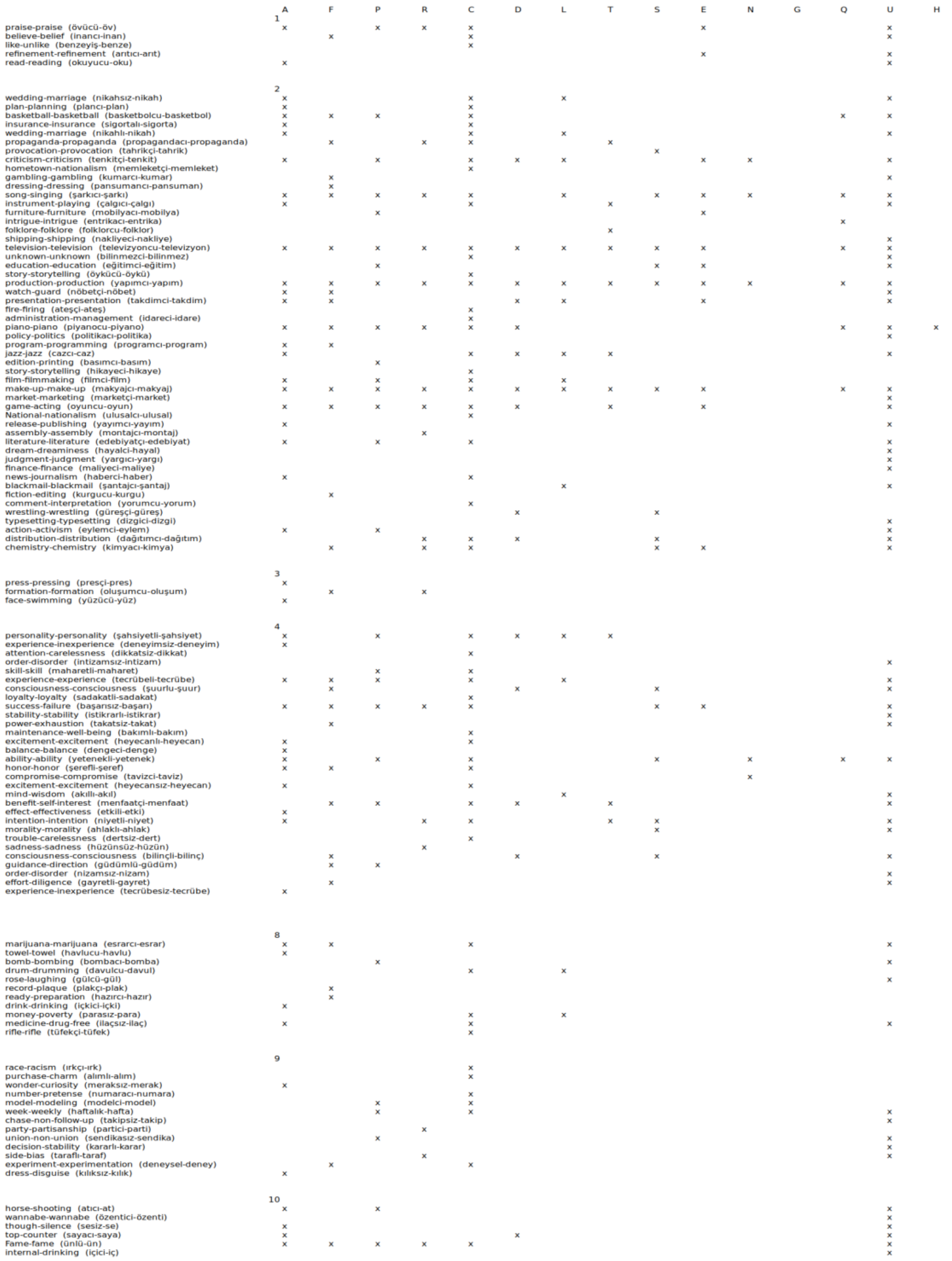
We marked the labels of the stems that matched with the framework as the first stage of our investigation with this resource for seeing the semantic properties of the stem that might cause the affix to combine with it, as we explained in the previous chapters. We end up with 2371 derivational pairs to work with. Figure 7.5 shows randomly selected portions of data from each cluster. Just like the tables presented in Chapter 7.2, the rows of the tables represent the derivational pairs and the columns are the representations of the UCCA labels. As the figure illustrates, the effect of the semantics of the stem cannot be seen from this analysis.

As the second stage of the investigation, we find the labels of the derived words and mark the labels that are both a label of the derived word and the stem of a pair in order to find the labels selected by the suffixes. When the pairs that include the tokens that could not be found in the UCCA framework and the pairs that did not have intersecting labels were extracted from the dataset, we end up with 120 pairs to analyze. We illustrate the analysis of these 120 pairs in Figure 7.6. As can be seen in the figure, after using the method with the UCCA framework, no semantic pattern emerged. With the analysis coming from Chapter 7.2 along with the analysis demonstrated in this chapter, it is possible to say that with the semantic resources in hand using our presented methods, it is not possible to explain the distribution of the derivational suffix embeddings.



