TÜRKÇE KELİME VEKTÖRLERİNDE GÖRÜLEN ANLAMSAL VE BİÇİMSEL YAKINLAŞMALAR

Mehmet Turgut Sübay
17 14 02 122

YÜKSEK LİSANS TEZİ
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Tezli Yüksek Lisans
Danışman: Dr. Mehmet Ali Aksoy Tüysüz

İstanbul
T.C. Maltepe Üniversitesi
Fen Bilimleri Enstitüsü
Temmuz, 2019

JÜRİ VE ENSTİTÜ ONAYI

JÜRİ VE ENSTİTÜ ONAYI

MEHMET TURGUT SÜBAY'ın "Türkçe Kelime Vektörlerinde Görülen Anlamsal ve Biçimsel Yakınlaşmalar" başlıklı tezi 28.08.2019 tarihinde aşağıdaki jüri tarafından değerlendirilerek "Maltepe Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliği" nin ilgili maddeleri uyarınca Bilgisayar Mühendisliği Anabilim Dalı Yüksek Lisans/~~Doktora~~ tezi oy birliğiyle/~~oy çokluğuyla~~, başarılı/~~başarısız~~ olarak kabul edilmiştir.

	Unvanı. Adı ve Soyadı	İmza
Üye (Tez Danışmanı)	Dr. Öğr. Üyesi Mehmet Ali Aksoy TÜYSÜZ	.....
Üye	Prof. Dr. Mesut RAZBONYALI	.....
Üye	Doç. Dr. Turgay Tugay BİLGİN	.....

Prof. Dr. İlter BÜYÜKDIĞAN
Enstitü Müdürü

ŞEKİL ONAY SAYFASI

	ŞEKİL ONAY SAYFASI	Doküman No	FR-105
		İlk Yayın Tarihi	20.12.2017
		Revizyon Tarihi	10.12.2018
		Revizyon No	01
		Sayfa	1/2

ŞEKİL ONAY SAYFASI

18/09/2019

T.C. MALTEPE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜNE,

Aşağıda bilgileri bulunan lisansüstü öğrencinin tezi şekil yönünden tarafımca incelenmiş ve Enstitüye teslim edilmesi uygun bulunmuştur.

Anabilim Dalı Başkanı
Adı-Soyadı
İmza

ÖĞRENCİ BİLGİLERİ	
ADI SOYADI	MEHMET TURGUT SÜBAY
ÖĞRENCİ NUMARASI	171402122
ANABİLİM DALI	BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI BİLGİSAYAR MÜHENDİSLİĞİ TEZLİ YÜKSEK LİSANS
PROGRAMI	(X) YÜKSEK LİSANS ( ) DOKTORA ( ) SANATTA YETERLİK
DANIŞMANI	Dr. MEHMET ALİ AKSOY TÜYSÜZ
TEZ BAŞLIĞI	TÜRKÇE KELİME VEKTÖRLERİNDE GÖRÜLEN ANLAMSAL VE BİÇİMSEL YAKINLAŞMALAR
SAVUNMA TARİHİ	28/08/2019
e-posta	m.turguts@hotmail.com

İç Kapak	☑ Var ☐ Yok
Jüri Onay Sayfası	☑ Var ☐ Yok
Etik İlke ve Kurallara Uyum Beyanı	☑ Var ☐ Yok
İntihal Raporu	☑ Var ☐ Yok
Teşekkür Sayfası	☑ Var ☐ Yok
Öz (Başlık-Öz-Anahtar Sözcükler)	☑ Var ☐ Yok

Hazırlayan İlgili Birim	Kalite Koordinatörü Dr. Öğr. Üyesi Şafak GÜNDÜZ	Kurumsal Yetkili Prof. Dr. Belma AKŞİT

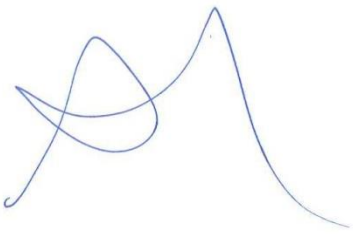
(Doküman No: FR-105; Yayın Tarihi 20.12.2017; Revizyon Tarihi: ; Revizyon No:00)

	ŞEKİL ONAY SAYFASI	Doküman No	FR-105
		İlk Yayın Tarihi	20.12.2017
		Revizyon Tarihi	10.12.2018
		Revizyon No	01
		Sayfa	2/2

Abstract (Title-Abstract-Key Words)	☑ Var ☐ Yok
İçindekiler	☑ Var ☐ Yok
Çizelgeler Listesi	☐ Var ☑ Yok
Şekiller Listesi (varsa)	☐ Şekil yok ☑ Uygundur ☐ Uygun Değildir
Kısaltmalar Listesi	☑ Var ☐ Yok
Tablolar Listesi (varsa)	☐ Tablo yok ☑ Uygundur ☐ Uygun Değildir
Ekler Listesi (varsa)	☑ Ek yok ☐ Uygundur ☐ Uygun Değildir
Özgeçmiş	☑ Var ☐ Yok
Sayfa Genişliği	☑ Uygundur ☐ Uygun Değildir
Yazı Tipi	☑ Uygundur ☐ Uygun Değildir
Referans Kullanımı	☑ Uygundur ☐ Uygun Değildir
Kaynakça Yazımı	☑ Uygundur ☐ Uygun Değildir
Ekler (varsa)	☐ Ek yok ☑ Uygundur ☐ Uygun Değildir

Hazar AKGÜL

İmza



Hazırlayan İlgili Birim	Kalite Koordinatörü Dr. Öğr. Üyesi Şafak GÜNDÜZ	Kurumsal Yetkili Prof. Dr. Belma AKŞİT

(Doküman No: FR-105; Yayın Tarihi 20.12.2017; Revizyon Tarihi: ; Revizyon No:00)

maltepe üniversitesi	ETİK İLKE VE KURALLARA UYUM BEYANI	Doküman No	FR-178
		İlk Yayın Tarihi	01.03.2018
		Revizyon Tarihi	
		Revizyon No	00
		Sayfa	iv/xxxiii

Revizyon Takip Tablosu

REVİZYON NO	TARİH	AÇIKLAMA
00	01.03.2018	İlk yayın.

ETİK İLKE VE KURALLARA UYUM BEYANI

28/08/2019

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarından bilimsel etik ilke ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilmeyen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; çalışmanın Maltepe Üniversitesinde kullanılan "bilimsel intihal tespit programı" ile tarandığını ve öngörülen standartları karşıladığını beyan ederim.

Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçlara razı olduğumu bildiririm.

(Islak İmza)
Mehmet Turgut Sübay

Hazırlayan	Kalite Koordinatörü	Kurumsal Yetkili
İlgili Birim	Dr. Öğr. Üyesi Şafak GÜNDÜZ	Prof. Dr. Belma AKŞİT

(Doküman No: FR-178; Yayın Tarihi: 01.03.2018; Revizyon Tarihi: ; Revizyon No:00)

İNTİHAL RAPORU

Türkçe Kelime Vektörlerinde Görülen Anlamsal Ve Biçimsel Yakınlaşmalar

ORIJINALLIK RAPORU

%8	%7	%2	%7
BENZERLIK ENDEKSI	İNTERNET KAYNAKLARI	YAYINLAR	ÖĞRENCI ÖDEVLERI

BIRINCIL KAYNAKLAR

1	Submitted to The Scientific & Technological Research Council of Turkey (TUBITAK) Öğrenci Ödevi	%1
2	www.eba.gov.tr İnternet Kaynağı	%1
3	docplayer.net İnternet Kaynağı	<%1
4	Submitted to Eskisehir Osmangazi University Öğrenci Ödevi	<%1
5	dolusozluk.com İnternet Kaynağı	<%1
6	kredidunyasi.blogspot.com İnternet Kaynağı	<%1
7	archive.org İnternet Kaynağı	<%1
8	Submitted to Nigde University Öğrenci Ödevi	<%1

TEŞEKKÜR

Bu tezi hazırlamamda bana yardımlarını esirgemeyen sayın Dr. Mehmet Ali Aksoy Tüysüz hocama ve bilgisayar, yazılım teknolojileri ile ilgili ufkumu genişletmemede yardımcı oldukları için master dönemindeki bütün hocalarıma teşekkür ederim.

Bütün master dönemimde derslere zamanında gidebilmem için işyerinden izin almamda yardımcı olan müdürlerime ayrıca teşekkür ederim.

Yıllarca yazılım sektöründe çalıştıktan sonra bilgisayar mühendisliği alanında master yapmamda bana destek olan abim Prof. Dr. R. Kemal Sübay'a teşekkür ederim.

Mehmet Turgut Sübay

Temmuz 2019

ÖZ

TÜRKÇE KELİME VEKTÖRLERİNDE GÖRÜLEN ANLAMSAL VE BİÇİMSEL YAKINLAŞMALAR

Mehmet Turgut Sübay
Yüksek Lisans Tezi
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Yüksek Lisans Programı
Danışman: Dr. Mehmet Ali Aksoy Tüysüz
Maltepe Üniversitesi Fen Bilimleri Enstitüsü, 2019

Bilgisayar bilimi ve teknolojisinin gelişmeye başladığı ilk yıllardan itibaren, insan ile bilgisayar arasındaki etkileşimi arttırmak, tercüme yapmak, büyük miktarlardaki doğal dil verilerini işlemek önemli araştırma alanları olmuştur. Bilgisayar bilimleri, yapay zeka ve bilgi teknolojilerinin kesişim noktasında bulunan doğal dil işleme teknikleri bu alanda çalışmakta ve araştırmacılara sürekli yeni ufuklar sunmaktadır.

Doğal dil işlemede önemli araştırma konularından biri kelimelerin reel sayılardan oluşan vektörlere çevrilmesi teknikleridir. Bu tekniklerle elde edilen vektörlerin, kelimeyi doğru temsil etmesi istenmekte, diğer bir deyişle kaliteli vektörler elde etmek hedeflenmektedir. Vektör kalitesinin artması, kelimeler arasında bulunan çok yönlü ilişkileri yansıtabilme kabiliyetlerini arttırmaktadır. Kelimeler arası ilişkilerinden doğan mantıksal sonuçlar, vektörler üzerinde yapılan basit aritmetik işlemler ile bulabilmektedir [8]. Tomas Mikolov ve ekibi tarafından geliştirilmiş olan Word2vec teknikleri bu alanda başarılı kabul edilmiştir.

Kelimelerden elde edilen vektörlerin kümelenmeleri ile ilgili çalışmaların çoğu İngilizce üzerine yapılmıştır. Türkçe üzerine yapılan çalışmalar halen başlangıç aşamasındadır. Belirtilen noktadan hareketle, sondan eklemeli ve ek açısından zengin bir dil olan Türkçe için hazırlanan derlem üzerinde Word2vec teknikleri bu çalışmada kullanılmıştır. Word2vec teknikleri ile elde edilen kelime vektörlerinin, ait oldukları kelimelerin anlam ilişkilerinin yanında, biçimsel özellikleri açısından da kümelenmeleri incelenmiştir.

Anahtar Sözcükler: Word2vec, Doğal dil işleme, Kosinüs benzerliği, Kelimelerin biçimsel ve anlamsal ilişkisi, Kelime vektörü.

ABSTRACT

THE SEMANTIC AND MORPHOLOGIC SIMILARITY IN TURKISH WORD EMBEDDINGS

Mehmet Turgut Sübay
Master Thesis
Department of Computer Engineering
Computer Engineering Programme
Advisor: Dr. Mehmet Ali Aksoy Tüysüz
Maltepe University Graduate School of Science and Engineering School, 2019

Natural language processing (NLP) is relevant research subject in the fields of artificial intelligence (AI), Information engineering and Computer science. It will also be relevant in future.

One of the most important topics in natural language processing is the word translation into vectors of real numbers (word embeddings). How the quality of word vectors improves using these techniques, syntactic and semantic clustering quality are increased [8]. Word2vec is one of the latest techniques developed by Tomas Mikolov et al, to study high quality vectors.

The majority of studies on clustering of the word vectors were made in English. The studies on Turkish language are still investigating. We base our research on the idea that by means of Word2vec techniques on Turkish corpus we get Turkish representations of word vectors. We searched semantic and morphological word vectors relations in Turkish.

Keywords: Word2vec, NLP, Cosine similarity, Morphological (linguistics) relations, Semantic relations, Word vectors, word embeddings.

İÇİNDEKİLER

TABLOLAR LİSTESİ

ŞEKİLLER LİSTESİ

KISALTMALAR

NLP : Doğal Dil İşleme (Natural language processing)

YP : Yapay Zeka (Artificial Intelligence)

NNLM : Sinir Ağı Dil Modeli (Neural Network Language Model)

RNN : Tekrarlayan Sinir Ağı (Recurrent Neural Network)

RNNLM :Tekrarlayan Sinir Ağı Dil Modeli (Recurrent Neural Network Language Model)

CPU : Merkezi İşlem Birimi (Central Processing Unit)

CBOW : Ardışık Kelimeler Topluluğu (Continuous Bag-of-Words) algoritması

Skip-gram : Komşu Kelimeleri Tahmin Algoritması

TDK : Türk Dil Kurumu

ALPAC : Otomatik Dil İşleme Danışma Kurulu (Automatic Language Processing Advisory Committee)

One-hot : Bir-Sıfır vektörleri

ÖZGEÇMİŞ

Mehmet Turgut Sübay

Bilgisayar Mühendisliği Anabilim Dalı

Eğitim

Y.Ls.	2017	Maltepe Üniversitesi, Fen Bilimler Enstitüsü Bilgisayar Mühendisliği (Tezli)
Ls.	1992	İstanbul Teknik Üniversitesi, Kimya Metalurji Fakültesi Metalurji Mühendisliği Bölümü.
Lise	1987	Kadıköy Kenan Evren Lisesi

İş/İstihdam

Yıl	*Görev*
2014 -	Piramit Danışmanlık A.Ş. Yazılım Uzamanı
2010- 14	WhiteCad Technologies Yazılım Uzmanı

Kişisel Bilgiler

Doğum yeri ve yılı : İstanbul, 1968 Cinsiyet: Erkek
Yabancı diller : İngilizce (çok iyi);
e-posta : turgutsubay@gmail.com

BÖLÜM 1. GİRİŞ

Bilgisayar bilimleri ve teknolojisinin gelişmeye başladığı ilk yıllardan itibaren her teknolojik adım, daha büyük miktarlarda verinin daha küçük hacimlerde ve daha ucuza depolanma olanağını insanlığa sunmuştur. Büyük boyutlardaki verinin depolanması, veri üzerinde hızlı analizlerin yapılması, verinin paylaşımı bilgisayar bilimlerini önemli faaliyet alanı haline getirmiştir. Günümüzde bilgisayar teknolojileri toplumların her kesimine hitap edebilen ürünler sunmaktadır.

Bilgisayarlar vasıtası ile depolanan verilerin büyük boyutlara ulaşması, beraberinde verinin hızlı bulunması problemini getirmektedir. Bu problemin çözüm teknikleri önemli bir endüstri alanı oluşturmakta ve aralıksız olarak geliştirilmeye devam edilmektedir. Veri üzerinde içerik, anlam, duygu, ticari ve benzeri ihtiyacı duyulan analizlerin insan eli ile yapılması, büyüyen veri ile beraber yavaş kalmaktadır. Büyük verinin insan denetimi ile işlenmesi yüksek maliyet problemini de beraberinde getirmektedir. Endüstriyel ihtiyaçlar doğrultusunda bu problemlerin çözümü, insan yardımı olmadan otomatik veri analizleri yapabilen yazılım teknolojileri olarak kendini göstermektedir. Günümüzde otomatik analizler artan endüstriyel ihtiyaçları karşılamak amacıyla sürekli olarak geliştirilmektedir. Otomatik analizler sayesinde, bilgi erişimi, fotoğraflardan kişilerin veya cisimlerin tanınması, e-postaların reklam içeriklerinin ayırt edilmesi, yazışmalardaki duygu analizlerinin yapılması, diller arası çevrim ve birçok benzer ihtiyaç karşılanabilmektedir.

İnsan denetimi olmadan öğrenebilen ve otomatik veri analizleri yapabilen bilgisayar teknolojileri, yapay zeka teknolojilerinin konusunu oluşturmaktadır. İngiliz matematikçi Alan Turing, bilim dünyasına sunduğu "Bilgisayar insan gibi düşünebilir mi?" teklifi yapay zeka fikrini tartışılmaya açmıştır[3]. İnsan gibi öğrenebilen, insanlarla iletişim kurabilen yapay zeka teknolojileri araştırmacılar için önemli araştırma alanları açmaktadır.

Bilgisayar biliminin gelişmeye başlaması ile beraber, insanların bilgisayarları daha kolay kullanabilmeleri ile ilgili istekler süregelmiştir. Birçok işlevin insan denetimine gerek kalmadan, bilgisayarlar tarafından hızlı ve doğru bir şekilde yapılması

talepleri de gün geçtikçe artmaktadır. Bu talepleri karşılamak ve insan ile bilgisayar arasındaki iletişimi arttırmak amacıyla, doğal dil işleme teknolojileri de sürekli büyüyen araştırma alanı haline gelmiştir.

1.1 Doğal dil işleme (Natural Language Processing)

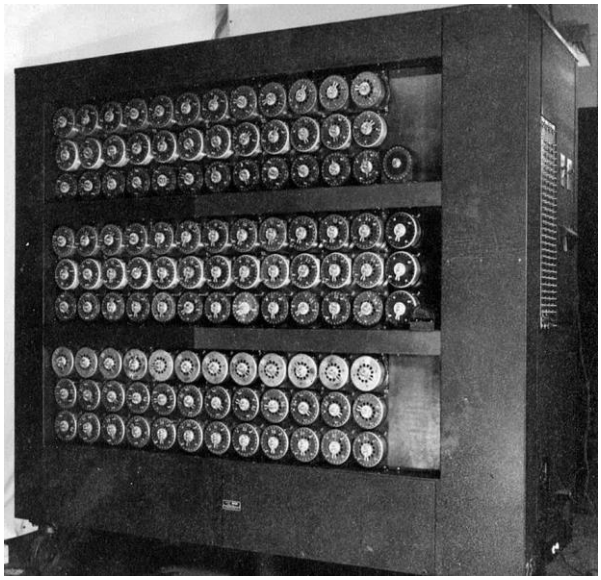
Bilgisayar bilimleri, yapay zeka ve bilgi teknolojilerinin kesişim noktasında bulunan doğal dil işleme (Natural Language Processing - NLP) insan ile bilgisayar arası iletişimi artırmak ve doğal dil analizleri yapmak amacını taşıyan iki temel konu üzerine odaklanmıştır. Bilgisayar teknolojilerinin gelişmesine paralel olarak, metinlerin otomatik sınıflandırılması, büyük miktarlardaki dil verilerinin işlenerek sonuçlar elde edilmesi vb. taleplerin karşılanması için doğal dil işleme teknikleri sürekli olarak geliştirilmektedir. Günlük hayatın her alanında kullanımı giderek artan doğal dil işleme teknolojileri son zamanlarda neredeyse bütün akıllı telefonlarda kullanılmaktadır. İnsan konuşmalarını analiz ederek kullanıcıya yardımcı olmaya çalışan asistan programlar, arama motorlarındaki bilgi erişim teknikleri bu teknolojinin ürünleridir. Yazılan makalelerin veya metinlerin farklı dillere otomatik tercümesi (makine tercümesi) halen üzerinde araştırılan önemli bir doğal dil işleme alanıdır. Doğal dil işleme faaliyet alanına giren bazı araştırma alanları aşağıdaki başlıklarda sıralanmıştır.

- Konuşma tanıma
- Diller arası tercüme
- Söyleşi/Sohbet uygulamaları(Chatbot)
- Soru cevaplama uygulamaları
- Bilgi erişimi
- Duygu analizi
- Anlam ve söz dizimi analizi
- Özet çıkartma
- Reklam içerikli e-posta süzme
- Doküman sınıflama

1.2 Doğal dil işleme gelişim süreci

Bilgisayar teknolojilerinin gelişmeye başladığı ilk yıllarda, doğal dil işleme kural tabanlı teknikler üzerinden geliştirilmeye başlanmıştır. Doğal dilin karmaşık yapısı, kuralların belirlenmesinde birçok soruna ve karmaşaya neden olduğu görülmüştür. Geliştirilen uygulamanın, doğal dilin detaylarına adapte edilmeye çalışılması ile beraber, kurallar arasındaki karmaşanın artması, içinden çıkılması zor problemlere neden olduğundan, istenen sonucun elde edilmesi mümkün olmamıştır. İstenen sonucun bir türlü elde edilememesi nedeni ile kural tabanlı tekniklerin doğal dil işlemede yetersiz kaldığı sonucuna varılmıştır. Kural tabanlı tekniklerden vazgeçilmesinden sonra, istatistik tabanlı tekniklerin kullanılmasında başarı sağlanmıştır. İstatistik tabanlı tekniklerin büyüyen veri ile beraber yavaş kalması üzerine, günümüzde makine öğrenmesi yapay zeka tekniklerinin kullanımı giderek arttığı görülmektedir.

Doğal dil işleme 1950 yılında Alan Turing'in "Computing machinery and intelligence" adlı yayını ile ("Turing test" olarak anılır) başladığı kabul görmüştür[1][2]. Kendisi teorik bilgisayar bilimi ve yapay zekanın babası olarak anılmaktadır. Turing ikinci dünya savaşı sırasında bir matematikçi ve kripto analizci olarak İngiliz kod çözümleme merkezinde görevlendirilmiştir[3]. Savaşı yıllarında Almanlar, karşı tarafın eline geçmesini istemedikleri bilgileri şifrelemek amacıyla, kripto mesajlar üretebilen Enigma adında bir makine kullanmışlardır. Enigma makinesi ile üretilen kripto mesajları çözümlemek için Turing, Victory isimli elektro-mekanik bir makine tasarlamıştır. Victory makinesi ile on, on beş dakika gibi bir süre içinde Enigma'nın şifreli mesajlarının çözümlenmesi başarılmıştır. İkinci dünya savaşı üzerine yapılan analizlerde, Victory adlı makine ile çözümlenen şifreli mesajlar sayesinde, Avrupa'da savaşın iki yıl daha erken bittiği tespit edilmiştir. Savaşın erken bitmesi, on dört milyon insan hayatının kurtulmasına vesile olduğu tahmin edilmektedir[4].



Şekil 1 Alan Turing'in tasarladığı Victory makinesi (Bu makine sayesinde ikinci dünya savaşının iki yıl erken bittiği ve on dört milyon insan hayatının kurtulduğu tahmin edilmiştir)

İstatistiksel doğal dil işleme ile ilgili önemli isimlerden biri de bilgi kuramının babası olarak anılan Claude Shannon'dur. 1948 ve 1950 yıllarında Shannon "A Mathematical Theory of Communication" ve "Prediction and Entropy of Printed English" [5][6] adlı çalışmalarını yayınlamıştır. Shannon N-gram modelini ortaya koyarak NLP'de istatik tabanlı tekniklerin gelişimine önemli katkı sağlamıştır. N-gram modelinde, bir harf veya kelime dizisinden sonra gelen harf veya kelimenin ne olabileceği sorusunun cevabı, olasılık olarak verilmeye çalışılmaktadır. Doğal dil işlemede N-gram modeli istatistik tabanlı bir teknik olarak makine tercümesi, konuşma tanıma gibi alanlarda başarıyla uygulanmıştır[22][23].

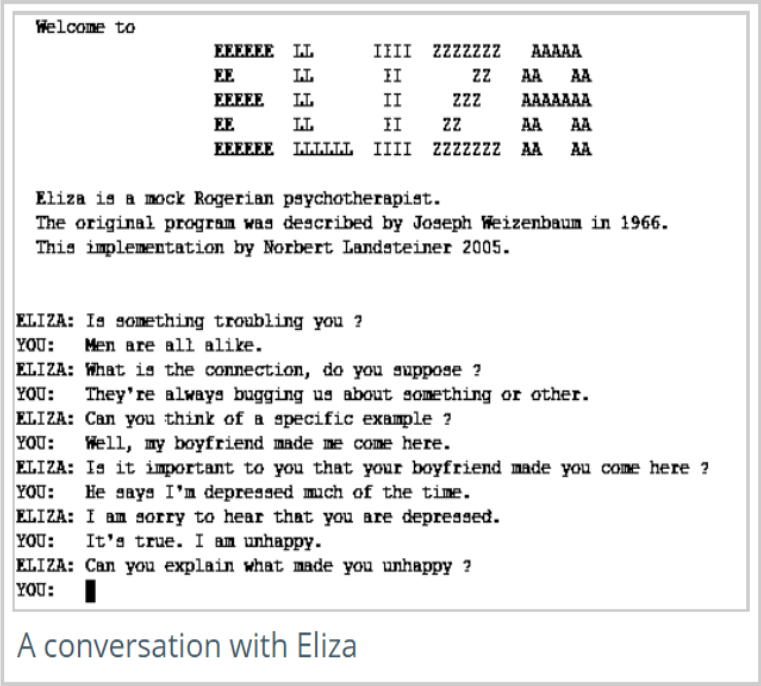
1954 yılında, Georgetown Üniversitesi ve IBM tarafından, IBM 701 mainframe bilgisayar sisteminde az sayıda dil bilgisi kuralı ve kelime sayısı ile sınırlı Rusça - İngilizce tercüme yapabilen, kural tabanlı bir algoritma geliştirilmiştir[18]. Geliştiriciler üç, beş yıl içinde bilgisayarlı otomatik çevirinin (makine tercümesi) tam olarak yapılabileceğini iddia etmişlerdir. Aradan geçen on yıllık süreçte, üzerinde çalışılmasına rağmen bu konuda fazla bir ilerleme sağlanamamıştır. 1964 yılında, bilgisayarlı dilbilim ve makine tercümesi alanında Amerika Birleşik Devletleri hükümeti tarafından yedi bilim insanı görevlendirilerek ALPAC (Automatic Language Processing Advisory Committee) komitesi kurulmuştur. ALPAC komitesi 1966 yılında, makine tercümesi hakkında olumsuz bulgu ve düşüncelerini içeren bir rapor hazırlamıştır[19]. Amerika Birleşik Devletleri hükümeti ALPAC komitesinin makine tercümesi için hazırladığı

olumsuz raporundan sonra makine tercümesi ile ilgili çalışmalara fonlarını azaltmıştır. Fonların azaltılması, makine tercümesi alanında istatistik tabanlı tekniklerin başarı sağlamasına kadar olan süreçte bir gelişme yapılamamasına neden olmuştur. 1990'ların başında istatistik tabanlı makine tercümesi tekniklerinin başarısı sayesinde, doğal dil işlemede istatistik tabanlı tekniklerin geliştirilmesi önemli bir araştırma alanı haline gelmiştir.

Doğal dil işleme alanında önemli yazılımlardan biri 1964-1966 yıllarında Massachusetts Institute of Technology yapay zeka laboratuvarlarında Joseph Weizenbaum tarafında geliştirilen "ELIZA" yazılımıdır[20]. Bu çalışmada, konuşma dili olarak İngilizce kullanılmıştır. Programın amacı, bir insan ile bilgisayar arasında sohbet yapılması olarak belirlenmiştir. Eliza ilk söyleşi/sohbet (Chatterbots veya Chatbot) türü yazılım olarak bilinmektedir. Bu ilk söyleşi türü yazılımda, kullanıcı Turing teste tabi tutulmuştur. Teste tabi tutulan birçok kişi, bir insanla söyleşi yaptıkları düşüncesine kapılarak, soruların veya cevapların bir bilgisayar tarafından üretildiğini anlayamamışlarıdır. Sonuç olarak, program başarılı olarak kabul edilmiştir.

Eliza yazılımının çalışma prensibi anahtar kelimeler üzerine kurulmuştur. Teste tabi tutulan kişilerin kurduğu cümleler girdi olarak Eliza'ya verilmektedir. Anahtar kelimelerin yazılımdaki kuralları tetiklemesi sayesinde, girdi cümlesinin analizi yapılarak kullanıcıya yeni soru veya cevap üretmektedir. Bir örnek vermek gerekirse, kullanıcı "anne" kelimesini bir cümlede kullandığı zaman, program "bana biraz ailenden bahset" diye yanıtlamakta ve söyleşi bu şekilde anahtar kelimelerin kullanılması ile devam etmektedir[24].

Söyleşi türü yazılımlar günümüzde sanal müşteri destek birimi olarak karşımıza çıkmaktadır. Birçok firma söyleşi türü yazılımları düşük maliyeti ve yılın her günü, yirmi dört saat aralıksız hizmet verebilmesi nedeniyle tercih etmektedirler. Sanal müşteri destek programları arasında çok kullanılanlara örnek olarak Apple Siri, Microsoft Cortana, Google Assistant, Amazon Alex, Facebook Messenger gösterilebilir.



Şekil 2 Eliza ilk söyleşi programı kullanıcı ara yüzü

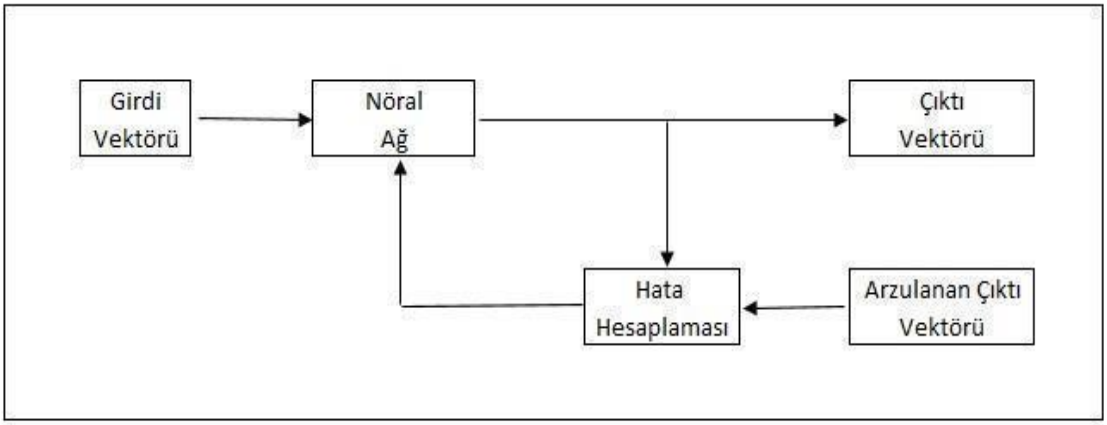
ALPAC komitesi raporundan sonra 1990'lara kadar yapay zeka ve doğal dil işlemede ağırlıklı olarak veri varlıklarının (entities) birbirleri arasındaki ilişkileri, paylaşımları, tekrarlı kullanıma uygun kütüphanelerin ağ üzerinden çağırımına uygun dizaynlar üzerine çalışılmıştır. Bu dönemde öne çıkan çalışmalardan biri Thomas R. Gruber'in "Toward Principles for the Design of Ontologies Used for Knowledge Sharing"[21] adlı yayınıdır. Gruber bu çalışmada veri varlıklarının nasıl dizayn edilebileceği, nasıl paylaşılabileceği ile ilgili kriterleri ortaya koymaktadır.

1990'larda IBM araştırma merkezinde Peter F. Brown ve ekibi, istatistik tabanlı yeni bir makine tercüme tekniği geliştirmişlerdir[7][16][33]. Geliştirme sürecinde Kanada Parlamentosunda yapılan tartışmaların İngilizce ve Fransızca dillerinde tutulan kayıtlarını (Hansard derlemleri) kullanmışlardır. Yeni geliştirilen istatistik tabanlı teknikte N-gram dil modeli kullanılmıştır. Maksimum olabilirlik tahmini yapılmasında beklenti maksimizasyonu (Expectation Maximization) algoritması kullanılarak başarılı sonuçlar elde edilmiştir. Doğal dil işlemede istatistik tabanlı yöntemlerin başarılı olabileceğinin anlaşılması ile istatistik tabanlı teknikler üzerinde çalışmalar artmıştır. İstatistik tabanlı çalışmalarda N-gram dil modeli, bir kelimeden sonra gelebilecek

kelimenin tahmin edilmesi problemi üzerine çalışmaktadır. Olasılık tahmini için maksimum olabilirlik tahmini (maximum likelihood estimation) yöntemi kullanılmaktadır[32].

1986 yılında David E. Rumelhart, "Learning representations by back-propagating errors"[13] adlı çalışmasında hatanın geri yayılımını yeni bir öğrenme tekniği olarak yapay zeka dünyasına tanıtmıştır. Yeni öğrenme tekniğinde yapılmaya çalışılan, eğitim aşaması sırasında doğru olduğu bilinen vektör ile doğruluk tahmini yapılan vektörlerin karşılaştırılmasıdır. Karşılaştırmada iki vektör arasındaki farktan yararlanılarak hata değeri elde edilir. Elde edilen hata değeri sayesinde vektörlerin boyutsal ağırlıklarında düzeltmelerin yapılması yeni tekniğin dayandığı temel prensiptir. İşlemde girdi vektörü, doğrusal olmayan gizli kademeler ile çıktı vektörü mevcuttur. Hatanın geriye yayılması (the backward propagation of errors) yapay sinir ağlarının gelişimine önemli katkı sunmuştur.

Hatanın geri yayılması tekniğinde, girdi verileri vektör olarak ağa verilir ve bir çıktı vektörü üretilir. Ağın ürettiği bu çıktı ile sonucu bilinen eğitim verisi karşılaştırılarak bir hata/kayıp miktarı (Error signal/Loss function) hesaplanır. Yapay sinir ağında bulunan boyutsal ağırlıklar elde edilen hata miktarına göre güncellenir. Bu güncellemeler, hata miktarının uygun bir seviyeye inmesine kadar devam ettirilir.



Şekil 3 Hatanın geri yayılması öğrenme tekniği

1.3 Doğal dil işlemede tek yönlü istatistik tabanlı tekniklerden geriye doğru hata düzeltici tekniklere (back-propagating errors) geçiş

1990'lı yılların başlarında hatanın geri yayılması tekniği doğal dil işlemede de kullanılmaya başlanmıştır. Doğal dil işlemenin gelişmesine önemli katkılar vermiş isimlerden biri olan Yashua Bengio tekrarlayan Sinir Ağı (RNN) üzerinde çalışmalar yapmıştır[25][27][46]. RNN tekniği, hatanın geri yayılması tekniğini temel almaktadır. İstatistik tabanlı tekniklerin eğitim sürelerinin uzun olması nedeniyle pratik kullanıma uygun olmamaları, araştırmacıları RNN üzerinde daha fazla çalışmaya sevk etmiştir[51]. Hatanın geri doğru yayılımının doğal dil işlemede kullanımının benimsenmesi üzerine, öğrenme teknikleri üzerine tartışmalar, derin öğrenme ve sığ öğrenme tekniklerinin karşılaştırılması ile ilgili konular geniş araştırma alanı haline gelmiştir[26]. Mikolov RNN üzerinde yaptığı çalışmalarda[43], istatistik tabanlı modellerin kelimelerin anlam ilişkilerini iyi yansıtmadığını belirtmiştir. Mikolov geliştirdikleri RNN modeliyle elde ettikleri vektörlerin kelimeleri daha doğru temsil edebildiklerini ifade etmektedir.