	A	F	P	R	C	D	L	T	S	E	N	G	Q	U	H
1															
save(kurt	x													x	
Stop(duruş-dur)			x	x	x									x	
together(t	x	x	x		x	x	x	x	x					x	
praise(övü	x		x	x	x					x				x	
present(sunuluş-sunul)					x				x					x	
burst(patlayış-patla)					x										
forward(ile	x	x	x	x	x					x				x	
stretch(geriliş-geril)				x	x			x							
bring(geti	x		x		x			x						x	
array(diziliş-dizil)			x												
narrow(daralış-daral)														x	
read(okuy	x	x		x	x	x			x					x	
listen(dinleyici-dinle)			x												
read(okuy	x	x		x	x	x			x					x	
live(yaşay	x	x	x	x	x	x	x	x	x	x				x	x
stop(kesili	x	x	x	x	x			x							
buried(gömülüş-gömül)			x											x	
2															
wedding(r	x				x		x							x	
system(sistemci-sist		x	x			x				x		x	x	x	
donation(bağışçı-bağış)					x										
school(oku	x	x	x	x	x	x	x	x	x	x				x	
history(tar	x	x		x	x	x			x	x				x	
analysis(analizci-analiz)							x								
estate(em	x	x							x					x	
media(me	x	x	x		x	x				x	x			x	
oil(petrolcü-petrol)		x													
Banglades	x														
revenge(intikamcı-intikam)							x							x	
sign(tabel	x	x	x		x				x					x	
bathroom	x				x										
town(kasabalı-kasabı		x	x		x		x						x	x	
Hitler(hitlerci-hitler)										x				x	
grammar(	x													x	
chemistry(kimyacı-k		x		x	x				x	x				x	
hour(saat	x	x	x		x	x			x			x		x	
marketing(pazarlam		x			x									x	
cartoon(ka	x				x			x							
3															
wax(ağdalı-ağda)				x											
face(surat	x		x	x	x	x				x	x			x	
face(yüzü	x		x	x	x	x				x	x			x	
handle(sapsız-sap)					x				x					x	
orchestra(orkestrasız-orkestra)					x									x	
hook(kanc	x	x													
sex(eşeyli	x		x	x			x	x	x	x				x	
edge(ken	x	x			x	x	x	x			x				
chest(göğ	x				x								x		
crown(taçsız-taç)		x													
side(yansı	x	x	x	x	x	x		x	x	x				x	
skin(ciltçi-	x		x				x							x	
label(etike	x	x	x	x			x		x	x				x	
chorus(nakaratlı-nakarat)				x										x	
patch(yamalık-yama)				x										x	
ear(kulaklı	x		x	x		x		x	x		x			x	
4															
view(man	x													x	
cost(masr	x								x	x	x			x	
odds(ihtimalli-ihtimal)					x										
cost(masr	x								x	x	x			x	
personalit	x		x		x	x	x	x							
pleasure(	x	x													
problem(s	x			x	x					x				x	
seriousness(ciddiyetsiz-ciddiyet)														x	
love(sevgi	x	x	x	x	x	x	x		x					x	
separation(ayrılıkçı-ayrılık)			x			x								x	
purpose(maksatsız-maksat)														x	
duty(vazifeli-vazife)														x	
meaning(	x	x	x		x			x	x					x	
doubt(kuşkucu-kuşku)														x	
truth(hakikatsiz-hakikat)				x											
light(ışıltılı-ışıltı)		x	x	x	x										
ambition(hırssız-hırs)														x	
5															
pleased(hoşnutluk-hoşnut)														x	
raised(kal	x	x	x		x				x	x				x	
dirty(kirlilik-kirli)			x	x											
fake(sahte	x														
funny(gülünçlük-gülünç)								x							
paid(ücret	x	x	x	x	x	x	x		x	x	x		x	x	
funny(gülünçlü-gülünç)								x							
unemploy	x														
permanen	x		x											x	
full(toklu-t	x	x	x	x	x		x		x	x				x	
unstable(irtibatsızlık-irtibatsız)														x	
busy(işlek	x														
sentient(duygunluk-		x													
irregular(kuralsızlık-kuralsız)					x										
cool(serinlik-serin)				x	x									x	
quickly(ça	x	x	x		x				x	x			x	x	
pretty(bayağılık-bayağı)			x		x					x					
6															
partial(tarafgirlik-tar		x												x	
faltering(tutukluk-tutuk)								x							
difficult(g	x	x	x	x	x	x			x		x		x	x	
inexperienced(tecrübesizlik-tec			x												
characterless(karaktersizlik-kar			x												
Bastard(piçlik-piç)		x													
lazy(tembellik-tembel)					x										
naive(bönlük-bön)					x				x						
liberal(libe	x				x				x					x	
conservati	x			x									x		
playful(şakacılık-şakacı)														x	
wild(yaba	x	x		x	x	x				x		x			
miserable(sefillik-sefil)				x											
generous(bonkörlük-bonkör)										x				x	
unemploy	x														
confused(şaşkınlık-şaşkın)			x		x									x	
lazy(miskinlik-miskin)					x										
ultimate(e	x														
intense(yeğinlik-yeği		x		x				x		x	x			x	
shy(utangaçlık-utangaç)				x			x		x						
cynical(alaycılık-alaycı)														x	
verbal(lafazanlık-lafa		x													
sassy(şımarıklık-şımarık)														x	
7															
entrepren	x			x	x			x							
homosexual(homoseksüellik-homoseksüel)					x								x	x	
Russian(rusçuk-rus)					x									x	
scale(pulculuk-pulcu)						x									
distributor(dağıtıcılık-dağıtıcı)					x		x								
director(y	x	x	x	x	x	x	x	x	x	x	x			x	
relative(akrabalık-akraba)				x	x				x						
critic(eleş	x	x	x	x	x	x		x	x					x	
detective(detektiflik-		x													
publisher(	x				x									x	
president(	x		x	x	x	x				x	x			x	
lieutenant(teğmenlik-teğmen)					x									x	
editor(baş	x	x		x	x									x	
poet(ozanlık-ozan)			x											x	
lifeguard(cankurtaranlık-cankur			x												
8															
tin(kalaysız-kalay)					x										
soap(sabu	x	x			x									x	
chair(sandalyeci-san		x		x	x									x	x
liver(ciğerci-ciğer)			x												
marijuana	x	x			x									x	
protein(proteinli-protein)					x										
fire(ateşlik-ateş)					x								x	x	
letter(mektupçu-mektup)					x									x	
rattle(çıngıraklı-çıngırak)			x											x	
stamp(pulcu-pul)					x										
decoration(bezemeci-bezeme)									x						
9															
eight(seki	x	x	x		x	x			x					x	
figure(figü	x	x		x	x		x		x	x				x	
continue(	x	x	x	x	x				x					x	
turning(d	x	x												x	
work(eser	x	x	x	x	x	x	x	x	x	x	x			x	
fiction(ku	x	x		x	x	x		x					x	x	
administr	x		x		x										
evening(a	x	x	x	x	x	x					x	x	x	x	
number(n	x	x	x	x	x	x		x	x	x	x		x	x	
state(devletçi-devlet		x	x	x			x								
contrast(tezatlı-tezat)			x	x	x		x								
connection(bağlantısız-bağlantı)							x	x		x				x	
amount(tu	x			x	x										
desk(sıras	x		x	x	x		x	x						x	
center(me	x	x	x		x									x	
impression(izlenimci-izlenim)			x												
judgment(yargısal-yargı)														x	
two(ikici-ik	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
birth(doğ	x	x	x	x	x		x		x		x			x	
10															
most(ensi	x	x	x	x	x	x	x	x	x	x	x		x	x	x
intrigue(dalavereli-dalavere)													x		
base(ussal-us)				x		x	x							x	
speech(te	x		x		x			x						x	
temper(m	x				x									x	
shoe(pab	x													x	
internatio	x	x	x	x	x	x	x		x	x	x			x	
matter(özdeksel-özdek)			x		x		x			x				x	
string(diz	x	x			x			x						x	
temper(ta	x				x										
moment(a	x			x	x								x	x	