Günümüzde doğal dil işlemede hatanın geri doğru düzeltilmesi tekniği güncel bir araştırma alanı olmuştur. Bu öğrenme tekniğinde hedeflenen, kelimenin reel sayılardan oluşan çok boyutlu koordinat sistemindeki vektör karşılıklarının bulunmasıdır. Kelimelerin karşılığı olan vektörlerin dilin yapısına uygun olarak kelimeyi doğru temsil etmesi (kaliteli vektörler) arzu edilmektedir. Vektör kalitesinin artması kelimeler arasında bulunan çok yönlü ilişkileri yansıtabilme kabiliyetlerini arttırmaktadır. Kaliteli vektörlerin toplanması ve çıkarılması ile anlamsal sonuçlar elde edilebilmektedir[8].

1.4 Doğal dil işlemede makine öğrenmesini etkileyen temel faktörler

Makine öğrenmesi ile elde edilen vektörler arasındaki kümelenmeler ve alt gruplaşmalar, kelimelerin söz dizimi, anlam ve biçimsel (yapısal) ilişkileri açısından paralellik göstermektedir. Kelimeler arasındaki bu ilişkiler özellikle arama motorları gibi endüstriyel alanlarda geniş uygulama alanı bulmaktadır. Doğal dil işlemede kelimelerin vektörler ile eşleştirilmesi (kelime vektörlerinin bulunması) teknikleri, kelime gömülümleri (word embeddings) olarak adlandırılır[28]. Kelime gömülümleri ile

elde edilen vektörler, kelimenin söz dizimi ve anlam ilişkilerini insan etkileşimine gerek kalmadan yansıtmaktadırlar[45]. Kelime gömülümleri tekniklerinin geliştirilme nedenlerinden biri de, makine öğrenmesi eğitim süresini kısaltmasıdır. Eğitimdeki sürenin kısaltılması pratikte daha çok vektör boyutu ve daha büyük derlemler ile çalışabilme olanağı sağlamaktadır. Büyük derlem ve daha çok vektör boyutu ile makine öğrenmesi eğitimi yapabilmek, vektörlerin kelimeleri doğru temsil etmelerini etkileyen önemli faktörler arasında gösterilmektedir.

Bu	→	[0,15 0,21 0,89 0,22]
gün	→	[0,32 0,14 0,84 0,45]
hava	→	[0,28 0,75 0,34 0,68]
çok	→	[0,19 0,78 0,54 0,48]
güzel	→	[0,92 0,85 0,27 0,67]

Tablo 1 Kelime vektör eşlemesi

Doğal dil işleme makine öğrenmesi ile elde edilen vektörlerin kelimeyi doğru olarak temsil etmelerini etkileye üç temel etken aşağıdaki sıralanmıştır.

- Eğitimde Kullanılan derlemin büyüklüğü
- Eğitilen vektörlerin boyutu
- Komşu kelimelerin sayısı

1.4.1 Eğitimde kullanılan derlemin büyüklüğü

Kullanılan derlemin büyüklüğü, elde edilen sonuçları etkileyen önemli bir etkendir[51]. Derlem büyüklüğünün artırılması, kelime vektörlerinin ağırlıkları üzerinde yapılan hata düzeltme işlemlerinin fazlalaşmasına neden olmaktadır. Vektör ağırlıkları üzerinde yapılan düzeltme işlemlerinin arttırılması sayesinde, daha doğru vektör değerleri elde edilmesinin mümkün olduğu anlaşılmıştır. Derlemin büyümesinin en önemli dezavantajı ise eğitim süresini uzatmasıdır. Bu sorun yeni tekniklerin geliştirilmesi ile aşılmaya çalışılmaktadır[8][29].

1.4.2 Eğitilen vektörlerin boyutu

Vektörün kelimeyi doğru temsil etmesini etkileyen diğer önemli bir etken, eğitilen vektörlerin boyutlarının büyüklüğüdür. Tablo 1'de verdiğimiz örnekteki vektörlerin dört boyutlu olarak temsil edildiği görülmektedir. Kelime vektörlerinin her boyutu, Tablo 3'de görülen denetimli öğrenme ile ilgili özelliklere benzetilebilir. Denetimsiz öğrenmedeki boyut adedi arttırıldıkça, denetimli öğrenmedeki özelliklere benzer şekilde, kelime vektörlerinin kelimeyi daha doğru temsil ettikleri gözlemlenmektedir. Word2vec tekniklerine göre daha eski olan istatistik tabanlı tekniklerde, eğitim süresini kısa tutabilmek için vektörlerin boyut adetleri 50 ile 100 arasında verilmekteydi. Bengio "A Neural Probabilistic Language Model" adlı yayınında, vektör boyutu büyümesinin eğitim sürelerini doğal olarak artırdığını belirtmiştir[46][47]. Eğitim sürelerinin uzun olması sorunu Word2vec'te daha aza indirgenmiştir. Süre sorununun azaltılması ile beraber vektör boyut adetleri Word2vec için arttırılmıştır. Word2vec'te vektör boyutu 300 ile 1000 arasında olacak şekilde belirlenmesi tavsiye edilmektedir[8].

1.4.3 Komşu kelimelerin sayısı

Vektörlerin kelimeyi doğru temsil etmelerini etkileyen diğer bir önemli etken, eğitim sırasında kullanılan komşu kelimelerin sayısıdır. Komşu kelime sayısının artması, vektörler üzerinde daha çok hata düzeltme hesaplarının gerçekleştirmesine neden olmaktadır. Vektörler üzerinde yapılan hata düzeltmelerinin arttırılması, vektörlerin daha doğru ağırlık değerleri almasına neden olmakta, fakat eğitim süresini artırıcı etki yapmaktadır. Word2vec için 5 ile 10 komşuluk adetleri önerilmektedir.

Derlemin büyüklüğü, eğitilen vektörlerin boyutu ve komşu kelime sayısı, vektör kalitesini etkileyen önemli faktörlerdir. Google araştırmacısı Mikolov ve ekibi "Efficient Estimation of Word Representations in Vector Space"[8] adlı yayınlarında yeni geliştirdikleri Word2vec teknikleri ile Sinir Ağı dil modeli (NNLM) tekniği arasındaki eğitim sürelerinin karşılaştırıldığı çalışma aşağıdaki tabloda verilmiştir.

Teknik	Vektör Boyutu	Eğitimdeki Kelime Adeti	Doğruluk (%)			Eğitim süresi [gün x CPU çekirdeği]
			Anlam	Sözdizim	Toplam	
NNLM	100	6B	34.2	64,5	50,8	14 x 180
CBOW	1000	6B	57.3	68,9	63,7	2 x 140
Skip-gram	1000	6B	66.1	65,1	65,6	2.5 x 125

Tablo 2 Makine öğrenme sürelerinin teknikler arası karşılaştırılması (Word2vec tekniğinin eski tekniklere göre daha çok vektör boyutuna sahip olmasına rağmen daha kısa sürede eğitimi tamamlaması)

Tablo 2'de görülen Sinir Ağı dil modeli, Word2vec'e göre daha eski bir tekniktir. Ardışık kelimeler topluluğu (CBOW) ve komşu kelimeleri tahmin (Skip-gram) ise Word2vec algoritmalarını temsil etmektedirler. Bu tabloda görüldüğü üzere 6 milyar kelime ile oluşturulmuş bir derlemde, NNLM tekniği 180 merkezi işlem birimi (CPU çekirdeği) ile 14 günde eğitim işlemi tamamlanmaktadır. Word2vec'in iki algoritmasından ilki CBOW 140 CPU çekirdeği ile 2 günde ve ikincisi Skip-gram 125 CPU çekirdeği ile 2,5 günde makine öğrenmesini tamamlamaktadırlar.

1.5 Doğal dil işlemede yeni teknik olarak Word2vec

Doğal dil işlemede kullanılan tekniklerde, eğitim sürelerinin uzun olması araştırmacıları süreyi azaltmak ve bu sayede büyük derlemlerin işlenerek daha doğru kelime vektörleri elde etmek konusuna sevk etmiştir. Bu konu üzerinde araştırma yapan Google araştırmacısı Tomas Mikolov ve ekibi 2013 yılında Word2vec adında yeni bir teknik geliştirdiklerini duyurmuşlardır[8]. Word2vec tekniği RNN'ye benzer şekilde hatanın geri yayılması tekniğini temel almaktadır. Tablo 2'de görüldüğü üzere, yeni geliştirilen Word2vec tekniğinde yer alan iki algoritma da NNLM tekniğinden daha iyi sonuçlar vermektedirler. Mikolov Word2vec ile elde edilen vektörlerin, doğal dildeki kelimelerin söz dizimi ve anlambilim ilişkilerine benzer şekilde kümelendiklerini belirtmektedir.

İngilizcede kelimeler arasındaki söz dizimi ilişkilerine "great", "greater" veya "easy", "easiest" kelime çiftleri örnek olarak verilebilir. Bu kelime çiftlerindeki söz dizimi ilişkilerine benzer şekilde, Word2vec ile elde edilen vektörlerin kümelendikleri Mikolov tarafından belirtmektedir.

Word2vec ile elde edilen vektörlerin, ait oldukları İngilizce kelimelerin anlam ilişkilerine göre de kümelendikleri yine Mikolov tarafından örneklerle gösterilmiştir. İngilizce kelime çiftleri olarak "Athens", "Greece" kelime çifti ülke ve başkent anlam ilişkisi içerisindedirler. Benzer şekilde "King", "Queen" kelime çifti soyluluk ifadesi olarak anlam ilişkisi içerisindedirler.

Kelimeler arasındaki anlam ilişkileri, kelime vektörlerinin ait oldukları kelimelerin anlam ilişkilerine göre kümelenmesine neden olmaktadır. Vektörlerin kelimeyi doğru temsil etmeleri, mantıksal sonuçlar elde edilebilmesine olanak sağlamaktadır. Word2vec ile elde edilen vektörlerin toplanması ve çıkarılması ile elde edilen yeni vektörün kosinüs benzerliklerinden anlamsal sonuçlar elde edilebilmektedir. Aritmetik işlemleri ile üretilebilen anlamsal sonuçlara örnek vermek gerekirse, soyluluk ifade eden "King" kelimesinde cinsiyet özelliğinin yer değiştirmesi ile elde edilen sonuç vektörü aşağıda görülmektedir[8].

('King') - ('Man') + ('Woman') sonucu ('Queen')

Bu örneğe benzer diğer bir örnekte ülkeler ve başkentleri arasında anlam ilişkisi olarak aşağıda verilmiştir.

('England') - ('London') + ('Athens) sonucu ('Greece)

1.6 Tezin Amacı / Problem

Türkçe köken akrabalığı olarak Ural – Altay dil grubu içinde yer alır ve biçim (kelime yapısı) olarak da eklemeli diller (agglutinative) grubuna dahildir. Türkçede ön ekler yoktur, ekler kök ve gövde sonuna geldiği için sondan eklemeli bir dildir[9]. Yukarıdaki örnekler İngilizce olarak verilmiştir ve çoğaltmak mümkündür, Word2vec ile eğitimi yapılan derlemler çoğunlukla İngilizcedir ve üzerlerinde oldukça çalışma yapılmıştır.

Tomas Mikolov "Efficient Estimation of Word Representations in Vector Space"[8] adlı yayınında aşağıda orijinal metni verilen kısımda, (serbest çeviri ile) kelimelerin birden fazla şekilde benzerlikleri olabileceği için sadece benzer kelimelerin birbirine yaklaşmasını beklememek gerektiğini, bu durumun (Türkçe gibi) çekimlemeli

dillerde daha önce de görüldüğünü belirtmiştir. Yine aynı kaynakta verilen örnekte, isimlerin birden fazla ek ile sonlanabildiğini ve benzer kelimeleri ararken benzer ekler ile sonlanmış kelimelere de ulaşılabileceğini dile getirmiştir. Buradan hareketle, yapılan bu çalışmada Türkçe'nin de (sondan) eklemeli ve ek açısından zengin bir dil olması göz önüne alınarak, durumun Türkçe için incelenmesi hedeflenmiştir.

"... with the expectation that not only will similar words tend to be close to each other, but that words can have multiple degrees of similarity[20]. This has been observed earlier in the context of inflectional languages - for example, nouns can have multiple word endings, and if we search for similar words in a subspace of the original vector space, it is possible to find words that have similar endings [13, 14]" (Mikolov [8])

Yapılan literatür taramasında, Türkçe derlemlerden Word2vec tekniği kullanılarak elde edilen vektörler üzerinde, kelimelerin anlam ilişkisi ve biçimsel özellikleri ele alınarak yapılmış yeterince çalışma olmadığı görülmüştür. Bu sebeple, tezimiz için hazırlanan Türkçe derlem üzerinden (Word2vec ile) elde edilen vektörler, anlamsal ve biçimsel incelemelere tabi tutulmuştur. Elde edilen sonuçlar ve değerlendirmeler 4. bölümde yer almaktadır.

BÖLÜM 2. WORD2VEC MAKİNE ÖĞRENMESİ

Doğal dil işleme istatistik tabanlı tekniklerde[46] eğitim süresinin uzun olması, endüstriyel uygulamalarda kullanım sorunlarına neden olmaktadır. Eğitim sürelerinin kısaltılması için vektör boyutu, eğitimde kullanılan derlemin büyüklüğü ve komşu kelimelerin sayısı sınırlı tutulmaya çalışılmaktadır. Eğitimden elde edilen vektörlerin, kelimeyi doğru olarak temsil etmesini etkileyen üç temel etkenin sınırlandırılması, kelime vektörlerin doğruluğunu olumsuz yönde etkilemektedir[51]. Bu sorunu aşabilmek amacıyla, Google araştırmacıları Mikolov ve ekibi 2013 yılında denetimsiz öğrenme tekniğine dayalı, Word2vec adını verdikleri yeni bir teknik geliştirmişlerdir. Bu yeni teknik, ardışık kelimeler topluluğu (CBOW) ve komşu kelimeleri tahmin (Skip-gram) adında, benzer işlem akışı ve hesap yöntemine sahip iki algoritmadan oluşmaktadır[8]. Bu yeni teknik derin öğrenmeye göre daha sığ iki katmanlı bir ağ yapısına sahiptir. Derin öğrenme tekniklerine göre sığ, diğer bir deyişle az katmanlı tutulmasının sebebi, hesaplama karışıklıklarını aza indirgemektir. Aza indirgenen hesaplamalar sayesinde, işlem döngüsü hızlanmaktadır. Word2vec ile kısaltılan eğitim süresi sayesinde istatistik tabanlı tekniklere göre büyük derlemler, daha çok vektör boyutları ve daha çok sayıda komşu kelimeler ile eğitim yapılabilme olanağı elde edilmiştir.

Doğal dili hızlı anlama ve yorumlama problemi, doğal dil işleme araştırma konularındandır. İnsanlar doğal dili çok hızlı anlamakta ve yorumlayabilmekteler[34]. Bir bilgisayarın insan kadar hızlı doğal dili anlaması ve yorumlaması günümüz makine öğrenmesinde önemli bir araştırma alanı olarak görülmektedir. Bu alandaki araştırmalar, makine öğrenmesi tekniklerinin geliştirilmesi araştırma alanına doğru kaymıştır. Doğal dil işleme makine öğrenmesi tekniklerinde, denetimli ve denetimsiz öğrenme teknikleri sıklıkla kullanılmaktadır. Word2vec'de makine öğrenmesi tekniği olarak denetimsiz öğrenme tekniği kullanılmıştır. Bu öğrenme tekniğini karşılaştırmalı olarak açıklayabilmek amacıyla, denetimli öğrenme tekniği ile ilgili bilgilendirme "Denetimli öğrenme" konu başlığı ile verilmiştir.

2.1 Denetimli öğrenme (Supervised learning)

Günlük hayata her gün e-posta kutularına birçok e-posta gelmektedir. E-posta kutusuna gelen bu postaların gerçekten posta amaçlı mı? yoksa reklam amaçlı mı? gönderildiği sorusu endüstriyel alanda sorulan sorular arasındadır. Bu soruya denetimli öğrenme; reklam içerikli kelimelerin reklam postalarında sıkça kullanıldığı varsayımından hareketle çözüm bulmaktadır. Gelen posta içerikleri, denetimli öğrenmede girdi verisi olarak eğitim işlemine verilirken, reklam içerikli postalar reklam postası olarak işaretlenir. İşaretli postalar, doğrusal olmayan öğrenme fonksiyonunun eğitimi ile reklam postası olarak sınıflandırılarak, gerçek postalardan ayırt edilebilir hale getirilmektedir[35]. Makine öğrenmesi ile elde edilen fonksiyon sayesinde, gelen postalardan reklam içerikli olanlar ayırt edilebilmektedir.

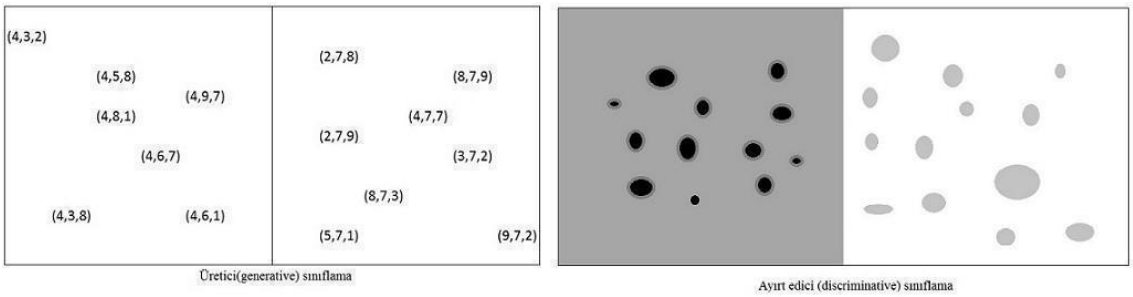
Denetimli öğrenme tekniğinde, vektörlerin her boyuta karşılık gelen özellikleri (features) makine öğrenmesi eğitiminden önce tespit edilir. Belirlenmiş özellikler ile hangi sınıfa ait olduğu bilinen etiketli verilerin, girdi verisi olarak kullanılması sonucu öğrenme fonksiyonunun eğitimi yapılır[40]. Özellik belirleme zor bir iştir ve iyi tasarlanmalıdır. Bir örnek vermek gerekirse, omurgalı hayvanların ayırt edilmek istendiğini varsayalım. Bütün omurgalıların derisi vardır özelliği belirlensin, ikinci özellik olarak, omurgalılar sıcakkanlı hayvanlardır dendiğinde, ikinci özellik birinciye göre çok anlamlı olacak ve birinci özellik gereksiz yere tanımlanmış olacaktır[42]. Çok miktarda özellik tanımlanması alt sınıflamaları artırdığından, sonucun doğruluğunu olumlu yönde etkilemekte, fakat eğitim süresini arttırmaktadır. Eğitim işleminden sonra, elde edilen fonksiyonun doğruluğu oranında, testi yapılacak yeni girdilerin (etiketsiz veriler) hangi sınıfa ait olduğuna, insan denetimi olmadan fonksiyon tarafından karar verebilmektedir.

	kral	kraliçe	kadın	erkek	elma	araba
unvan	0,91	0,92	0,05	0,04	0,001	0,002
cinsiyet	0,15	0,2	0,98	0,95	0,002	0,05
yaş	0,58	0,45	0,2	0,15	0,05	0,12
besin	0,001	0,002	0,02	0,035	0,75	0,021
araç	0,002	0,001	0,002	0,007	0,06	0,82
..	..	..	..	..	..	..
..	..	..	..	..	..	..