Figure 7.5 UCCA labels of the stems extracted from the English UCCA framework



	A	F	P	R	C	D	L	T	S	E	N	G	Q	U	H
1															
praise-praise (övücü-öv)	x		x	x	x					x				x	
believe-belief (inancı-inan)		x			x									x	
like-unlike (benzeyiş-benze)					x										
refinement-refinement (arıtıcı-arıt)										x				x	
read-reading (okuyucu-oku)	x													x	
2															
wedding-marriage (nikahsız-nikah)	x				x		x							x	
plan-planning (plancı-plan)	x				x										
basketball-basketball (basketbolcu-basketbol)	x	x	x		x								x	x	
insurance-insurance (sigortalı-sigorta)	x				x										
wedding-marriage (nikahlı-nikah)	x				x		x							x	
propaganda-propaganda (propagandacı-propaganda)		x		x	x			x							
provocation-provocation (tahrikçi-tahrik)									x						
criticism-criticism (tenkitçi-tenkit)	x		x		x	x	x			x	x			x	
hometown-nationalism (memleketçi-memleket)					x										
gambling-gambling (kumarcı-kumar)		x												x	
dressing-dressing (pansumancı-pansuman)		x													
song-singing (şarkıcı-şarkı)	x	x	x	x	x		x		x	x	x		x	x	
instrument-playing (çalgıcı-çalgı)	x				x			x						x	
furniture-furniture (mobilyacı-mobilya)			x							x					
intrigue-intrigue (entrikacı-entrika)													x		
folklore-folklore (folklorcu-folklor)								x							
shipping-shipping (nakliyeci-nakliye)														x	
television-television (televizyoncu-televizyon)	x	x	x	x	x	x	x	x	x	x			x	x	
unknown-unknown (bilinmezci-bilinmez)					x									x	
education-education (eğitimci-eğitim)			x						x	x				x	
story-storytelling (öykücü-öykü)					x										
production-production (yapımcı-yapım)	x	x	x	x	x	x	x	x	x	x	x		x	x	
watch-guard (nöbetçi-nöbet)	x	x												x	
presentation-presentation (takdimci-takdim)	x	x				x	x			x				x	
fire-firing (ateşçi-ateş)					x										
administration-management (idareci-idare)					x										
piano-piano (piyanocu-piyano)	x	x	x	x	x	x							x	x	x
policy-politics (politikacı-politika)														x	
program-programming (programcı-program)	x	x													
jazz-jazz (cazcı-caz)	x				x	x	x	x						x	
edition-printing (basımcı-basım)			x												
story-storytelling (hikayeci-hikaye)					x										
film-filmmaking (filmci-film)	x		x		x		x								
make-up-make-up (makyajcı-makyaj)	x	x	x	x	x	x	x	x	x	x			x	x	
market-marketing (marketçi-market)														x	
game-acting (oyuncu-oyun)	x	x	x	x	x	x		x		x				x	
National-nationalism (ulusalcı-ulusal)					x										
release-publishing (yayımcı-yayım)	x													x	
assembly-assembly (montajcı-montaj)				x											
literature-literature (edebiyatçı-edebiyat)	x		x		x									x	
dream-dreaminess (hayalci-hayal)														x	
judgment-judgment (yargıcı-yargı)														x	
finance-finance (maliyeci-maliye)														x	
news-journalism (haberci-haber)	x				x										
blackmail-blackmail (şantajcı-şantaj)							x							x	
fiction-editing (kurgucu-kurgu)		x													
comment-interpretation (yorumcu-yorum)					x										
wrestling-wrestling (güreşçi-güreş)						x			x						
typesetting-typesetting (dizgici-dizgi)														x	
action-activism (eylemci-eylem)	x		x											x	
distribution-distribution (dağıtımcı-dağıtım)				x	x	x			x					x	
chemistry-chemistry (kimyacı-kimya)		x		x	x				x	x				x	
3															
press-pressing (presçi-pres)	x														
formation-formation (oluşumcu-oluşum)		x		x											
face-swimming (yüzücü-yüz)	x														
4															
personality-personality (şahsiyetli-şahsiyet)	x		x		x	x	x	x							
experience-inexperience (deneyimsiz-deneyim)	x														
attention-carelessness (dikkatsiz-dikkat)					x										
order-disorder (intizamsız-intizam)														x	
skill-skill (maharetli-maharet)			x		x										
experience-experience (tecrübeli-tecrübe)	x	x	x		x		x							x	
consciousness-consciousness (şuurlu-şuur)		x				x			x					x	
loyalty-loyalty (sadakatli-sadakat)					x										
success-failure (başarısız-başarı)	x	x	x	x	x				x	x				x	
stability-stability (istikrarlı-istikrar)														x	
power-exhaustion (takatsiz-takat)		x												x	
maintenance-well-being (bakımlı-bakım)					x										
excitement-excitement (heyecanlı-heyecan)	x				x										
balance-balance (dengeci-denge)	x														
ability-ability (yetenekli-yetenek)	x		x		x				x		x		x	x	
honor-honor (şerefli-şeref)	x	x			x										
compromise-compromise (tavizci-taviz)											x				
excitement-excitement (heyecansız-heyecan)	x				x										
mind-wisdom (akıllı-akıl)							x							x	
benefit-self-interest (menfaatçi-menfaat)		x	x		x	x		x						x	
effect-effectiveness (etkili-etki)	x														
intention-intention (niyetli-niyet)	x			x	x			x	x					x	
morality-morality (ahlaklı-ahlak)									x					x	
trouble-carelessness (dertsiz-dert)					x										
sadness-sadness (hüzünsüz-hüzün)				x											
consciousness-consciousness (bilinçli-bilinç)		x				x			x					x	
guidance-direction (güdümlü-güdüm)		x	x												
order-disorder (nizamsız-nizam)														x	
effort-diligence (gayretli-gayret)		x												x	
experience-inexperience (tecrübesiz-tecrübe)	x														
8															
marijuana-marijuana (esrarcı-esrar)	x	x			x									x	
towel-towel (havlucu-havlu)	x														
bomb-bombing (bombacı-bomba)			x											x	
drum-drumming (davulcu-davul)					x		x								
rose-laughing (gülcü-gül)														x	
record-plaque (plakçı-plak)		x													
ready-preparation (hazırcı-hazır)		x													
drink-drinking (içkici-içki)	x														
money-poverty (parasız-para)					x		x								
medicine-drug-free (ilaçsız-ilaç)	x				x									x	
rifle-rifle (tüfekçi-tüfek)					x										
9															
race-racism (ırkçı-ırk)					x										
purchase-charm (alımlı-alım)					x										
wonder-curiosity (meraksız-merak)	x														
number-pretense (numaracı-numara)					x										
model-modeling (modelci-model)			x		x										
week-weekly (haftalık-hafta)			x		x									x	
chase-non-follow-up (takipsiz-takip)														x	
party-partisanship (partici-parti)				x											
union-non-union (sendikasız-sendika)			x											x	
decision-stability (kararlı-karar)														x	
side-bias (taraflı-taraf)				x										x	
experiment-experimentation (deneysel-deney)		x			x										
dress-disguise (kılıksız-kılık)	x														
10															
horse-shooting (atıcı-at)	x		x											x	
wannabe-wannabe (özentici-özenti)														x	
though-silence (sesiz-se)	x													x	
top-counter (sayacı-saya)	x					x								x	
Fame-fame (ünlü-ün)	x	x	x	x	x									x	
internal-drinking (içici-iç)														x	

Figure 7.6 UCCA labels selected by suffixes extracted from the English UCCA framework

7.4 Method of Semantic Investigation with the UD Turkish BOUN treebank as Semantic Resource

The UD Turkish BOUN treebank (Türk et al., 2020) is a resource that is available online which consists of 9757 annotated Turkish sentences having different topics. It is in CoNNL-U format, which contains many fields such as the POS tag, morphosyntactic analysis, and universal dependency relation of a word. Similar to the UCCA framework, this dataset does not have the restriction of the Propbank that we discussed in Chapter 7.1, which prevented us from analyzing the words whose grammatical category is not a verb. Our main focus in this thesis is the UD relations of the derived words and the stems UD Turkish BOUN treebank consists of. Let us look into the annotated sentences in the dataset closely that we briefly illustrated in Chapter 2.3 before presenting the overall analysis in table format.

Below are two sentences containing the word *gazeteci* (journalist) taken from the UD Turkish BOUN treebank.