Tablo 3 Denetimli öğrenme temsili tablosu

Tablo 3' te doğal dil işlemede kullanılabileceği varsayılan bazı kelime özellikleri örneklendirilmiştir. Belirlenmiş özellikler olarak unvan, cinsiyet, yaş, besin, araç vb. özellikleri görülmektedir. Kelimeler bu özelliklere göre sınıflandırılabilir. Bir örnek vermek gerekirse, "Kral" ve "Kraliçe" kelimelerinin unvan özelliği birbirine yakın ve yüksek değerler alması beklenen bir sonuç olmasına karşın, "Besin" kelimesindeki unvan özelliğinin daha düşük bir değere sahip olması beklenen bir sonuç olmaktadır.

Denetimli öğrenmede, üretici (generative) sınıflama ve ayırt edici (discriminative) sınıflama olmak üzere iki genel öğrenme modeli bulunmaktadır[36]. Üretici sınıflama modelinde ortak özelliklerin olasılık dağılım tahmini p(x, y) yapılarak modellenir. Bir örnek vermek gerekirse, kaşık ve bıçakların bulunduğu bir kümeden rastgele bir numune seçildiğinde sapı olma olasılığı 1 olacaktır. Rastgele seçilen numunenin keskin bir kenarı olma olasılığı ½ olacaktır. Ayırt edici sınıflama modelinde koşullara bağlı olasılık tahmini p(y | x) yapılmaktadır. Bir örnek vermek gerekirse, yemek tariflerinin yapıldığı bir dokümanın bir cümlesinde geçen "Elma" kelimesinden sonra gelen kelimenin "Sirkesi" kelimesi olma olasılığı ile "Otobüsü" kelimesi olma olasılığı nedir? Sorusunun cevabını bulmak için ayırt edici sınıflama modellemesi kullanılabilir.



Şekil 4 Üretici (generative) sınıflama (birinci elemanı dört, ikinci elemanı yedi olanlar) ve ayırt edici (discriminative) sınıflama (gri yüzerde siyah, beyaz yüzerde gri elemanlar)

2.2 Denetimsiz öğrenim (Unsupervised learning)

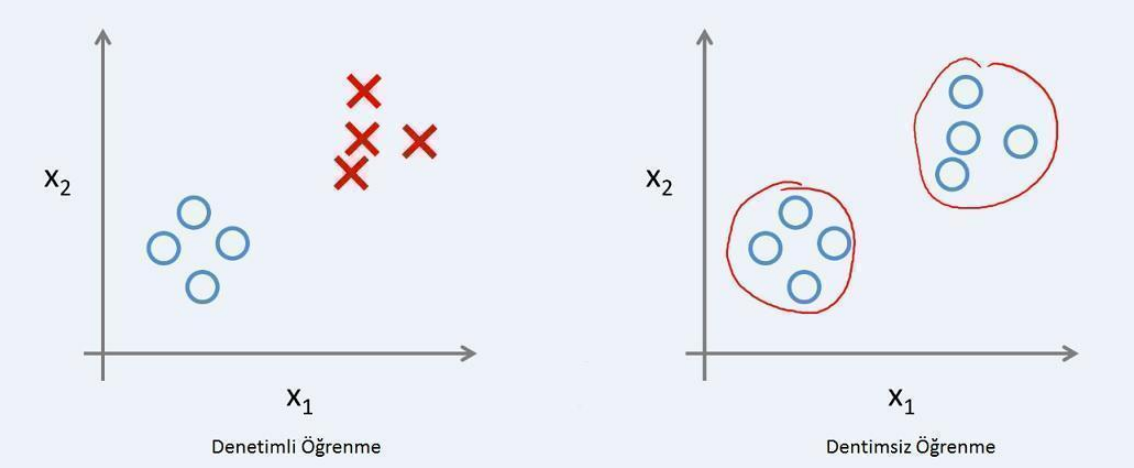
Kümeleme (clustering) ve alt gruplara ayırma denetimsiz öğrenmede kullanılan tekniklerdir. Kümeleme güçlü bir veri analiz aracı olarak, girdi verilerini homojen bir şekilde kümeleme, alt gruplara ayırma işlemini yürütür. Burada homojenlik, girdi verilerinin aynı küme veya alt grup elemanı olarak benzer özelliklere sahip olmaları anlamını taşımaktadır[39]. Doğal dil işleme denetimsiz öğrenme teknikleri, kümeleme algoritması tarafından, insan denetimi olmadan doküman veya kelimeleri kümeleme işlevini gerçekleştirmektedir.

Denetimsiz makine öğrenmesinde, doküman seviyesinde k-ortalama (k-mean)[48] ve hiyerarşik kümeleme (hierarchical clustering) teknikleri[50] oldukça geniş kullanım alanı bulmaktadır. Doküman kümelemede benzer özelliklere sahip dokümanların benzer içeriklere sahip olma varsayımına dayanmaktadır[44].

Doğal dil işleme kelime seviyesinde makine öğrenmesi, girdi verilerinin komşu kelimeleri ile benzerliklerinden yararlanılır. Komşu kelimelerin diğer bir deyişle bağlam (context) benzerlikleri sayesinde kümeleme işlemi yapılır. Kelimelerin cümle içinde kullanımına göre farklı anlamlar yüklenebilmesi ile oluşan anlam ilişkilerinin çözümlenmesi, komşu kelimeler ile beraber ele alındığında yapılabilmektedir[49].

Denetimsiz öğrenmede, denetimli öğrenmeden farklı olarak önceden özellik tanımlaması yapılmaz ve girdi verileri etiketsiz olarak işleme sokulur. Bir örnek vermek

gerekirse, denetimli öğrenmede kadın/erkek verisi bir cinsiyet özelliği olarak etiketli veri durumunda iken, denetimsiz öğrenmede cinsiyet diye bir özellik yoktur. Denetimsiz öğrenme kümelenmelerinde hassasiyet genellikle yüksek değildir.



Şekil 5 Denetimli, denetimsiz öğrenme farkı (Denetimli öğrenmede etiketli veriler yuvarlak ve çarpı işareti ile iki ayrı sınıfa ayrılırken denetimsiz öğrenmede etiketsiz veriler kümelenmekte)

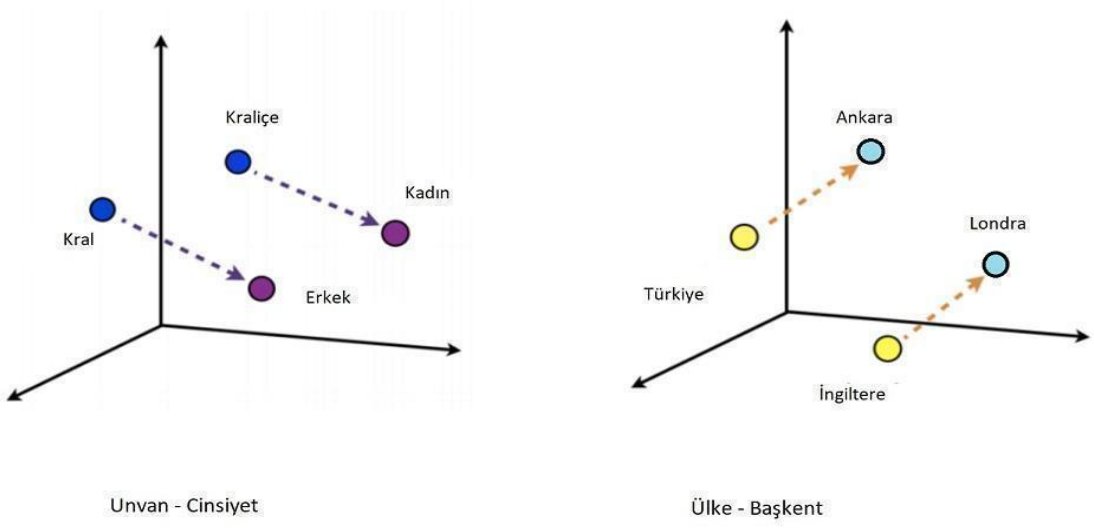
2.3 Kelime vektörlerin kosinüs benzerliği

Kelimelerin benzerliklerine örnek olarak "Fransa" ve "İtalya" kelimeleri verilebilir. Ülke adlarını temsil eden kelimeler olarak benzerlik taşımaktadırlar. Kelimelerin anlam ilişkilerine paralel olarak, bu kelimelere ait vektörlerin kümelenmesi beklenmektedir. Word2vec'ten elde edilen kelime vektörlerinin arasındaki kosinüs benzerliği, kümelenmenin ölçümlenebilmesi için uygun bir yöntemdir.

Kelimelerin söz dizimi ilişkilerine göre, bu kelimelere ait vektörlerin kümelenebildiklerini Mikolov belirtmektedir. Bir örnek vermek gerekirse, İngilizcede "Big" kelimesi "Bigger" kelimesi ile söz dizimi benzerliği ilişkisi içindedirler. Benzer şekilde "Small" kelimesi "Smallest" ile söz dizimsel benzerlik taşımaktadır. "Small" kelimesi ile "Biggest" kelimesi arasında nasıl bir ilişki vardır? Kelime vektörlerin basit cebirsel işlemlerle bu soruya cevap verebildikleri anlaşılmıştır[8].

(‘biggest’) – (‘big’) + (‘small’) İşlemi ile elde edilen vektörün en yakın kosinüs benzeri (‘smallest’) vektörü olduğu, vektörlerin kosinüs benzerliğinden faydalanılarak hesaplanabilmektedir.

Kelimelerin arasındaki anlambilim ilişkilerine benzer ilişkiler, Word2vec ile elde edilen kelime vektörlerinin kümelenmelerinde görülmektedir[10]. “İngiltere” ve “Londra” kelimeleri arasındaki ülke ve başkent ilişkisine benzer bir şekilde, bu kelimelere ait vektörlerin kümelendiği görülmektedir. Mikolov Word2vec ile elde edilen vektörlerin kosinüs benzerliklerinin SemEval-2012 Görev-2 ilişkisel benzerlik ölçüleri[52] ölçümlerine uygun olduğunu belirtmektedir. Kelime çiftleri arasındaki ilişkisel benzerlik karşılaştırıldığında “köpek:havlama”, “kedi:miyavlama” kelime çiftlerinin “araba:korna” kelime çiftlerine göre daha fazla ilişkisel benzerlik gösterdikleri söylenebilir.



Şekil 6 Kelimeler arasındaki ilişkisel benzerlik

Uzunlukları sıfırdan büyük iki vektör arasındaki kosinüs benzerliği, iki vektör arasındaki kosinüs açı değerini ifade etmektedir. İki vektör arasındaki kosinüs değerinin sıfır olması iki vektörün aynı yönde olduğunu, doksan derece olması, iki vektörün birbirine dik olduğunu ifade etmektedir. İki vektör arasındaki kosinüs benzerliğinin hesaplanması aşağıdaki formül ile ifade edilmektedir.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|}$$

A vektörü ile B vektörünün iç çarpım (dot product) değeri bize sayısal bir değer vermektedir. Vektör uzunlukları da sayısal değerlerdir. Bu vektörlerin sayısal çarpımlarının, uzunlukları çarpımına bölünmesi, yine sayısal olan kosinüs benzerliğini vermektedir.

Doğal dil işleme doküman bulma alanında sıklıkla vektör uzay modellemesi[53] (Vector Space Model) kullanılır. Vektör uzay modellemesinde, dokümanlar vektörler ile temsil edilmektedir. Dokümanlar arası benzerlikler kosinüs benzerliği ile hesaplanmaktadır[54].

Kelime düzeyinde benzerlikleri bulmak için vektör uzay modellemesinden faydalanılmaktadır. Kelime vektörlerinin, vektör uzayında kümelenmesinin ölçümü için vektörlerin kosinüs benzerliklerinden faydalanılmaktadır.

$$A \,.\, B = \begin{vmatrix} a1 \\ a2 \\ a3 \end{vmatrix} \,.\, \begin{vmatrix} b1 \\ b2 \\ b3 \end{vmatrix} = a1b1 + a2b2 + a3b3$$

Şekil 7 İki vektörün iç çarpımı (dot product) sonucu sayısal değer elde edilmesi

İki vektör arasında Cos (θ) = 1 olduğu durum iki vektörün aynı açı değerine sahip olduğu anlamına gelir. Kelime vektörlerde de iki vektör arasındaki açı farkının sıfır olması iki kelime vektörün muhtemel aynı kelime vektör olduğu anlamındadır.

Tez çalışmaları sırasında eğitim için kullanılan derlemden elde edilen vektörlerden (‘istanbul’) ile (‘ankara’) vektörleri arasındaki Cos(θ) değeri 0,645 olarak hesaplanmıştır. Eğitimde kullanılan bazı parametrelerin değişmesi ile yeniden eğitim yapılarak elde edilen sonuçlarda farkların olması beklenmelidir. Aynı parametrelerle derlem iki kere eğitime tabi tutulduğunda, projeksiyon kademesindeki başlangıç değerlerinin rastgele seçilmiş olmasından kaynaklı farklar gözlemlenecektir.

2.4 Doğal dil işlemede bir-sıfır vektörleri (one-hot vector)

Doğal dil işleme eğitiminde kullanılan derlemde bulunan bütün kelimelerden bir sözlük çıkarıldığında, bu sözlükteki her kelime benzersiz olmalıdır. Bu sözlükte yer alan kelime adeti kadar boyuta sahip bir vektör uzayında, kelimeler 1/0 kodlaması ile temsil edilebilir. Bu durumda her kelimenin, vektör uzayında bir boyutu temsil ettiği düşünülebilir. Bir tane bir ve sıfırlardan oluşan ve sözlükteki her benzersiz kelimeyi ayırt edebilen vektörler İngilizcede “one-hot vector” olarak adlandırılmaktadır.

Bir tane bir ve sıfırlardan oluşan, ayırt edici vektörler için bir örnek vermek gerekirse, Word2vec’te eğitim için kullanılacak derlem “Bu sabah hava çok güzel” cümlesi olarak belirlensin. Bu derlemden elde edilen sözlüğün bir tane bir ve sıfırlardan oluşan ayırt edici vektörleri;

Bu = [1, 0, 0, 0, 0]

sabah = [0, 1, 0, 0, 0]

hava = [0, 0, 1, 0, 0]

çok = [0, 0, 0, 1, 0]

güzel = [0, 0, 0, 0, 1]

Şeklinde olacaktır.

2.5 Kelime ve İçinde Bulunduğu Bağlam (Context)

TDK'da bağlam "bir cümlede, bir konuşmada veya bir metin içinde yer alan herhangi bir kelimenin anlamının daha iyi belirlenebilmesi ve başka anlamlarından ayırt edilebilmesi için, kendisini çevreleyen ve karşılıklı ilişkide bulunduğu öteki öge veya ögelerle oluşturduğu bütün"[11] olarak geçmekte. Bağlam, kelimenin kullanıldığı ortama göre aldığı anlamdır.

Bir kelime kullanıldığı yere göre eğitim, eşya, renk, finans, sağlık, cinsiyet, yaş vb. birçok özelliği üzerinde bulundurabilmektedir. Bağlamsal anlam kuramı alanında dilbilim uzmanı John Rupert Firth[37][38] bir kelimenin anlamını bağlam belirler manasındaki sözüyle "You shall know a word by the company it keeps" dile getirmiştir.

Word2vec'te bağlam yaklaşımından hareket edilmektedir. Üzerinde işlem yapılan kelimenin (hedef/merkez kelime) sağ ve sol kısmındaki komşu kelimelerin diğer bir deyişle bağlam, önceden belirlenmiş bir mesafeye (kelime adedi) kadar olanları alınmaktadır(Window size). Kristina Toutanova ve ekibi tarafından 2003 yılında yapılan bir çalışmada, merkez kelimenin her iki tarafındaki kelimelerin bağlamda kullanılmasının daha iyi sonuç verdiği ortaya koymuştur. Merkez kelimenin sadece sağ tarafındaki veya sadece sol tarafındaki kelimelerin bağlam olarak kullanılması doğru sonuçlar vermektedir. Merkez kelimenin hem sağ, hem de sol tarafındaki kelimelerin bağlam olarak kullanılması durumunda, kelime vektörlerin daha doğru değerler aldığı ifade edilmektedir[41]. Toutanova ve ekibinin araştırmasından çıkan sonuç Word2vec içinde geçerlidir. Word2vec eğitimi sırasında merkez kelimenin her iki tarafındaki kelimeler bağlam olarak kullanılmaktadır.