(7.2)

```
# text = Gazeteci kadınların gazeteci erkeklerle evlenmesi daha da zordur,"diyorum.
1         Gazeteci  gazeteci  NOUN      Noun      Case=Nom|Number=Sing|Person=32        compound
          _         _
2         kadınların          kadın     ADJN      Adj       Case=Gen|Number=Plur|Person=35
          nmod:poss           _         _
3         gazeteci  gazeteci  NOUN      Noun      Case=Nom|Number=Sing|Person=34        nmod      _
          _
4         erkeklerleerkek     NOUN      NounCase=Ins|Number=Plur|Person=35            obl       _
          _
5         evlenmesi           evlen     VERB      Verb
          Case=Nom|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3|Polarity=Pos8    nsubj
          _         _
6         daha      daha      ADV       Adverb    _         8         advmod    _         _
7         da        da        CCONJ     Conj      _         6         advmod:emph         _         _
8-9       zordur    _         _         _         _         _         _         _         SpaceAfter=No
8         zor       zor       ADJN      Adj       Case=Nom|Number=Sing|Person=312       obj       _
          _
9         durdur    AUX       Zero      Mood=Imp|Number=Sing|Person=2|Polarity=Pos      8         cop
          _         _
10        ,         ,         PUNCT     Punc      _         8         punct     _         SpaceAfter=No
11        "         "         PUNCT     Punc      _         8         punct     _         _
12        diyorum   de        VERB      Verb
          Aspect=Prog|Number=Sing|Person=1|Polarity=Pos|Tense=Pres      0
          root_SpaceAfter=No
13        .         .         PUNCT     Punc      _         12        punct     _         SpacesAfter=\r\n
```

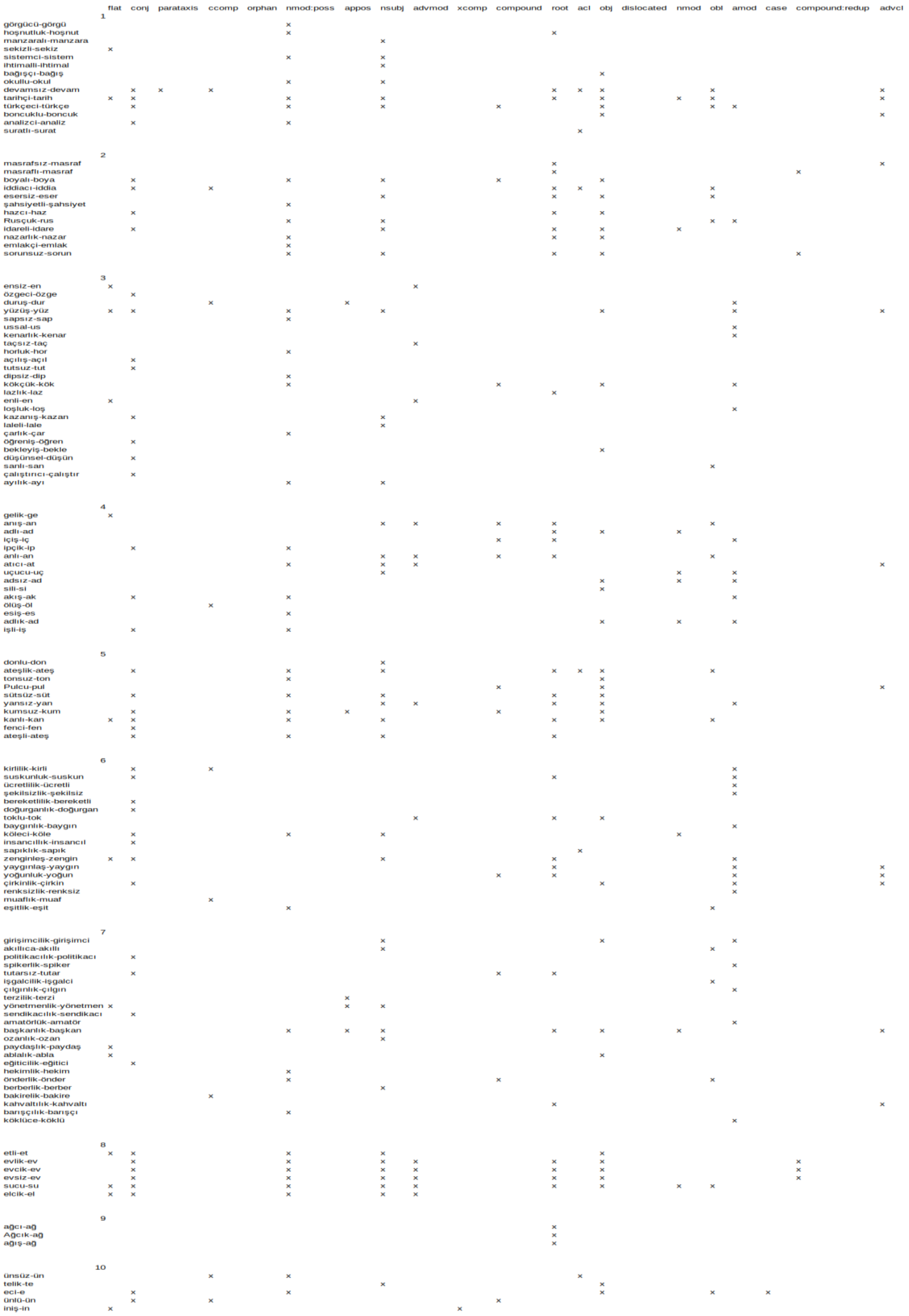
(7.3)

```
# text = 12 Ağustos Perşembe günü sabah saat 10.45'te, roman ve öykü yazarı, gazeteci Abbas Sayar, tedavi edilmekte
olduğu İzmir 9 Eylül Üniversitesi Tıp Fakültesi Hastanesi'nde öldü.

1       12          12          NUM     Anum    NumType=Card    2       nummod  _       _
2       Ağustos     ağustos     PROPN   Prop    Case=Nom|Number=Sing|Person=3   29      obl     _
        _
3       Perşembe    perşembe    NOUN    Noun    Case=Nom|Number=Sing|Person=3   2       list    _
        _
4       günügün     NOUN        Noun    Case=Nom|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3  2
        list        _           _
5       sabah       sabah       NOUN    Noun    Case=Nom|Number=Sing|Person=3   2       list    _
        _
6       saat        saat        NOUN    Noun    Case=Nom|Number=Sing|Person=3   7       obl     _
        _
7       10.         10          NUM     Anum    NumType=Ord     2       list    _
        SpaceAfter=No
8       45'te       45P         ROPNPropCase=Loc|Number=Sing|Person=3   7       list    _
        SpaceAfter=No
9       ,           ,           PUNCT   Punc    _       29      punct   _       _
10      roman       Roman       NOUN    Noun    Case=Nom|Number=Sing|Person=3   13      nmod:poss
        _           _
11      ve          ve          CCONJ   Conj    _       12      cc      _       _
12      öykü        öykü        NOUN    Noun    Case=Nom|Number=Sing|Person=3   10      conj    _
        _
13      yazarı      yazar       NOUN    Noun
        Case=Nom|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3  16      appos   _
        SpaceAfter=No
14      ,           ,           PUNCT   Punc    _       15      punct   _       _
15      gazeteci    gazeteci    NOUN    Noun    Case=Nom|Number=Sing|Person=3   13      conj    _
        _
16      Abbas       Abbas       PROPN   Prop    Case=Nom|Number=Sing|Person=3   29      nsubj   _
        _
17      Sayar       say         ADJN    Adj     Aspect=Hab|Number=Sing|Person=3|Polarity=Pos|Tense=Aor  16
        flat        _           SpaceAfter=No
18      ,           ,           PUNCT   Punc    _       29      punct   _       _
19      tedavi      tedavi      NOUN    Noun    Case=Nom|Number=Sing|Person=3   28      acl     _
        _
20      edilmekte   et          VERB    Verb
        Aspect=Prog|Number=Sing|Person=3|Polarity=Pos|Tense=Pres|Voice=Pass             19
        compound:lvc            _       _
21      olduğu      ol          VERB    Verb
        Aspect=Perf|Number[psor]=Sing|Person[psor]=3|Polarity=Pos|Tense=Past|VerbForm=Part      19
        aux                     _       _
22      İzmir       İzmir       PROPN   Prop    Case=Nom|Number=Sing|Person=3   28      nmod:poss
        _           _
23      9           9           NUM     Anum    NumType=Card    24      nummod  _       _
24      Eylül       eylül       NOUN    Noun    Case=Nom|Number=Sing|Person=3   25      nmod:poss
        _           _
25      Üniversitesi            üniversite NOUN Noun
        Case=Nom|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3  28      nmod:poss       _
        _
26      Tıp         tıp         VERB    Verb    Case=Nom|Number=Sing|Person=3   27      nmod:poss
        _           _
27      Fakültesi   fakülte     NOUN    Noun
        Case=Nom|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3  28      nmod:poss       _
        _
28      Hastanesi'nde           hastane NOUN    Noun
        Case=Loc|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3  29      obl     _       _
29      öldü        öl          VERB    Verb
        Aspect=Perf|Evident=Fh|Number=Sing|Person=3|Polarity=Pos|Tense=Past     0
        root_SpaceAfter=No
30      .           .           PUNCT   Punc    _       29      punct   _       SpacesAfter=\n
```