Mikolov Word2vec eğitimi için, bağlamda kullanılan komşu kelime mesafesinin beş ve vektör boyutunun üç yüz olarak belirlenmesi durumunda, eğitim sonucunda elde edilen vektörlerin yeterince doğru değerler aldıklarını belirtmektedir[12].

Şekil 8 Komşuluk mesafesi üç olarak belirlenmiş bağlam

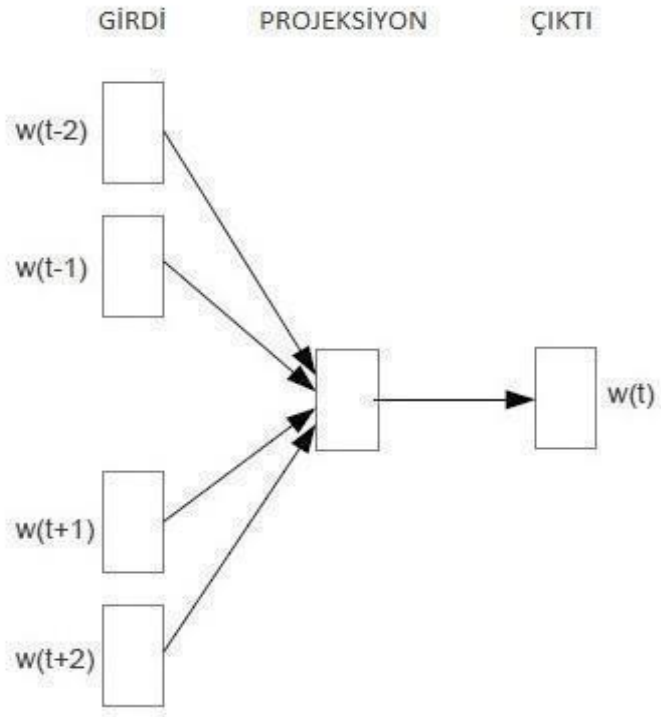
2.6 Word2vec teknikleri

Sinir Ağı dil modeli (NNLM), bir kelime dizisinden sonra gelen kelimenin ne olduğunu tahmin etme mantığı üzerine kurulmuş, istatistik tabanlı ileri tek yönlü bir tekniktir. NNLM tekniğinden farklı olarak, Word2vec tekniklerinde Rumelhart'ın[13] hatanın geri yayılımı tekniğini kullanılmıştır. Komşu kelimelerin merkez kelime ile bağlam ilişkisi içerisinde olması nedeniyle kelime benzerlikleri oluşmaktadır. Word2vec'te bu yaklaşımdan faydalanılmış ve vektör uzayında NNLM'ye göre daha iyi doğruluk değerleri almış vektörlerin (kaliteli vektörler) üretebilmesi sağlanmıştır. Eğitim sırasında benzerlik tahmini yapıldığında, çıktısı doğru olarak bilinen komşu kelimeler, doğru cevap olarak doğrulama tablosunda işaretlenir. Doğrulama tablosunda doğru olarak işaretli vektörler ile yapılan tahmin karşılaştırılır ve karşılaştırma sonucu ortaya çıkan kayıp miktarı hesaplanır. Hesaplanmış olan hata geriye doğru projeksiyon katmanındaki kelime vektörde bulunan boyutlar üzerinde düzeltme yapılarak daha doğru ağırlıklara sahip vektörler elde edilmeye çalışılır. Mikolov ve ekibi "Efficient Estimation of Word Representations in Vector Space"[8] adlı yayınlarında, Word2vec teknikleri ile elde ettikleri vektörlerin arasında, ait oldukları kelimeler arasındaki ilişkiye benzer ilişkiler elde ettiklerini belirtmektedir. Kelimelerin arasında görülen söz dizimi ve anlambilim benzerlikler kelime vektörlere de yansımaktadır.

Word2vec'te yer alan iki teknikte de istatistik tabanlı tekniklere göre hesaplama karmaşıklıkları aza indirilmiş ve bu sayede daha kısa eğitim süreleri elde edilmiştir. Eğitim süresinin kısalması daha büyük derlemlerin pratikte eğitimi için kullanılmalarına imkan tanımıştır[14]. Word2vec'te kelime vektörleri elde etmek için Ardışık Kelimeler Topluluğu (CBOW) ve Komşu kelimeleri tahmin (Skip-gram) adında iki algoritma bulunmaktadır.

2.6.1 Ardışık Kelimeler Topluluğu (CBOW)

Word2vec'te geliştirilen iki algoritmada hesaplama yöntemleri benzerlik göstermektedir. İki algoritmayı birbirinden ayıran, kelimeler arası bağlam ilişkisinin modellenmesidir. Geliştirilen algoritmalardan ilki CBOW algoritmasıdır. Komşu kelimelerin ortasında bulunan merkez kelimenin ne olabileceği tahmini yapmak üzerine modellenmiştir. Bu modellemenin şematik gösterimi, Şekil 9'da görülmektedir.



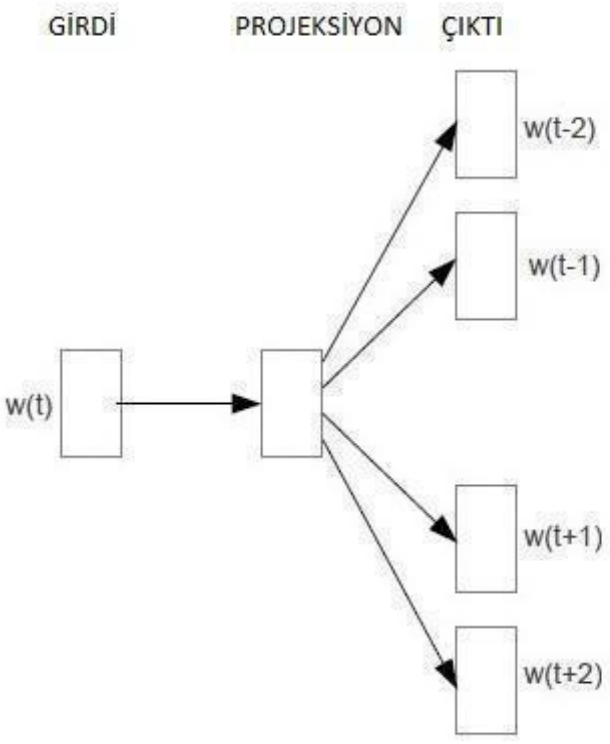
Şekil 9 Ardışık kelimeler topluluğu grafik gösterimi

Bağlam konusunda verilen örnekte (Şekil 8) "Bu sabah hava güneşli ve sıcak güneş gözlüklerinizi yanınıza almayı unutmayınız" cümlesinde komşuluk sınırı üç olarak verilmişti. "Güneş" kelimesi merkez kelime olarak tahmin edilmeye çalışıldığı ifade edilmişti. "Güneş" kelimesinin sol tarafında "güneşli ve sıcak" kelimeleri ve sağ tarafında bulunan "gözlüklerinizi yanınıza almayı" kelimeleri merkez kelime ile bağlam ilişkisi içerisindedirler. Eğitim sırasında bu altı kelime girdi olarak verildiğinde doğru tahmin edilmesi gereken merkez kelime "Güneş" kelimesi olmaktadır. Yapılan tahmin için bir doğrulama tablosu oluşturulduğunda "Güneş" kelimesi doğru kelime

olarak işaretlenmektedir. “Güneş” kelimesinden farklı bir kelime bu altı kelime için doğru kelime olarak tahmin edilirse ortaya hatalı bir tahmin çıkmaktadır. Hatanın miktarına göre projeksiyon katmanında düzeltme yapılarak (hatanın geri yayılımı) doğru vektör ağırlıkları elde edilmeye çalışılmaktadır.

2.6.2 Komşu kelimeleri tahmin (Skip-gram)

Word2vec’te geliştirilen ikinci algoritma Skip-gram algoritmasıdır. Kelimeler arası bağlam ilişkisinin modellemesi CBOW algoritmasından farklıdır. Skip-gram algoritmasında merkez kelime girdi olarak verilmektedir ve komşu kelimelerin hangi kelimeler olabileceği üzerine tahmin yapılması olarak modellenmiştir. Bu modellemenin şematik gösterimi, Şekil 10’da görülmektedir.



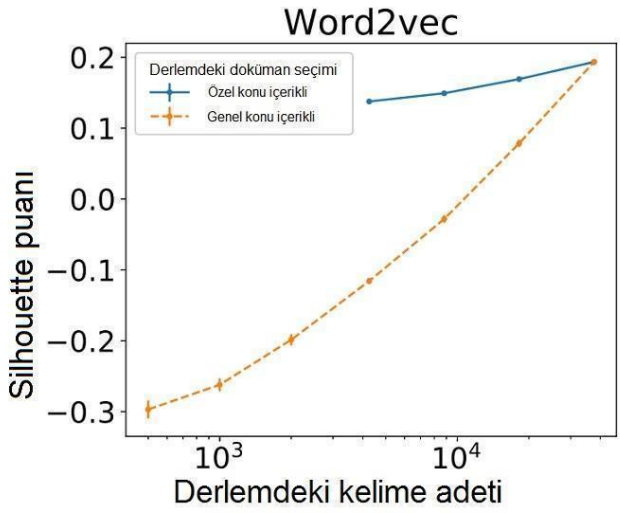
Şekil 10 Komşu kelimeleri tahmin tekniği (Skip-gram) grafik gösterimi

Bağlam konusunda verdiğimiz örnekte (Şekil 8) “Bu sabah hava güneşli ve sıcak güneş gözlüklerinizi yanınıza almayı unutmayınız” cümlesindeki merkez kelime olan “Güneş” kelimesi girdi vektörü olarak işleve verilmektedir. “Güneş” kelimesinin sol

tarafında “güneşli ve sıcak” kelimeleri ve sağ tarafında bulunan “gözlüklerinizi yanınıza almayı” kelimeleri tahmin edilmeye çalışılmaktadır. Mikolov Skip-gram algoritmasının hızlı ve kelime vektörlerinin daha iyi kosinüs benzerliği sonuçları verdiğini belirtmektedir[12].

2.7 Word2vec’de kullanılan derlemin içeriği ve vektörler üzerindeki etkisi

Kelime vektörlerin Word2vec teknikleri ile elde edilebilmesi için diğer makine öğrenmesi tekniklerinde olduğu gibi makine öğrenmesi eğitim sürecinden geçirilmesi gereklidir. Word2vec’de makine öğrenmesi eğitimi yapabilmek için metin içerikli dokümanlardan bir derlem hazırlanmalıdır. Hazırlanan derlem roman, gazete haberleri, sözlük ve benzeri genel içerikli konulardan oluşan kaynaklardan seçilebileceği gibi finans, tıp, psikolojik destek konuşma metinleri gibi belirli bir konu içerikli metinlerden de seçilerek oluşturulabilir. Derlemin belirli bir konu üzerine oluşturulması durumunda, kelimenin o konu üzerinde ifade ettiği belirli bir anlam kategorisi çerçevesinde, eğitimden elde edilen kelime vektörlerde kümelenmeler olması beklenmelidir. Belirli bir konu üzerine oluşturulan derlem, genel içerikli derleme göre daha küçük boyutta bile olsa, anlamsal benzerlik bakımında iyi performans gösterdiği belirtilmiştir[55][58]. Hazırlanan derlemin büyüklüğü, eğitimi etkileyen önemli bir parametredir. Derlemin büyüklüğünün arttırılması oranında, eğitim sonunda elde edilen vektörlerin kelimeyi doğru temsil etme eğilimi artmaktadır [56].

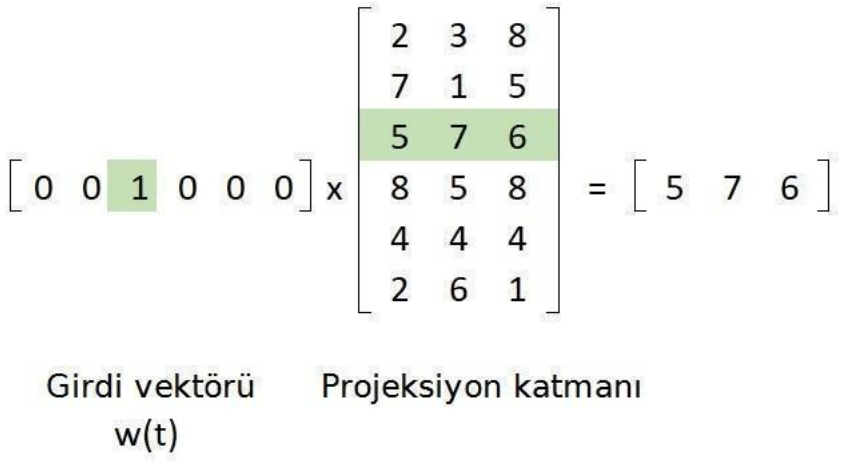


Şekil 11 Özel ve genel konu içerikli metinlerden oluşturulan derlemlerin karşılaştırılması (Silhouette[57] puanı; kelimenin içinde bulunduğu anlam kategorisi "içecek, ülke, eşya vb." ve komşu kelimelerin karşılaştırılması ile elde edilmiştir[56])

Hazırlanan derlem üzerinde, denetimsiz öğrenmeye uygun olarak kelimenin anlamı veya dil bilgisi özellikleri üzerine herhangi bir etiketleme yapılmaz. Derlemde bulunan büyük harflerin küçük harflere çevrilmesi, noktalama işaretlerinin, sayıların derlemden çıkarılması, metinlerde çok tekrarlayan(stop words) "ve", "veya", "ile" gibi benzeri kelimelerin derlemden çıkarılması tavsiye edilmektedir.

2.8 Girdi vektörü ve Projeksiyon katmanı

Kelime vektörlerin elde edilmesi eğitimi aşamasında, ilk adım eğitimde kullanılacak derlemde bulunan kelimelerden bir sözlük (vocabulary) oluşturulmasıdır. Oluşturulan bu sözlükte, derlemde bulunan her kelime temsil edilmeli ve benzersiz olmalıdır(unique). Sözlükte bulunan her kelimenin bir tane bir ve sıfırlardan oluşan ayırt edici vektörleri elde edilir. Elde edilen ayırt edici vektörler, eğitimde girdi vektörü (input) olarak kullanılır.



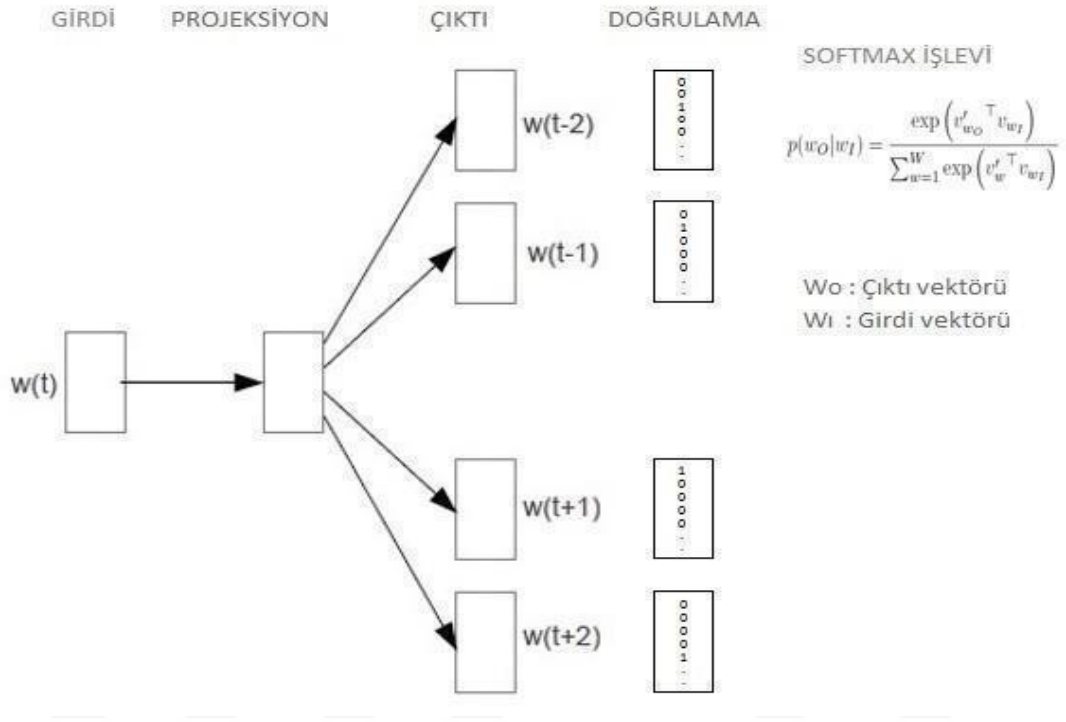
Şekil 12 Girdi vektörü - Projeksiyon katmanı ilişkisi

Şekil 12'de ayırt edici vektör, w(t) vektörü olarak gösterilmiştir. Ayırt edici vektörün girdi vektörü olarak kullanımı ile gösterge indisi gibi davrandığı görülmektedir. Projeksiyon katmanının arama tablosu (lookup table) olarak gördüğü işlev de Şekil 12'de görülmektedir.

Projeksiyon katmanı eğitim sonucunda elde edilmek istenen kelime vektörleri içermektedir. Hatanın geriye doğru düzeltilmesi işlemi ile yapılan hata/kayıp hesaplamaları sonucu, elde edilen hata değeri projeksiyon katmanında hata düzeltmede kullanılmaktadır. Projeksiyon katmanında bulunan vektörler eğitimin başlangıcında sıfırdan farklı rastgele değerler verilerek eğitim işlevine başlatılırlar. Başlangıç değerlerinin sıfırdan farklı olmasının sebebi, sıfır değerine sahip boyutların çarpımlarının sıfır değeri döndürmesidir. Sürekli sıfır değerlikle çarpımlar hatanın düzeltilmesi işlemini yapılamaz hale getirmektedir. Eğitimin her döngüsünde elde edilen hata değeri, projeksiyon katmanında bulunan vektörlerin boyut değerleri üzerinde düzeltme işleminin yapılması ile devam eder. Hata değerleri sayesinde, projeksiyon katmanında bulunan vektörlerin boyut değerleri düzeltilerek sonuçta istenen kelime vektörler elde edilmektedir.