(7.2) shows the annotation of the sentence “*Gazeteci kadınların gazeteci erkeklerle evlenmesi daha zordur,” diyorum”* (I say it is more difficult for journalist women to marry journalist men) and (7.3) is the UD annotation of the sentence *“12 Ağustos Perşembe günü sabah saat 10.45'te, roman ve öykü yazarı, gazeteci Abbas Sayar, tedavi edilmekte olduğu İzmir 9 Eylül Üniversitesi Tıp Fakültesi Hastanesi'nde öldü.”* (On Thursday, August 12, at 10.45 am, novel and short story writer and journalist Abbas Sayar died in Izmir 9 Eylül University Medical Faculty Hospital, where he was being treated.) in CoNNL-U format. From the annotations of the sentences, which we explained its field in Chapter 2.3, we see that the word *gazeteci* (journalist) takes on different UD relation labels, which are compound, nmod, and conj respectively.

We analyzed our clusters by looking into the UD relations of the pairs, we find the relations of the stem and the derived word then we marked the intersecting relations of a pair, along with the relation of a stem of a pair only. Below in Figure 7.7 we illustrate the randomly selected portion of the results for the clusters formed using agglomerative clustering for the relations of the stems only, and in Figure 7.8 we show the results for the intersecting relations of the stems and the derived words, with the rows of the table are the representations of the derivational pairs and the columns of the table represent the UD relations.



	flat	conj	parataxis	ccomp	orphan	nmod:poss	appos	nsubj	advmod	xcomp	compound	root	acl	obj	dislocated	nmod	obl	amod	case	compound:redup	advcl
1																					
görgücü-görgü						×															
hoşnutluk-hoşnut						×						×									
manzaralı-manzara								×													
sekizli-sekiz	×																				
sistemci-sistem						×		×													
ihtimalli-ihtimal								×													
bağışçı-bağış														×							
okullu-okul						×		×													
devamsız-devam		×	×	×								×	×	×			×				×
tarihçi-tarih	×	×				×		×				×		×		×	×				×
türkçeci-türkçe		×				×		×			×			×			×	×			
boncuklu-boncuk														×							×
analizci-analiz		×				×															
suratlı-surat													×								
2																					
masrafsız-masraf												×									×
masraflı-masraf												×								×	
boyalı-boya		×				×		×			×			×							
iddiacı-iddia		×		×								×	×				×				
esersiz-eser								×				×		×			×				
şahsiyetli-şahsiyet						×															
hazcı-haz		×										×		×							
Rusçuk-rus						×		×									×	×			
idareli-idare		×						×				×		×		×					
nazarlık-nazar						×						×		×							
emlakçi-emlak						×															
sorunsuz-sorun						×		×				×		×						×	
3																					
ensiz-en	×								×												
özgeci-özge		×																			
duruş-dur				×			×											×			
yüzüş-yüz	×	×				×		×						×				×			×
sapsız-sap						×															
ussal-us																		×			
kenarlık-kenar																		×			
taçsız-taç									×												
horluk-hor						×															
açılış-açıl		×																			
tutsuz-tut		×																			
dipsiz-dip						×															
kökçük-kök						×					×			×				×			
lazlık-laz												×									
enli-en	×								×												
loşluk-loş																		×			
kazanış-kazan		×						×													
laleli-lale								×													
çarlık-çar						×															
öğreniş-öğren		×																			
bekleyiş-bekle														×							
düşünsel-düşün		×																			
sanlı-san																	×				
çalıştırıcı-çalıştır		×																			
ayılık-ayı						×		×													
4																					
gelik-ge	×																				
anış-an								×	×		×	×					×				
adlı-ad												×		×		×					
içiş-iç											×	×						×			
ipçik-ip		×				×															
anlı-an								×	×		×	×					×				
atıcı-at						×		×	×												×
uçucu-uç								×								×		×			
adsız-ad														×		×		×			
sili-si														×							
akış-ak		×				×												×			
ölüş-öl				×																	
esiş-es						×															
adlık-ad														×		×		×			
işli-iş		×				×															
5																					
donlu-don								×													
ateşlik-ateş		×				×		×				×	×	×			×				
tonsuz-ton						×								×							
Pulcu-pul											×			×							×
sütsüz-süt		×				×		×				×		×							
yansız-yan								×	×			×		×				×			
kumsuz-kum		×				×	×				×			×							
kanlı-kan	×	×				×		×				×		×			×				
fenci-fen		×																			
ateşli-ateş		×				×		×				×									
6																					
kirlilik-kirli		×		×														×			
suskunluk-suskun		×										×						×			
ücretlilik-ücretli																		×			
şekilsizlik-şekilsiz																		×			
bereketlilik-bereketli		×																			
doğurganlık-doğurgan		×																			
toklu-tok									×			×		×							
baygınlık-baygın																		×			
köleci-köle		×				×		×								×					
insancıllık-insancıl		×																			
sapıklık-sapık													×								
zenginleş-zengin	×	×						×				×						×			
yaygınlaş-yaygın												×						×			×
yoğunluk-yoğun											×	×						×			×
çirkinlik-çirkin		×												×				×			×
renksizlik-renksiz																		×			
muaflık-muaf				×																	
eşitlik-eşit						×											×				
7																					
girişimcilik-girişimci								×						×				×			
akıllıca-akıllı								×									×				
politikacılık-politikacı		×																			
spikerlik-spiker																		×			
tutarsız-tutar		×									×	×									
işgalcilik-işgalci																	×				
çılgınlık-çılgın																		×			
terzilik-terzi							×														
yönetmenlik-yönetmen	×						×	×													
sendikacılık-sendikacı		×																			
amatörlük-amatör																		×			
başkanlık-başkan						×	×	×				×		×		×					×
ozanlık-ozan								×													
paydaşlık-paydaş	×																				
ablalık-abla	×													×							
eğiticilik-eğitici		×																			
hekimlik-hekim						×															
önderlik-önder						×					×						×				
berberlik-berber								×													
bakirelik-bakire				×																	
kahvaltılık-kahvaltı												×									×
barışçılık-barışçı						×															
köklüce-köklü																		×			
8																					
etli-et	×	×				×		×						×							
evlik-ev		×				×		×	×			×		×						×	
evcik-ev		×				×		×	×			×		×						×	
evsiz-ev		×				×		×	×			×		×						×	
sucu-su	×	×				×		×	×			×		×		×	×				
elcik-el	×	×				×		×	×												
9																					
ağcı-ağ												×									
Ağcık-ağ												×									
ağış-ağ												×									
10																					
ünsüz-ün				×		×							×								
telik-te								×						×							
eci-e		×				×								×			×		×		
ünlü-ün		×		×							×										
iniş-in	×									×											