Projeksiyon katmanının boyutsal büyüklüğü, başlangıçta seçilen vektör boyut adedi kadar kolon (bu tezde üç yüz olarak belirlendi ki bu Mikolov'un tavsiye ettiği bir

değerdir) ve derlemden elde edilen sözlükte bulunan kelime adedi kadar satırdan oluşmaktadır.



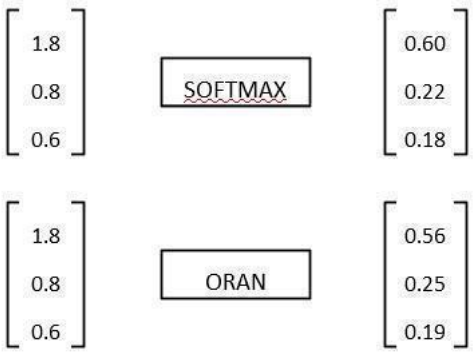
Şekil 13 Skip-gram, Softmax tahmini ile doğrulama tablosu karşılaştırması

2.9 Softmax işlevi

Softmax doğal dil işleme makine öğrenmesi algoritmalarında olasılık dağılımı işlevi olarak kullanılmaktadır[59].

$$S(y_i) = \frac{e^{y_i}}{\sum_j e^{y_i}}$$

Softmax işlevi, sonucun sıfır ile bir arasında pozitif değerde olmasını sağlamaktadır. Softmax işlevi ile elde edilen olasılık dağılımı değerlerinin toplamları bire eşit olmaktadır ve normalizasyon sağlanmaktadır[46].



Şekil 14 Softmax işlevi ile oransal hesaplama arasındaki fark

Şekil 14'te bir dizi reel sayının Softmax işlevi ve bire oranının sonucu karşılaştırılmaktadır. Karşılaştırmada görülmektedir ki Softmax işlevi, oran ile yapılan hesaba göre büyük sayıları bire yaklaştırmakta, küçük sayıları ise sıfıra daha yakın değerlerde hesaplamaktadır.

Şekil 13'deki diyagramda, skip-gram tekniği için eğitim sırasında cümlede hedef/merkez kelime olarak seçilen kelimeyi temsil eden w(t) vektörü ile komşu kelimeleri temsil eden vektörler w(t-2), w(t-1), w(t+1), w(t+2) arasındaki olasılık Softmax işlevi ile hesaplandığını ifade etmektedir. Softmax işlevinden gelen sonucun, doğrulama tablosundaki doğruluğuna göre, hata değeri belirlenerek bu hatanın düzeltmesi geri doğru projeksiyon katmanında yapılmaktadır. Doğrulama tablosu w(t-2) vektörü için w(t-2) vektörünün bir tane bir ve sıfırlardan oluşan ayırıcı vektörü olarak düşünülebilir.

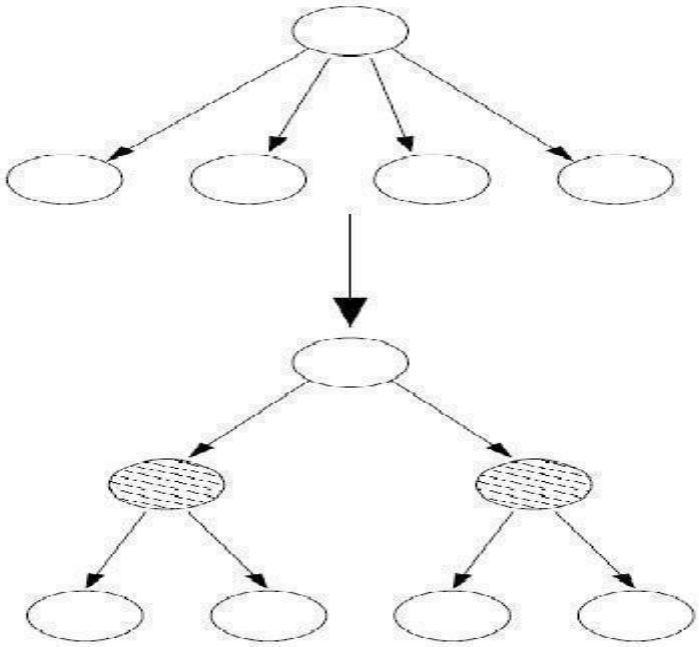
Aşağıda Softmax işlevinin skip-gram tekniğinde her bir komşu kelimenin olasılığı için nasıl formüle edildiği görülmektedir.

$$p(w_O|w_I) = \frac{\exp\left({v'_{w_O}}^{\top} v_{w_I}\right)}{\sum_{w=1}^{W} \exp\left({v'_w}^{\top} v_{w_I}\right)}$$

Formülde, v_w ve v'_w vektörleri w işaretçisi için girdi ve çıktı vektörlerini, W sözlükteki toplam kelime sayısını ifade etmektedir. Her bir komşu kelime için olasılık hesap edilirken sözlükte bulunan bütün değerler üzerinde hesaplama yapıldığı görülmektedir. Sözlükteki bütün kelimelerin olasılık hesabına dahil edilmesi çok fazla hesaplama maliyeti getirmektedir. Hesaplama maliyetini azaltmak için Mikolov hiyerarşik Softmax veya negatif örnekleme algoritmalarının kullanılmasını tavsiye etmektedir[12].

2.10 Hiyerarşik Softmax

Öğrenme süresi uzun olan istatistik tabanlı dil modellemelerinin, öğrenme süresini kısaltma/hızlandırma çalışmalarında Frederic Morin ve Yoshua Bengio tarafında 2005 yılında yeni bir hiyerarşik modelleme teklif edilmiştir[61]. Bu hiyerarşik modelleme kelimelerden oluşturulan ikili ağaç yapısındadır. Kelimelerden ikili ağaç yapımında WordNet adında, kelimelerin anlambilim ilişkileri kullanılarak ağaç yapısında hazırlanmış bir kaynaktan faydalanılmıştır[60]. Hiyerarşik olmayan modelleme ile karşılaştırıldığında iki yüz kat daha hızlı olduğu, fakat az miktarda doğruluk performans kaybı meydana geldiği belirtilmiştir.



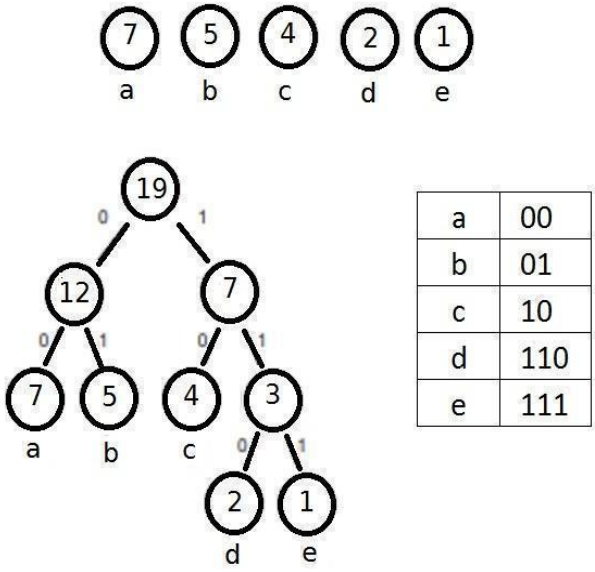
Şekil 15 WordNet ağaç hiyerarşisinden iki ağaç hiyerarşisine evrilmesi

2009 yılında Andriy Mnih ve Geoffrey E Hinton "A scalable hierarchical distributed language model"[62] adlı yayında WordNet gibi bir kaynak gereksinimi olmadan, basit bir yapıda, hızlı ve daha yüksek doğruluk performansı gösteren ikili ağaç üretme algoritmaları geliştirmişlerdir.

Word2vec algoritmalarında ikili Huffman ağaç yapısı kullanılabilir[63]. Hiyerarşik ikili ağaç yapısında, hesaplamalar kökten yapraklara (kelimelere) kadar olan yolun bulunması seviyesine indirgenmiştir. Olasılık hesap maliyeti $p(w_o|w_I)$ seviyesinden $\log p(w_o|w_I)$ ve $\nabla \log p(w_o|w_I)$ sevisine düşürülmüştür.

$$p(w|w_I) = \prod_{j=1}^{L(w)-1} \sigma\left([\![n(w,j+1) = \mathrm{ch}(n(w,j))]\!] \cdot {v'_{n(w,j)}}^{\top} v_{w_I}\right)$$

n(w, j) ifadesi kökten kelimeye kadar olan j inci düğümü ifade eder. Formülde j'nin değeri bir olduğunda n(w, 1) ikili ağacın kökünü ifade etmektedir. Formüldeki ch(n) n inci düğümün çocuğu durumundaki düğümü ifade etmektedir[12].



a	00
b	01
c	10
d	110
e	111

Şekil 16 Huffman ağacı, her hangi bir metinde bulunan harflerin kullanım miktarı göz önüne alınarak yapılan ikili ağaç yapısındaki hiyerarşi

2.11 Negatif örnekleme (Negative Sampling)

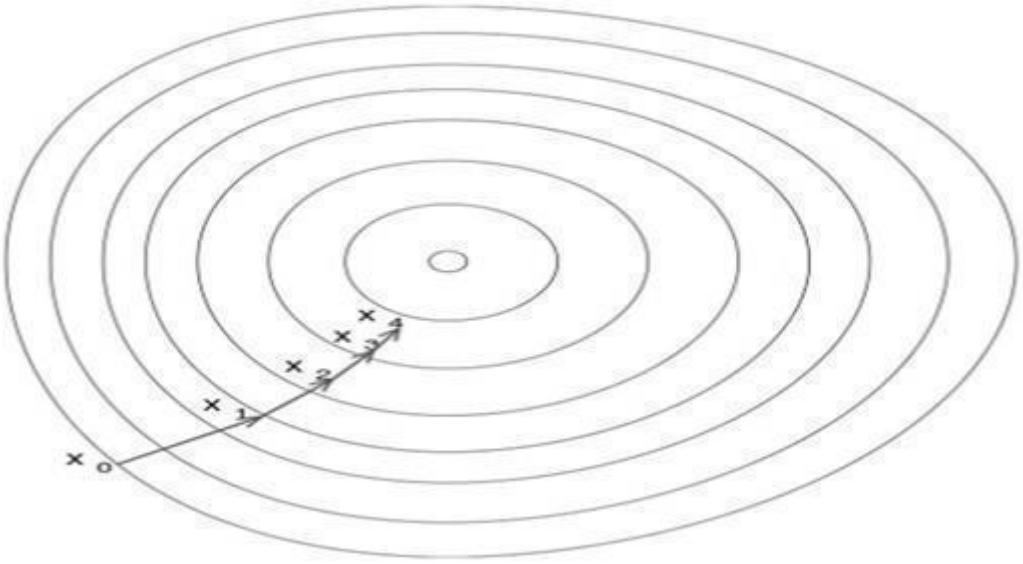
Hiyerarşik Softmax'a alternatif bir teknik, gürültü kontrast tahmini (Noise Contrastive Estimation) görüntü işleme alanında Gutmann ve Hyvarinen[64] tarafından önerilmiş bir algoritmadır. Doğal dil işlemede yavaş olan istatistik tabanlı teknikleri hızlandırmak için ilk uygulaması Mnih and Teh[65] tarafından yapılmıştır. Hiyerarşik Softmax iyi tasarlanmış ikili ağaç yapısına ihtiyaç duyar, bu problemi ortadan kaldırmak için gürültü kontrast tahmini iyi bir alternatif teknik olarak sunulmaktadır. Negatif örnekleme, gürültü kontrast tahmini modelinin basitleştirilmesi ile elde edilmiş bir modeldir[12].

Şekil 13'de projeksiyon katmanı Softmax işlevine tabi tutulduğunda, bütün sözlüğün hesaba dahil edildiği görülmektedir. Sözlüğün tamamı için hesaplama yapılması, çok hesaplama maliyeti getirmektedir. Mikolov hesaplama maliyetini düşürmek ve hiyerarşik Softmax işlevine alternatif olması için negatif örnekleme modelini tavsiye etmektedir. Negatif örnekleme kullanıldığında derlemin büyüklüğüne göre, küçük boyutlu derlemlerde 5-20 kadar kelimenin olasılık hesabına dahil

edilmesini, büyük boyutlu derlemlerde 2-5 kelimenin olasılık hesabına dahil edilmesini tavsiye etmektedir. Olasılık hesabına giren kelimeler, frekansı yüksek kelimelerden ve işlemdeki kelimeye o sırada komşu olmayan kelimelerden seçilmesi tavsiye edilmektedir. Frekansı yüksek kelimelere İngilizce örnek olarak "the", "a", "of", "and", "or" kelimeleri verilebilir.

2.12 Word2vec hata işlevi

Olasılık işlevinden gelen sonuca göre kayıp hesaplanarak boyutsal düzeltme yapılır. Şekil 17'de hata işlevinin görsel benzetmesi görülmektedir. Bu görselde, bir kelime vektör için en doğru değerlerin merkezde olduğu varsayılmaktadır. Her işlem döngüsünde, yapılan hata düzeltme işlemi ile daha doğru vektör değerleri elde ettiği ifade edilmektedir.



Şekil 17 Hatanın geri doğru düzeltilmesi ile daha doğru değerler elde edilmesi

BÖLÜM 3. TÜRKÇE DERLEMİN HAZIRLANMASI

Bu bölümde, bir Türkçe derlem üzerinde Word2vec teknikleri kullanarak çalışma yapabilmek için gerekli yazılım ve pratikte derlemin hazırlanması ile ilgili bilgiler verilmektedir. Tez çalışmaları sırasında, Word2vec teknikleri ile kelime vektörü elde edebilmek ve vektörlerin kosinüs yakınlıkları ile ilgili hesaplamalar için açık kaynak lisanslı (open-source) Python programa dili kullanılmıştır. Python üzerinde Word2vec teknikleri ile makine öğrenmesi eğitimi yapabilmek için "Gensim" adlı açık kaynak lisanslı yazılım kütüphaneleri içeren kaynaktan faydalanılmıştır. Elde edilen vektörlerin kosinüs yakınlıkları ile ilgili hesaplamalar, yine Gensim kütüphaneleri kullanılarak yapılmıştır. Türkçe derlem olarak, genel konu içerikli dökümanlar kullanılmıştır. Bu kaynaklar ve nasıl ulaşılabileceği hakkında ilerleyen kısımlarda bilgiler verilmiştir.

3.1 Python programlama dili hakkında

Python Guido van Rossum tarafından oluşturulmuş ve ilk sürümü 1991 yılında çıkmış olan açık kaynak kodlu bir programlama dilidir. Basic, PHP, PERL, JavaScript vb. programlama dillerinde olduğu gibi Python da yorumlanan (interpreted) bir dildir. Python'da kod yazıldığında, ayrıca bir derleme gerektirmez. Nesne odaklı (object-oriented) programlamaya uygundur ve özyineli (recursive) işlevler yapılabilmektedir. Etkileşimli olarak programlanabilir söyle ki, komut penceresi satırından komut girişi yapılabilir ve komutun ürettiği çıktı, ekrana Python tarafından hemen yazdırılır. Yüksek seviyeli bir dildir, C ve Pascal gibi daha alt seviyeli dillere göre kod yazımı ve okuması kolay ve bu açıdan programlamaya yeni başlayanlar içinde başlangıç eğitimine uygun olan bir programlama dilidir.

Örnek Python kodu;

```
liste = [“Elma”, “Portakal”, “Erik”, “Üzüm”]
i=1
for x in liste:
    print(str(i) +  ". " +  x)
    i += 1
```

Sonuç çıktısı;

1. Elma
2. Portakal
3. Erik
4. Üzüm

3.2 Word2vec için hazır yazılım kütüphanesi Gensim

Gensim, doğal dil işlemede kullanılmak üzere büyük metin kütüphanelerinin işlenebilmesine elverişli açık kaynak kodlu yazılım kütüphanesidir. Kütüphanede Word2vec teknikleri, gizli anlam analizi (latent semantic analysis), terim frekansı – ters metin frekansı (TF-IDF) gibi teknikler yer almaktadır.

Gensim’in Python’da kullanımı, NumPy ve Scipy adlı iki Python bilimsel hesaplama kütüphanesi üzerine kurulu olduğu için öncelikli olarak bu iki kütüphanenin Python’a eklenmesi gerekmektedir. Python’a kütüphane eklemek için Python kütüphane yükleyicisi “pip” kullanılabilir. Bu iki kütüphanenin “pip” ile çalışılan bilgisayar sistemine yüklenebilmesi için internet erişiminin olması gereklidir. Windows 10 işletim sistemi üzerine bu iki kütüphanenin yüklenmesi, aşağıda görülen iki komut ile yapılabilmektedir.

```
C:\...\Python\Scripts>pip install scipy
C:\...\Python\Scripts>pip install numpy
```

Yine Windows 10 işletim sistemi üzerine Gensim kütüphanesini yüklemek içinde aşağıdaki komut kullanılabilir.

```
C:\...\Python\Scripts>pip install –U gensim
```

Bu kütüphanelerin yüklenmesinden sonra Word2vec teknikleri, Gensim kütüphaneleri üzerinden kullanıma hazır hale gelmektedir.

3.3 Derlem hazırlanması

Yapılacak projeye uygun konulardan veya genel içerikli konulardan metinlerin bir araya getirilmesi ile derlem oluşturulabilir. Derlemin büyüklüğü oranında elde edilecek vektörlerin kelimeyi doğru temsil etmesinde (vektör kalitesi) artma olmaktadır. Tez çalışmamız için mümkün olduğunca büyük boyutlu, aynı zamanda pratikte çalışılan bilgisayar sistemini zorlamayacak büyüklükte bir derlem hazırlanmasına karar verilmiştir.