Figure 7.7 UD relations of the stems of the pairs extracted from the BOUN Turkish UD Treebank

	amod	advcl	compound	parataxis	xcomp	flat	dislocated	advmod	appos	conj	compound:redup	orphan	root	nmod	nmod:poss	ccomp	case	obl	obj	nsubj
1																				
tarihçi-tarih		×											×							
dizili-dizi	×																			
bakımsız-bakım										×										
girişimci-girişim																	×			
gazeteci-gazete											×			×						
kayalık-kaya												×								
gebelik-gebe														×						
bedensel-beden	×																			
radyocu-radyo																×				
şeriatçı-şeriat															×					
bayramlık-bayram	×																			
barışçı-barış															×					
amerikancı-amerikan								×												
deveci-deve																				×
şarkıcı-şarkı															×					
liseli-lise													×							
karmaşıklık-karmaşık											×									
eğitimci-eğitim									×											
tiyatrocu-tiyatro										×								×		
bağlılık-bağlı						×														
liralık-lira	×																			
incecik-ince	×																			
bilgisiz-bilgi									×											
devrimci-devrim														×						
yapımcı-yapım																	×			
hollandalı-hollanda																			×	
politikacı-politika										×										
şampiyonluk-şampiyor																×				
kıbrıslı-kıbrıs									×											
haftalık-hafta	×																			
özgürlükçü-özgürlük												×								
futbolcu-futbol													×							
detaylı-detay	×																			
üniversiteli-üniversite																×				
toplumsal-toplum							×					×								
darbeci-darbe													×							
tutumlu-tutum	×																			
oyuncu-oyun								×			×		×			×	×	×		
avrupalı-avrupa	×					×							×							
karıncık-karın															×					
elbiseli-elbise		×																		
silahlı-silah	×											×	×							
denizci-deniz								×												
güreşçi-güreş																		×		
sanatçı-sanat						×				×				×						
gündüzlü-gündüz	×																			
amerikalı-amerika	×							×												
motorlu-motor														×						
2																				
sevgisiz-sevgi							×													
anlamsız-anlam									×											
sınırlı-sınır	×																			
zararlı-zarar								×			×									
şanslı-şans														×						
renksiz-renk	×																			
faydasız-fayda										×										
düşünceli-düşünce									×											
güvenli-güven											×									
gözlü-göz	×								×			×								
boyutlu-boyut	×																			
önemsiz-önem	×									×										
renkli-renk	×							×						×						
isimli-isim													×							
boylu-boy	×								×											
şartlı-şart									×											
farksız-fark												×								
tarihli-tarih									×											
yönlü-yön	×																			
tepkisel-tepki										×										
önemli-önem	×									×	×			×						
ruhlu-ruh	×									×										
habersiz-haber	×																			
devirli-devir	×																			
zevkli-zevk									×			×								
adaletli-adalet															×					
hareketsiz-hareket											×									
farklı-fark	×									×		×								
tehlikeli-tehlike	×																			
resimli-resim	×																			
istekli-istek									×											
saygılı-saygı	×																			
mantıksız-mantık										×										
ayrıntılı-ayrıntı	×																			
tepkisiz-tepki										×										
eğitimsiz-eğitim	×									×										
katlı-kat											×									
başarılı-başarı								×												
şartsız-şart								×												
arzulu-arzu										×										
akıllı-akıl	×																			
köylü-köy												×				×				
başhekimlik-başhekim															×					
kararlı-karar										×										
görevli-görev												×				×	×			
meraklı-merak								×			×									
akılcı-akıl	×																			
fiziksel-fizik	×																			
uyumlu-uyum									×				×							
paralık-para														×						
kurallı-kural	×																			
sabırsız-sabır																×				
yönsüz-yön	×																			
saygısız-saygı	×																			
huzurlu-huzur										×										
eğitimli-eğitim	×																			
markalı-marka	×																			
nedensiz-neden											×									
3																				
duruş-dur																	×			
yapış-yap										×										
veriş-ver			×																	
bağlı-bağ									×											
golcü-gol													×							
köklü-kök	×																			
düşüş-düş																		×		
yorucu-yor												×								
4																				
adlı-ad	×												×							
akış-ak													×							
işsiz-iş									×											
5																				
tuzlu-tuz	×								×			×								
tuzsuz-tuz	×																			
saçlı-saç	×									×										
yağlı-yağ								×												
6																				
kölelik-köle													×							
ufaklı-ufak	×																			
geçiş-geç									×											
zenginlik-zengin							×													
temizlik-temiz										×										
iyimserlik-iyimser									×			×								
iyilik-iyi										×										
yorgunluk-yorgun	×								×											
kesinlik-kesin	×																			
7																				
akılıca-akıllı	×																			
başkanlık-başkan											×									
hekimlik-hekim															×					
milliyetçilik-milliyetçi																×				
arkadaşlık-arkadaş													×							
engelli-engel	×																			
oyunculuk-oyuncu											×									
öğretmenlik-öğretmen														×						
bilirkişilik-bilirkişi															×					
taraflı-taraf																	×			
öğrencilik-öğrenci										×										
işsizlik-işsiz										×										
valilik-vali													×							
8																				
sucu-su												×								
9																				
10																				
ünlü-ün																×				

Figure 7.8 UD relations selected by suffixes extracted from the BOUN Turkish UD Treebank

As the Figure 7.7 and Figure 7.8 shows, similar to the analyses performed before with other semantic data resources, any apparent relation between the derivational suffixes in the distributional vector space and the UD relations annotated in the BOUN Turkish Treebank is not visible.

CHAPTER 8

RESULTS

Illustrated results of clusters in Chapter 6 show that regardless of which clustering method or distance metric is used, whether the word vectors are normalized or not, the suffix categories “Ness”, “Agt”, “With”, “Without”, and “Inf3” are able to dominate one or more clusters with the presence of over 20%, whereas remaining suffix categories that are included in clustering do not have that property.

As we briefly stated in Chapter 6, this is expected since the mentioned categories in the previous paragraph have the highest frequencies in the dataset. This might indicate two things, (i) that the categories with lower frequencies do not contribute to derived words semantic content, (ii) that the currently used techniques of unsupervised clustering are not likely to perform well when there is a big difference in frequencies between the clustering data categories.

Another conclusion that does not manifest much between the methods of clustering is the regularity of different suffix categories. We see that “With” is usually spread across many clusters while others usually dominate a maximum of 3 clusters. This might indicate that With has more diverse meaning coverage, depending on the meaning of the attached root.

As we stated in Chapter 4, different clustering algorithms have their own strengths and weaknesses in terms of the shapes of data that will be clustered. Having similar results in each clustering algorithm with their different parameters is strong evidence of Turkish frequent derivational affixes with semantic content does contribute to the semantics of the derived word in distributional vector space. Slight differences are expected due to the different nature of the clustering algorithms.

We believe that derivation is a process that is mostly semantic rather than syntactical when the derivational affix that combines with the root carries a semantic content. We investigate if this can be captured by the word embeddings created in distributional vector space and if so, whether semantics is the reason why we see some derivational affixes to distribute in multiple clusters while others dominate relatively less or one. In Chapter 8, we perform our investigation using the presented methods in the chapter. We exhausted all of the Turkish semantic resources to our knowledge, that are annotated manually and available online. As the result of our investigations, we did not come across an apparent pattern that would indicate that the semantic differences are causing the similarity difference between word vectors. This could be happened because of two reasons, (i) currently present open-source semantic resources are not suitable for detecting such relation, (ii) semantics is not the main reason that causes the similarity difference in the word embeddings that are created using FastText method. The actual reason can be determined by some further investigation, some of which we present in Chapter 11.

CHAPTER 9

LIMITATIONS OF THE APPROACH

This thesis is an exploratory work on the effect of semantics on the similarity of derivational suffixes in distributional vector space. Although distributional semantics is a kind of semantics that gives some sort of similarity between words, morphemes, or syllables, it does not give us the actual semantics of the phrases we are looking at. Since the word embeddings are very popular and proven to be useful in many studies, we wanted to explore whether the vector similarity is coming from the similarities of the semantics of the phrases, and since OOV is a serious problem in agglomerative languages, we think that we might have a more robust solution to the problem of OOV by classifying the derivational suffixes not just by looking at their orthographic structure but also their semantic properties.

One of the limitations of the approach is that Turkish is a rich language with its words capable of having ambiguous morphological structures. We try to overcome this problem by excluding the words that have more than one morphological analysis in order to extract the word vectors of the purely derivational suffixes while forming our clustering dataset. Our approach shows that it is possible to obtain reliable word embeddings even by categorizing the derivational suffixes by their orthographies without considering their semantics.

Another limit that we were not able to overcome was the lack of semantic resources. Although the propositional bank is a rich semantic resource since it only contains the thematic roles of the verbs by its nature, it prevents us from using the whole of our dataset while trying to find a semantic pattern from our clustered suffix vectors. On the other hand, the UCCA framework did not have that drawback, but it did not have a golden annotated dataset for Turkish. Although we do not think the semantic roles of the words vary between languages, there is a possibility that sometimes things get lost in translation. The BOUN Turkish UD treebank do not hold any of the mentioned drawbacks, but it also did not reveal a semantic pattern that can explain the frequent derivational suffixes we used in the distributional vector space. Since this is exploratory work and the role of the semantics on clustering word embeddings of derivational suffixes is not yet to be proven, we do not claim that a semantic pattern will certainly be found if sufficient semantic resources are developed, but we do think that it is worth look such pattern in the future with the development of resources, we discuss this again on Chapter 11, Future Work.

CHAPTER 10

CONCLUSION

Our thesis aims to find whether it is possible to obtain reliable word embeddings of the frequent Turkish derivational suffixes and analyze them regarding their semantic properties. Based on several clustering methods with different similarity metrics it can be concluded that it is possible to generate reliable word vectors for many categories of frequent derivational suffixes of Turkish. Our results indicates that with the presented method and available semantic resources, it is not possible to show a direct effect of semantic properties on the similarity of derivational word vectors.

For agglutinating languages like Turkish, OOV is a serious problem of machine learning models even if the training corpus is very large, if the models are trained on continuous word representations, due to the productive nature of the languages. As we discussed in Chapter 1.2, models trained using characters instead of continuous representations do not suffer from OOV that much, but they prevent us from having information on words and morphemes themselves. With our applied method, we are able to find the word embeddings of the most frequent derivational suffixes of Turkish by simply subtracting the word vector of the derived word from its stem’s. We aimed to show the researchers who suffers from the problem of OOV that a simple method like this is able to produce word embeddings of reliable derivational affixes. In order to find the derivational pairs we used for obtaining suffix vectors, we used an open-source dataset of Turkish words that includes 88705 distinct Turkish words presented by Özbek (2012) with the open-source library Zemberek (Akın and Akın, 2007). Using a library like Zemberek eliminates the need for trained annotators and reduces the time that needs to be spent on annotation. For extracting the suffixes that are purely derivational, we included the pairs to our dataset that only have a single morphological analysis. We also removed the pairs that includes structural suffixes because they are able to combine with any stem and they do not have significant semantic contribution to the derived word. We also removed the suffix categories that have less than 20 instances. That left us with 10 distinct derivational categories.

We used multiple unsupervised clustering algorithms with different similarity metrics in order to evaluate the quality of word embeddings. Previous research performed on Turkish inflectional and derivational affixes (Güngör and Yıldız, 2017) concluded that inflectional affixes were more successful than derivational suffixes to detect in distributional vector space. Our results for each evaluation method which reported in confusion matrices shows that regardless of the clustering method and the parameters that are used, the categories that have higher frequencies, which are Ness, With, Inf3, Agt, and Without, are able to dominate one or more clusters whereas the categories with lower frequencies, which are Acquire, Ly and Dim do not hold that property. Looking at these results, it can be said that the categories with large frequencies affects the semantics of the derived word vectors and are able to have reliable word vectors.

Our results also shows that many of the derivational suffix categories are distributed between multiple clusters. We used three main semantic resources for investigating the semantic reasons that underlies under this distribution. The first ones we used in our thesis are the Propositional Banks both in Turkish (Şahin and Adalı, 2018), and English (Palmer et al.,2005), both are available online. We first look into the thematic roles of the stems from each semantic resource and saw that there is no direct visible connection between them and the distribution of the vectors of the suffix categories between clusters. We than used a method that we call *inverted list.* We analyzed the role selectivity of the frequent derivational suffixes with that method. The results of that analysis were not successful of explaining the distribution of the suffix vectors in clusters.

Second resource we used for the semantic analysis on clusters is the UCCA framework (Abend and Rappoport, 2013). Similar to the Propositional banks, the labels extracted from the UCCA framework did not show any indications that semantics can be used to explain the distribution of the embeddings of the suffix categories among the clusters. The third semantic resource we used is the BOUN Turkish UD treebank (Türk et al., 2020), which also did not reveal an apparent semantic pattern. This can indicate several things; (i) semantics, and by semantics we mean the roles, labels, relations that suffixes can select in a sentence and the stems of the suffixes can have, cannot be used to explain the similarity of word vectors of derivational suffixes extracted from the FastText models by the subtraction method, (ii) the semantic resources that are currently available are not enough to find such connection. Annotations has to go beneath the word level to be more use for understanding affix choice. The resources we used are annotated on full word forms in terms of their role in a sentence. For making explorations on our research topic, we believe root sense stem annotation is required. For instance, with the examples we made in Chapter 1.1, we see that “şekerci” (confectioner) is the person who sells “şeker” (candy) whereas “kurucu” (founder) do not hold such property. Having a resource with such information might reveal more in terms of the selection of the stems of the suffix categories. Information researchers might be helpful when such a resource wanted to be constructed.

CHAPTER 11

FUTURE WORK

The available resources of the field of semantics are constantly developing with many researchers working on the area. Re-performing the analysis with our proposed methods with newly developed resources might reveal the connection between semantics and the vector similarity of derivational affixes or confirm that there is no visible connection between formal semantics and the vector similarity of the derivational affixes.

In order to reduce the curse of the dimensionality in word embeddings, different dimension reduction techniques might be performed. This might help with getting better clustering results in the clustering stage. Also, we performed our clustering experiments with the assumption of having a cluster for each category, therefore we used 10 clusters in each technique. Optimizing the number of clusters might also reveal some more information regarding the distribution of the suffix categories in distributional vector space.