Derlemin eğitime hazırlanması safhasında, derlem üzerinde yapılması tavsiye edilen bazı işlemler aşağıda sıralanmıştır.

- Büyük harflerin, küçük harflere çevrilmesi.
- Derlemde bulunan noktalama işaretleri, parantezler, büyük/küçük işaretleri ve çeşitli sembollerin derlemden çıkarılması.
- Derlemde bulunan hatalı yazılmış kelimelerin düzeltilmesi. Örnek vermek gerekirse, hatalı yazılmış “teşekür” kelimesi yerine doğru yazımla “teşekkür” olarak düzeltilmesi.
- Şaşkınlık belirten ifadeler birçok metinde abartıya kaçmaktadır. Düzeltme yapılması tavsiye edilmektedir. Örnek vermek gerekirse, “aaah”, “aah” yerine “ah” yazılması.
- Sayısal ifadeler, Romen rakamları, tarih bildirimlerin vb. derlemeden çıkarılması tavsiye edilmektedir.

Tez çalışmasında, genel içerikli konulardan Türkçe derlem elde edilmesinde iki farklı kaynaktan yararlanılmıştır. Kaynaklardan ilki GitHub sitesinde “Python ile Türkçe derlem (corpus) hazırlama” konu başlıklı kaynaktır[15]. Kaynakta Word2Vec gibi

işlemlerde kullanılmaya uygun Türkçe metin dosyalarının bulunduğu internet adresi bulunmaktadır[68], tez için bu adresten yararlanılmıştır. İkinci kaynak olarak da Wikimedia dump Türkçe servisi kullanılmıştır[66]. Bu servisten gelen xml içerikli bilgiyi Word2vec eğitimi için düzenlemekte Gensim "corpora.wikicorpus" adlı kütüphane kullanılmıştır[67]. Bu kütüphaneyi kullanan Python kod örneği, Ekler bölümünde Ek-1'de bulunmaktadır. Elde edilen Türkçe derlemde kötü söz vb. hassas içerikler elden geldiğince temizlenmiştir.

Tez çalışması sırasında yukarıda kullanılan kaynaklar kullanılmış ve elde edilen Türkçe derlemde bulunan kelime sayısı yüz altmış bir milyondan fazla olduğu hesaplanmıştır. Derlemden elde edilen sözlük kelimesi üç yüz altmış yedi binden fazladır. Derlem disk üzerinde bin iki yüz yetmiş megabayt yer kaplamaktadır.

3.4 Word2vec eğitim aşaması

Elde edilen derlemin eğitimini yapmak için Gensim "models.word2vec" ve "models.keyedvectors" kütüphaneleri kullanılmıştır[17]. Word2vec makine öğrenmesi eğitimde kullanılabilecek kod örneği ekler bölümünde Ek-2'de bulunmaktadır.

3.5 İşlem adımları

- Python programlama dilinin son versiyonu indirilerek kurulur.
- Python kütüphane yükleyicisini (pip) kullanarak NumPy ve Scipy adlı iki kütüphane kurulur.
- Python kütüphane yükleyicisini (pip) kullanarak Gensim kütüphanesi kurulur.
- Python dilinde kod yazmak için, komut penceresi satır veya "Pycharm" vb. bir editör kullanılabilir.
- Araştırma yapılacak konu ile ilgili veya genel içerikli konulardan oluşan metinler mevcut ise, yukarıda belirttiğimiz GitHub sitesinde bulunan kaynaktan faydalanılabilir ve/veya farklı yöntemler kullanılarak derlem düzenlenebilir. Elde metin yok ise, bu kaynata hazır metinlere ulaşmak için bir internet adresi bulunmaktadır. Wikimedia dump Türkçe servisi kullanıldığında, Gensim'de bulunan "corpora.wikicorpus" adlı kütüphaneyi kullanarak derlem düzenlenebilir.

- Word2vec ile kelime vektörlerin elde edilmesi için, Gensim'de bulunan "models.word2vec" veya "models.keyedvectors" kütüphaneleri kullanılabilir.
- Vektörlerin kosinüs yakınlıklarının hesaplanması için, Gensim'de bulunan "KeyedVectors" kütüphanesi kullanılabilir.

BÖLÜM 4. TESTLER VE SONUÇ

Doğal dilde kullanılan kelimeler, cümle içindeki kullanımlarında, komşu kelimeler ile bağlam ilişkisi kurarak farklı anlamlar yüklenebilirler. Kelimeler arasındaki bağlam ilişkisi, çok yönlü benzerlik ilişkileri kurulmasına neden olmaktadır. Bu benzerlikler Türkçe gibi çekimlemeli dillerde alınan eklere göre de olabilmektedir. İsimlerin birden fazla ek ile sonlanabildiği ve Word2vec'te benzer kelimeleri ararken benzer ekler ile sonlanmış kelimelere de ulaşılabileceği Mikolov tarafından da belirtilmiştir[8]. Türkçe'nin (sondan) eklemeli ve ek açısından zengin bir dil olması göz önüne alınarak, durumun Türkçe için incelenmesi bu bölümde yapılmıştır.

Türkçe derlem üzerinde kelimelerin anlam ve biçim özellikleri göz önüne alınarak Word2vec ile yeterince çalışma yapılmadığı görülmüştür. Bu tez çalışmasında hazırlanan Türkçe derlemden (Word2vec ile) elde edilen vektörler üzerinde incelemeler yapılmıştır. Türkçe kelimelerin anlam ve biçim özelliklerini, kelime vektörlerinin ne kadar yansıtabildiği çalışma sırasında incelenmiştir. Elde edilen sonuçlar ilerleyen kısımlarda verilmiştir.

4.1 Türkçe derlem ve eğitim parametreleri

Tez çalışmasında genel konu içerikli Türkçe derlem kullanılmıştır. Kullanılan derlemde yüz altmış bir milyondan fazla kelime bulunmaktadır. Derlemden elde edilen kelime sözlüğü (benzersiz kelimeler) üç yüz altmış yedi binden fazladır.

Makine öğrenmesi için CBOW algoritması kullanılmıştır. Vektör boyutu üç yüz, komşu kelime (window size) sayısı on olarak alınmıştır. Öğrenme süreci beş çevrimde (EPOCH) yapılmıştır.

Tez çalışmasında programlama dili olarak Python kullanılmıştır. Python üzerinde Word2vec teknikleri ile makine öğrenmesi eğitimi yapabilmek için "Gensim" adlı kütüphaneden faydalanılmıştır. Vektörlerin kosinüs benzerliklerinin hesaplanması için yine "Gensim" kütüphaneleri kullanılmıştır.

Elde edilen sonuçlardaki kosinüs benzerlik değerleri, vektör boyutlarının ağırlığından (boyut değerleri) gelmektedir. Vektör ağırlıkları eğitimde kullanılan derlemin büyüklüğüne, komşu kelime sayısına ve vektör boyutuna göre değişim göstermektedir. Bir derlem aynı parametrelerle iki kere eğitime tabi tutulduğunda, elde edilen vektörlere en yakın kosinüs benzeri vektörün değişmemesi beklenir. Vektörlerin başlangıçtaki ağırlıkları, rastgele değer atanmaları ile ilklendirildiklerinden dolayı sonuç vektörlerinde ve kosinüs benzerlik değerlerinde farklar meydana gelebilmektedir. Oluşan farkların büyük olması durumunda, derlemin yeterince büyük olmadığı düşünülebilir. Çevrim sayısını arttırmak, vektörlerin daha doğru değerler almalarına katkı sağlamaktadır.

4.2 Türkçe kelime vektörlerinin anlamsal kümelenmesi

Bu bölümde, genel içerikli Türkçe derlem üzerinden Word2vec ile elde edilen kelime vektörlerinin, Türkçe kelimelerin anlam ilişkileri açısından nasıl kümelendiği ve ilişkilendiği incelenmektedir.

İlk örnek bir isim olan “Elma” kelimesi ele alınmıştır. (‘elma’) vektörüne en yakın kosinüs benzerliğine (cosine similarity) sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘elma’) vektörünün kosinüs benzerliği sonuçları:

[('çilek', 0.7261281609535217),

('vişne', 0.6900818943977356),

('armut', 0.6884721517562866),

('dut', 0.6787133812904358),

('şeftali', 0.6731953024864197)]

“Elma” kelimesi Türk Dil Kurumu (TDK) güncel Türkçe sözlüğünde,

1. İsim, bitki bilimi Gülgillerden, çiçekleri pembe veya beyaz bir ağaç (Pirus malus).

2. İsim, Bu ağacın kabuğu parlak, sert, kırmızı, sarı ve yeşil renkte, kokusu hoş, tadı ekşi veya tatlı, dokusu gevrek, ufak çekirdekli meyvesi

olarak tanımlanmaktadır.

Türkçe derlemden eğitilerek elde edilen vektörler arasında (‘elma’) vektörüne en yakın kosinüs benzeri olarak (‘çilek’) vektörünün bulunduğu görülmektedir. “Çilek” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, bitki bilimi Gülgillerden, sapları sürüngen, çiçekleri beyaz bir bitki.

2. İsim, bu bitkinin güzel kokulu, pembe, kırmızı renkli meyvesi.

olarak tanımlanmaktadır. İki kelimenin anlam ilişkisi içerisinde olduğu net şekilde görülebilmektedir.

Türkçe derlemden eğitilerek elde edilen vektörler arasında (‘elma’) vektörüne en yakın ikinci kosinüs benzeri (‘armut’) vektörünün bulunduğu görülmektedir. “Armut” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “İsim, bitki bilimi Gülgillerden, çiçekleri beyaz, Türkiye'nin her yerinde yetişen bir ağaç (Pirus communis)” olarak tanımlanmaktadır. İki kelimenin anlam ilişkisi içerisinde olduğu yine net şekilde görülebilmektedir.

Benzer şekilde, elde edilen diğer sonuçlar da yine “Elma” kelimesinin anlamı ile ilişkili, meyve adlarını temsil eden kelimelere ait vektörlerdir.

Şehir adı olan “İstanbul” kelimesinin incelenmesi sonucu (‘istanbul’) vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘istanbul’) vektörünün kosinüs benzerliği sonuçları:

[('ankara', 0.6938591599464417),

('bursa', 0.6174916625022888),

('trabzon', 0.591408371925354),

('üsküdar', 0.581426739692688),

('yenibosna', 0.5711308121681213)]

“İstanbul” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “Türkiye'nin Marmara Bölgesi'nde yer alan illerinden biri” olarak tanımlanmaktadır.

Türkçe derlemden eğitilerek elde edilen vektörler arasında (‘istanbul’) vektörüne en yakın kosinüs benzeri (‘ankara’) vektörünün bulunduğu görülmektedir. “Ankara” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “Türkiye'nin İç Anadolu Bölgesi'nde yer alan illerinden biri, Türkiye'nin başkenti” olarak tanımlanmaktadır. İki kelimenin anlamsal bir yakınlık ilişkisi içerisinde olduğu net şekilde görülmektedir.

(‘istanbul’) vektörüne en yakın kosinüs benzeri olarak (‘ankara’) vektörü ilk sırada bulunmaktadır. “İstanbul” kelimesi ile “Ankara” kelimeleri Türkiye’nin iki önemli şehri olmaları nedeniyle anlam ilişkisi içerisindedirler. Bulunan diğer vektörlere bakıldığında, Bursa ve Trabzon da yine Türkiye’nin diğer önemli şehirleri olarak anlamsal bir yakınlık ilişkisi içerisindedirler. “Üsküdar” ve “Yenibosna” kelimeleri, İstanbul’un iki önemli semti olmaları nedeniyle yine bir anlamsal yakınlık olduğu düşünülebilir.

Özel isim olan “Ahmet” kelimesinin incelenmesi sonucu (‘ahmet’) vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘ahmet’) vektörünün kosinüs benzerliği sonuçları:

[('osman', 0.6758742332458496),

('muhittin', 0.6753208637237549),

('niyazi', 0.6559439897537231),

('halit', 0.6479822993278503),

('mehmet', 0.6463955044746399)]

“Ahmet” kelimesi Türk Dil Kurumu kişi adları sözlüğünde, “Köken: Arapça, Cinsiyet: Erkek. Övülmeye layık, övülmüş.” olarak tanımlanmaktadır.

Türkçe derlemden eğitilerek elde edilen vektörler arasında (‘ahmet’) vektörüne en yakın kosinüs benzeri (‘osman’) vektörünün bulunduğu görülmektedir. “Osman” kelimesi Türk Dil Kurumu kişi adları sözlüğünde,

“Köken: Arapça, Cinsiyet: Erkek.

1. Bir tür kuş veya ejderha.

2. Hz. Muhammet’in damadı, üçüncü halife.

3. Osmanlı İmparatorluğu'nun kurucusu ve ilk hükümdarı.

olarak tanımlanmaktadır.

“Ahmet” kelimesi Türkçede erkek adı olarak kullanılmaktadır. (‘ahmet’) vektörüne en yakın kosinüs benzeri olan vektörler incelendiğinde, “Ahmet” kelimesinin kullanımıyla benzer kullanımda erkek adlarını temsil eden kelimelere/özel isimlere ait vektörler olduğu görülmektedir. Elde edilen sonuçlarda, “Ahmet” kelimesinin kullanım alanı ile ilgili anlamsal bir kümelenme olduğu net şekilde görülmektedir.

Yine bir özel isim olan “Ayşe” kelimesinin incelenmesi sonucu (‘ayşe’) vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘ayşe’) vektörünün kosinüs benzerliği sonuçları:

[('melike', 0.796585202217102),

('cemile', 0.7877158522605896),

('merve', 0.7801972031593323),

('hatice', 0.7799881100654602),

('zeynep', 0.7753742933273315)]

“Ayşe” kelimesi Türk Dil Kurumu kişi adları sözlüğünde, “Köken: Arapça, Cinsiyet: Kız. Rahat ve huzur içinde yaşayan.” olarak tanımlanmaktadır.

Türkçe derlemden eğitilerek elde edilen vektörler arasında (‘Ayşe’) vektörüne en yakın kosinüs benzeri olarak (‘melike’) vektörünün bulunduğu görülmektedir. “Melike” kelimesi Türk Dil Kurumu kişi adları sözlüğünde,

Köken: Arapça, Cinsiyet: Kız.

1. Kadın hükümdar.

2. Padişah karısı.

olarak tanımlanmaktadır.

“Ayşe” kelimesi Türkçede kadın adı olarak kullanılmaktadır. (‘ayşe’) vektörüne en yakın kosinüs benzeri olan vektörler incelendiğinde, “Ayşe” kelimesinin kullanımıyla benzer kullanımda kadın adlarını temsil eden kelimelere ait vektörler olduğu görülmektedir. Yine elde edilen sonuçlarda, “Ayşe” kelimesinin kullanım alanı ile ilgili anlam ilişkisi içerisinde bir kümelenme olduğu net şekilde görülmektedir.

“Ahmet” ve “Ayşe” kelimelerinin özel isim anlam ilişkisi içerisinde kümelenmesine ilave olarak, cinsiyet özelliğine göre de ayrıştıkları (alt grup oluşturdukları) elde edilen sonuçlarda görülmektedir.

Bir cins isim örneği olan “Okul” kelimesinin incelenmesi sonucu, (‘okul’) vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘okul’) vektörünün kosinüs benzerliği sonuçları:

[('okulun', 0.7467690110206604),

('ilkokul', 0.6787807941436768),

('dershane', 0.6465392708778381),

('lise', 0.6133290529251099),

('ortaokul', 0.6094698905944824)]

“Okul” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “İsim, Her türlü eğitim ve öğretimin toplu olarak yapıldığı yer, mektep” olarak tanımlanmaktadır.

Türkçe derlemden eğitilerek elde edilen vektörler arasında (‘okul’) vektörüne en yakın ikinci kosinüs benzeri olarak (‘ilkokul’) vektörü bulunduğu görülmektedir. “İlkokul” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “Zorunlu öğrenim çağındaki kız ve erkek çocuklarının temel eğitim ve öğretimini sağlamak için devletçe açılan veya açılmasına izin verilen dört yıllık okul, ilk mektep, iptidai, iptidai mektep” olarak tanımlanmaktadır. İki kelimenin anlam ilişkisi içerisinde olduğu net şekilde görülmektedir.

Elde edilen ilk vektör, “Okul” kelimesinin sondan ek alması ile oluşturulan kelimeye ait vektör olmuştur. “Okulun” kelimesi, “Okul” isim köküne “-in” ilgi eki alması ile biçimsel olarak türetilmiş halidir.

(‘okul’) vektörüne kosinüs benzeri olarak bulunan diğer vektörlere bakıldığında, “Dershane” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, derslik.

2. İsim, öğrencilere okul dışında para ile ders veren özel kuruluş.

olarak tanımlanmaktadır.

“Lise” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, sekiz yıllık ilköğretimden sonra en az dört yıllık bir eğitimle hayata veya yükseköğretime hazırlayan ortaöğretim kurumu.

2. İsim, eskimiş, üç yıllık ortaokuldan sonra en az üç yıllık bir eğitimle hayata veya yükseköğretime hazırlayan ortaöğretim kurumu.

olarak tanımlanmaktadır.

“Ortaokul” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “İsim, eskimiş, (orta'okul) öğrencileri genel eğitim yoluyla bir yandan hayata, bir yandan da liseye hazırlayan, genellikle üç yıllık ortaöğretim okulu” olarak tanımlanmaktadır. Kelimelerin anlam ilişkisi içerisinde olduğu net şekilde görülmektedir.

4.2.1 Kelime vektörlerinin aritmetik işlemleri ve kelimeler arası anlam ilişkisi

Türkçe derlem ile elde edilen kelime vektörlerinin toplama ve çıkarma işlemleri sonucu yeni vektörler elde edilebilmektedir. Aritmetik işlemler ile elde edilen vektörlerin kosinüs benzerlikleri ve ait oldukları kelimelerin anlamsal ilişkileri bu bölümde incelenmiştir.