Another study might be performed by changing the machine learning model used for extracting the word embeddings of the suffixes. We performed our clustering experiments based on the suffix vectors extracted from a pre-trained model trained by the FastText algorithm, which uses the principle of looking at the neighbors of the words. Investigating the effect of semantics with the models trained using different training algorithms might behave differently and performing analysis by changing the model might help the researchers explore the area of derivational morphology under machine learning perspective. Using morpheme-based embeddings instead of full word forms might also reveal different perspectives and in our opinion worth experimenting.

In our study we investigate the research question of whether the distribution of the frequent derivational suffixes of Turkish can be explained by their semantics, therefore we used the word embedding vectors while investigating. Looking into how different dimensions of word embeddings encode semantics as in Şenel et al. (2018) did could be useful and in our opinion worth looking at as a future study.

REFERENCES

Abend, O., & Rappoport, A. (2013, August). Universal conceptual cognitive annotation (ucca). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 228-238).

Akın, A. A., & Akın, M. D. (2007). Zemberek, an open source nlp framework for turkic languages. Structure, 10, 1-5.

Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003). A neural probabilistic language model. *Journal of machine learning research*, *3*(Feb), 1137-1155.Dowty, Dowty, D. (1991). Thematic proto-roles and argument selection. Language, 67(3), 547– 619.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, *5*, 135-146.

Booij, G. (2008). Inflection and derivation. In *1. Halbband* (pp. 360-369). De Gruyter Mouton.

Bozşahin, C. (2018). Some Binary Concepts in Turkish Morphology. Unpublished manuscript.

Choi, S., Kim, T., Seol, J., & Lee, S. G. (2017). A syllable-based technique for word embeddings of korean words. *arXiv preprint arXiv:1708.01766*.

Chen, X., Xu, L., Liu, Z., Sun, M., & Luan, H. (2015, June). Joint learning of character and word embeddings. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Cotterell, R., & Schütze, H. (2019). Morphological word embeddings. arXiv preprint arXiv:1907.02423.

Çakıcı, R., Steedman, M., & Bozşahin, C. (2018). Wide-Coverage Parsing, Semantics, and Morphology. In *Turkish Natural Language Processing* (pp. 153-174). Springer, Cham.

De Marneffe, M. C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., & Manning, C. D. (2014, May). Universal Stanford dependencies: A cross-linguistic typology. In *LREC* (Vol. 14, pp. 4585-4592).

Dorr, B., Habash, N., & Traum, D. (1998, October). A thematic hierarchy for efficient generation from lexical-conceptual structure. In *Conference of the Association for Machine Translation in the Americas* (pp. 333-343). Sprin

Fillmore, C. J. (1968). The case for case. In E. Bach & R. T. Harms (Eds.), Universals in linguistic theory (pp. 1–88). New York: Holt, Rinehart and Winston.

Firth, J. R. (1957). A synopsis of linguistic theory, 1930-1955. Studies in linguistic analysis.

Gladkova, A., Drozd, A., & Matsuoka, S. (2016). Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn’t. Proceedings of the NAACL Student Research Workshop (pp. 8-15).

Grave, E., Bojanowski, P., Gupta, P., Joulin, A., & Mikolov, T. (2018). Learning word vectors for 157 languages. ArXiv preprint arXiv:1802.06

Gruber, J. (1965), Studies in Lexical Relations, PhD thesis, Indiana University Linguistics Club.

Güngör, O., & Yıldız, E. (2017, May). Linguistic features in Turkish word representations. In *2017 25th Signal Processing and Communications Applications Conference (SIU)* (pp. 1-4). IEEE.

Jackendoff, R. S. (1990). Semantic structures. Cambridge, MA: MIT Press.

Kodinariya, T. M., & Makwana, P. R. (2013). Review on determining number of Cluster in K-Means Clustering. *International Journal*, *1*(6), 90-95.

Luong, M. T., Socher, R., & Manning, C. D. (2013, August). Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning* (pp. 104-113).

Madhulatha, T. S. (2012). An overview on clustering methods. *arXiv preprint arXiv:1205.1117*.

Majumder, P., Mitra, M., & Chaudhuri, B. B. (2002, November). N-gram: a language independent approach to IR and NLP. In *International conference on universal knowledge and language*.

McDonald, R., Nivre, J., Quirmbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., ... & Lee, J. (2013, August). Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 92-97).

Musil, T., Vidra, J., & Mareček, D. (2019). Derivational morphological relations in word embeddings. arXiv preprint arXiv:1906.02510.

Oflazer, K. (2003). Dependency parsing with an extended finite-state approach. *Computational Linguistics*, *29*(4), 515-544.

Oflazer, K., Göçmen, E., & Bozşahin, C. (1994). An outline of Turkish morphology. *Report to NATO Science Division SfS III (TU-LANGUAGE), Brussels*.

O'Gorman, T., Pradhan, S., Palmer, M., Bonn, J., Conger, K., & Gung, J. (2018, May). The new Propbank: Aligning Propbank with AMR through POS unification. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Özbek, A. (2012). Türkçe eşanlamlı kelimeler sözlüğü (Turkish thesaurus), Github repository, https://github.com/maidis/mythes-tr.

PALMER, Martha; GILDEA, Daniel; KINGSBURY, Paul. The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 2005, 31.1: 7

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, *12*, 2825-2830.

Petrov, S., Das, D., & McDonald, R. (2011). A universal part-of-speech tagset. *arXiv preprint arXiv:1104.2086*.

Shalizi, C. (2009). Distances between clustering, hierarchical clustering. *Lectures notes*.

Şahin, G. G., & Adalı, E. (2018). Annotation of semantic roles for the Turkish proposition bank. *Language Resources and Evaluation*, *52*(3), 673-706

Şahin, G. G., & Steedman, M. (2018). Character-Level Models versus Morphology in Semantic Role Labeling. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 386–396).

Şenel, L. K., Utlu, I., Yücesoy, V., Koc, A., & Cukur, T. (2018). Semantic structure and interpretability of word embeddings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *26*(10), 1769-1779.

Toutanova, K., Klein, D., Manning, C. D., & Singer, Y. (2003, May). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 conference of the North American chapter of the association for computational linguistics on human language technology-volume 1* (pp. 173-180). Association for Computational Linguistics.

Türk, U., Atmaca, F., Özateş, Ş. B., Berk, G., Bedir, S. T., Köksal, A., ... & Özgür, A. (2020). Resources for Turkish Dependency Parsing: Introducing the BOUN Treebank and the BoAT Annotation Tool. *arXiv preprint arXiv:2002.10416*.

Wilson, Benjamin J., and Adriaan MJ Schakel. "Controlled experiments for word embeddings." *arXiv preprint arXiv:1*

Yüret, D., & Türe, F. (2006). Learning morphological disambiguation rules for Turkish. Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (pp. 328-334). Blakemore, S. J., & Robbins, T. W. (2012). Decision making in the adolescent brain. Nature Neuroscience, 15, 1184-1191.

Zeman, D. (2008, May). Reusable Tagset Conversion Using Tagset Drivers. In *LREC* (Vol. 2008, pp. 28-30).

Zeman, D., Popel, M., Straka, M., Hajic, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., Tyers, F., Badmaeva, E., Gokirmak, M., Nedoluzhko, A., Cinkova, S., Hajic jr., J., Hlavacova, J., Kettnerová, V., Uresova, Z., . . . Li, J. (2017). CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Published. https://doi.org/10.18653/v1/k