Mikolov İngilizce derlemden elde ettiği vektörler üzerinde toplama ve çıkarma işlemi yaparak elde ettiği yeni vektörün kosinüs benzerlerinin, kelimeler arası mantıksal çıkarımlarla örtüştüğünü aşağıdaki örnekle göstermiştir.

(‘king’) - (‘man’) + (‘woman’) = (‘queen’)

Görüldüğü üzere, İngilizce derlem kullanılarak elde edilen vektörler üzerinde yapılan toplama ve çıkarma işlemleri ile (‘queen’) vektörü elde edilmektedir. Buna bağlı olarak, “Türkçe derlem ile elde edilen vektörlerden de benzer sonuçlar çıkabilir mi?” sorusu da yapılan tez çalışmasında incelenen bir diğer konudur. İlgili çalışma ve elde edilen sonuçlar aşağıda ele alınmıştır.

İngilizce örnek ile benzer şekilde incelenen (‘kral’) - (‘erkek’) + (‘kadın’) işleminin sonuç vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘kral’) - (‘erkek’) + (‘kadın’) işlemi sonuç vektörünün kosinüs benzerliği sonuçları:

[('kraliçe', 0.5500485897064209),

('prens', 0.5298552513122559),

('kralın', 0.514844536781311),

('kralı', 0.49624234437942505),

('kraliçenin', 0.46907928586006165)]

Türkçe derlem ile elde edilen sonuç, İngilizce derlemde elde edilen sonuç ile benzerlik göstermektedir. “Queen” kelimesinin Türkçe karşılığı olan “Kraliçe” kelimesine ait

vektör, işlemden gelen sonuç vektöre en yakın kosinüs benzeri vektör olarak bulunmuştur.

(‘kral’) - (‘erkek’) + (‘kadın’) işlemi, soyluluk ifade eden “Kral” kelimesinde bulunan cinsiyet özelliğinin yer değiştirilmesi işlemidir. Kelimelerin anlamı açısından bakıldığında da işlemin sonucu “Kraliçe” kelimesidir. Vektörlerin toplama, çıkarma işlemi sonucu ile kelime anlamının uyumlu olduğu görülmektedir. “Kraliçe” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “İsim, kral karısı veya krallığı yöneten kadın, ece:” olarak tanımlanmaktadır.

Diğer bir örnek olarak, (‘ingiltere’) - (‘londra’) + (‘ankara’) işleminin sonuç vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘ingiltere’) - (‘londra’) + (‘ankara’) işlemi sonuç vektörünün kosinüs benzerliği sonuçları:

[('türkiye', 0.6439434885978699),

('kırıkkale', 0.5729399919509888),

('niğde', 0.5030767917633057),

('eskişehir', 0.4853522777557373),

('tbmm', 0.4850592315196991)]

(‘ingiltere’) - (‘londra’) + (‘ankara’) işlemi, ülke ve şehirleri (veya başkentleri) arasındaki ilişkiyi sorgulamaktadır. Kelimelerin anlamı açısından bakıldığında da işlemin sonucu “Türkiye” kelimesidir. Vektörlerin toplama, çıkarma işlemi sonucu ile kelime anlamlarının uyumlu olduğu görülmektedir.

Benzer şekilde, (‘finans’) - (‘para’) + (‘altın’) işleminin sonuç vektörüne en yakın kosinüs benzerliğine sahip ilk altı kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘finans’) – (‘para’) + (‘altın’) işlemi sonuç vektörü kosinüs benzerliği sonuçları:

İşlemi kosinüs benzerliği sonuçları:

[('bankacılık', 0.439474880695343),

('gayrimenkul', 0.4268363118171692),

('kuyumculuk', 0.4161675274372101),

('mücevherat', 0.41351592540740967),

('mücevher', 0.3932022750377655),

('sigortacılık', 0.3760865330696106)]

"Finans" kelimesi Türk Dil Kurumu bilim ve sanat terimleri sözlüğünde

1. Fon ve sermaye sağlamaya yönelik ticari etkinlik.

2. İktisadın, para ve diğer varlıkların yönetimi konusunu inceleyen bir alt dalı.

3. Para, kredi, bankacılık ve yatırımların yönetimi.

olarak tanımlanmaktadır.

"Para" kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, "İsim, ekonomi devletçe bastırılan, üzerinde değeri yazılı kâğıt veya metalden ödeme aracı, nakit" olarak tanımlanmaktadır. "Altın" kelimesi Türk Dil Kurumu bilim ve sanat terimleri sözlüğü, "Doğada az bulunması dolayısıyla para olarak kullanılan ya da devletlerce para karşılığında saklanan değerli maden" olarak tanımlanmaktadır.

('finans') - ('para') + ('altın') işlemi, bir ticari etkinlik ifadesi olan "Finans" kelimesinden "Para" kelimesinin çıkarılarak değerli bir maden olan "Altın" kelimesinin eklenmesi ile elde edilen vektöre en yakın vektör "Bankacılık" kelimesine ait vektördür. "Bankacılık" kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, bankada yapılan işlemlerin tümü.

,2. İsim, bankacının yaptığı iş

olarak tanımlanmaktadır.

İkinci sırada “Gayrimenkul” kelimesine ait vektör bulunmuştur. “Gayrimenkul” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “Taşınmaz.” olarak tanımlanmaktadır. “Taşınmaz” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. Sıfat, taşınamayan.

2. İsim, hukuk, ev, tarla vb. taşınamayan mülk, gayrimenkul.

olarak tanımlanmaktadır.

Altıncı sırada “Sigortacılık” kelimesine ait vektör bulunmuştur. “Sigortacılık” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “Sigortacının işi” olarak tanımlanmaktadır. “Sigorta” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “İsim, bir şeyin veya bir kimsenin herhangi bir yönden ileride karşılaşabileceği zararı gidermek için önceden ödenen prim karşılığında bu işle uğraşan kuruluşla yapılan iki taraflı bağlantı sözleşmesi” olarak tanımlanmaktadır. Vektörlerin toplama, çıkarma işlemi sonucu ile kelimeler arası anlam ilişkilerinin uyumlu olduğu görülmektedir.

İncelenen (‘spor’) - (‘futbol’) + (‘yüzme’) işleminin sonuç vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘spor’) – (‘futbol’) + (‘yüzme’) işlemi sonuç vektörünün kosinüs benzerliği sonuçları:

[('olimpik', 0.5659219026565552),

('havuzu', 0.524342954158783),

('sporları', 0.5239308476448059),

('havuzları', 0.5116350650787354),

('binicilik', 0.49981582164764404)]

“Spor” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, bedeni veya zihni geliştirmek amacıyla kişisel veya toplu olarak gerçekleştirilen, bazı kurallara göre uygulanan hareketlerin tümü.

2. Sıfat, kullanışı rahat, kolay olan.

olarak tanımlanmaktadır (İşlemde kelimenin beden hareketleri ile ilgili anlamı incelenmiştir. Kelimenin Bitki bilimi ve Hayvan bilimi ile ilgili anlamları yapılan işlemde bulunmamaktadır).

(‘spor’) - (‘futbol’) + (‘yüzme’) işleminden dönen sonuç vektörlerinden ilki “Olimpik” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “Olimpiyatlarla ilgili, olimpiyat ölçülerinde olan” olarak tanımlanmaktadır. “Havuzu” ve “havuzları” kelimeleri “Havuz” kelimesinde türetilmiş kelimelerdir. “Havuz” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “İsim, su biriktirme, yüzme, çevreyi güzelleştirme vb. amaçlarla altı ve yanları mermer, beton benzeri şeylerden yapılarak içine su doldurulan, genellikle üstü açık yer.” olarak tanımlanmaktadır. Yüzme sporunu yapıldığı yer anlamında olan “Havuz” kelimesi, yapılan vektör işlemi ile anlamsal ilişki içerisindedir.

“Sporları” kelimesi, “Spor” kelimesinden türemiş bir kelime olarak biçim ilişkisi içerisindedir. “Binicilik” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, binici olma durumu.

2. İsim, ata binilerek yapılan spor.

olarak tanımlanmaktadır. Vektör işleminde iki spor dalının yer değiştirmesi sonucu elde edilmiş diğer bir spor dalı olarak anlamsal bir sonuç elde edilmiştir denilebilir.

Sonuçlar değerlendirildiğinde, sonuç vektörlerinin, ait oldukları kelimelerin anlam ilişkileri açısından uyumlu olduğu görülmektedir.

Yukarıda Türkçe derlem ile elde edilen vektörler arasındaki kümelenmeler, ait oldukları kelimelerin arasındaki anlam ilişkisi ele alınarak incelenmiştir. Türkçe kelimelerin aralarındaki anlam ilişkilerin, bu kelimelere ait vektörlerde kümelenmeler oluşturduğu görülmektedir. İngilizce derlemden elde edilen vektörler üzerindeki toplama, çıkarma işlemleri ile elde edilebilen anlamsal sonuçların, Türkçe derlem ile elde edilen vektörlerden de elde edilebildiği anlaşılmaktadır. Bir sonraki bölümde, Türkçeye özgü biçimsel özelliklerin bu kelimelere ait vektörlere nasıl yansıdığı ele alınmaktadır.

4.3 Türkçe kelime vektörlerinin biçimsel kümelenmeleri

Türkçe (sondan) eklemeli ve ek açısından zengin bir dildir. Eklemeli dillerin genel özelliği, kelime köklerinin sabit tutulup çeşitli görevleri bulunan yapım ve çekim eklerinin köklere eklenmesidir. Kelimenin köklerine farklı eklerin eklenmesi ile yeni kelimeler türetilmektedir ve dilin söz varlığı bu şekilde oluşmaktadır. Türkçede bütün değişme ve gelişmeler kök ek birleşimine dayanmıştır[9].

Kelime vektörleri kullanılarak benzer kelimeleri ararken benzer ekler ile sonlanmış kelimelere de ulaşılabildiği görülmektedir[8]. Türkçe derlem üzerinden (Word2vec ile) elde edilen kelime vektörlerinin, Türkçeye özgü eklere göre nasıl kümelendiği ve ilişkilendiği bu bölümde incelenmiştir.

İlk incelenecek olan kelime, "Git" fiil köküne "mek" mastar eki ekleyerek türetilmiş "Gitmek" kelimesidir. ('gitmek') vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

('gitmek') vektörünün kosinüs benzerliği sonuçları:

[('dönmek', 0.7897772192955017),

('yetişmek', 0.7705608606338501),

('götürmek', 0.7535400390625),

('inmek', 0.7440905570983887),

('yerleşmek', 0.7398502230644226)]

"mek" mastar eki alarak fiil kökünden türemiş kelimelerin vektörleri kosinüs benzeri olarak kümelenmiştir. Burada kelime vektörlerin kümelenmesi kelimenin biçimsel özelliği ile ilişkili olduğu net şekilde görülmektedir.

"Git" fiil köküne geçmiş zaman kipi ve birinci şahıs ekli alarak türetilmiş "Gittim" kelimesi incelendiğinde en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘gittim’) vektörünün kosinüs benzerliği sonuçları:

[('gitmiştim', 0.8377403616905212),

('gittiğimde', 0.8276962637901306),

('gidiyordum', 0.7992637753486633),

('gidiyorum', 0.7966102361679077),

('gideceğim', 0.7883756160736084)]

İlk beş sırada bulunan “Gitmiştim”, “Gittiğimde”, “Gidiyordum”, “Gidiyorum”, “Gideceğim” kelimeleri “Git” fiil köküne birinci tekil şahıs eki eklenerek türetilmiş çekimli kelimelere ait vektörlerdir. “Gittim” kelimesi ile biçimsel ilişki içerisindeki kelimelere ait vektörlerin kümelendiği net şekilde görülmektedir.

Daha önce vektörler arası anlam ilişkileri incelenirken “Elma” kelimesi ele alınmıştı. “Elmalı turta severim” cümlesindeki “Elmalı” kelimesi için (‘elmalı’) vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğuna aşağıda görülmektedir.

(‘elmalı’) vektörünün kosinüs benzerliği sonuçları:

[('kumluca', 0.7562471628189087),

('akseki', 0.7351764440536499),

('ibradı', 0.7255643606185913),

('karacaören', 0.7211636304855347),

('akçapınar', 0.7149443626403809)]

“Elmalı turta severim” cümlesindeki “Elmalı” kelimesi, Elma meyvesi anlamına gelen kelimeye, kelimeyi sıfat yapan “lı” eki eklenmesi ile türetilmiştir. “Elmalı” kelimesi aynı zamanda Antalya şehrinin ilçesini de ifade etmektedir. Elde edilen (‘kumluca’), (‘akseki’), (‘ibradı’) vektörleri, Antalya şehrinin ilçelerini temsil eden kelimelere ait

vektörlerdir. “Elmalı” kelimesi için elde edilen sonuçların, Antalya şehrine bağlı ilçeler olma anlamındaki anlamsal bir kümelenme olduğu açık şekilde görülmektedir. Buradaki kümelenmenin, kelimenin anlam ilişkisine göre gerçekleştiği görülmektedir.

“Ağaç” isim köküne isimden yer ismi yapan “lık” eki eklenerek türetilmiş olan “Ağaçlık” kelimesi vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğuna aşağıda görülmektedir.

(‘ağaçlık’) vektörünün kosinüs benzerliği sonuçları:

[('ormanlık', 0.8628451228141785),

('çalılık', 0.7839390635490417),

('sazlık', 0.7809475660324097),

('makilik', 0.7765018939971924),

('otluk', 0.772311270236969)]

Elde edilen sonuçlar, isim köküne eklenen, isimden yer ismi yapan “lık”, “lik”, “luk” ekleri ile türetilen yer adlarıdır. Buradaki kümelenmenin, kelimenin biçim ve anlam ilişkisi içerisinde gerçekleştiği görülmektedir.

“Avukat” isim köküne isimden meslek ismi yapan “lık” eki eklenerek türetilmiş olan “Avukatlık” vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘avukatlık’) vektörünün kosinüs benzerliği sonuçları:

[('muhasebecilik', 0.6621850728988647),

('doktorluk', 0.6428958177566528),

('hakimlik', 0.6376224160194397),

('yargıçlık', 0.635696530342102),

('memurluk', 0.593788206577301)]

Elde edilen sonuçlardan ilk beş tanesi, isim köküne eklenen, isimden meslek ismi yapan “lık”, “lik”, “luk” ekleri ile türetilen meslek adlarıdır. Burada kümelenmenin kelimenin biçim ve anlam ilişkisi içerisinde gerçekleştiği görülmektedir.

“Temiz” kelimesine “lik” eki eklendiğinde, sıfattan durum ismi türetilerek “Temizlik” kelimesine ulaşılır. (‘temizlik’) vektörüne en yakın kosinüs benzerliğine sahip ilk beş kelime vektörünün hangileri olduğu aşağıda görülmektedir.

(‘temizlik’) vektörünün kosinüs benzerliği sonuçları:

[('temizleme', 0.5787885785102844),

('temizliği', 0.5512673258781433),

('banyo', 0.5476162433624268),

('kumlama', 0.5201424360275269),

('tamirat', 0.5156590342521667)]

“Temizlik” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, temiz olma durumu, arılık, saffet, nezafet.

2. İsim, temiz durma veya tutma durumu.

3. İsim, temizleme işi.

4. İsim, argo, ortadan kaldırma, yok etme, öldürme.

olarak tanımlanmaktadır.

“Temizlik” kelimesi için ilk sırada bulunan “Temizleme” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, temizlemek işi

2. İsim, yüzeylere yapışmış leke ve kirlerin giderilmesi, çözelti veya asıltı durumuna getirilmesi olayı.

olarak tanımlanmaktadır. Biçimsel açıdan bakıldığında “Temiz” kelimesinden türetilmiş bir kelimedir. İki kelime anlam ilişkisi ve biçim ilişkisi içerisinde olduğu görülmektedir.

“Temizlik” kelimesi için ikinci sırada bulunan “Temizliği” kelimesi, “Temizlik” kelimesinin belirtme hal eki alarak, ünsüz yumuşaması ile türetilmiştir. İki kelime biçimsel ilişki içinde olduğu açıktır.

Üçüncü sırada bulunan “Banyo” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, yapılarda, içinde yıkanılan bölüm.

2. İsim, banyo küvetinde yıkanma işi.

olarak tanımlanmaktadır. İki kelime yine anlam ilişkisi içerisindedir.

Dördüncü sırada bulunan “Kumlama” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “1. İsim, çam türü ağaçlarda yıl halkaları arasındaki görüntü ayrımını daha da belirtmek için yüzeye, hava basıncından yararlanarak kum püskürtme.” olarak tanımlanmaktadır.

“Temizlik” kelimesi için ilk beş sırada “lik” ekli kosinüs benzeri kelime vektör bulunmamaktadır. Elde edilen vektörler “Temizlik” kelimesinin anlam ve/veya biçim ilişkisi içerisindeki kelimelere ait vektörler olduğu görülmektedir.

4.3.1 Kelime vektörlerinin aritmetik işlemleri ve kelimeler arası biçim ilişkisi

Türkçe derlem ile elde edilen vektörlerin, Türkçenin biçimsel özelliklerine göre nasıl kümelenebildikleri bir önceki bölümde incelenmiştir. Bu bölümde, Türkçenin ek bakımından zenginliğinin kelime vektörlere nasıl yansıdığı, vektörler üzerinde yapılan toplama, çıkarma işlemleri ile incelenmiştir. Bu kısımda “Vektörlerin toplanması, çıkarılması ile elde edilen anlamsal sonuçlara benzer olarak biçimsel sonuçlar da çıkarılabilir mi?” sorusu incelenmiştir.

Biçimsel kümelenmenin incelendiği bölümde incelenen “Gitmek” kelimesine bakılacak olursa (‘gitmek’) vektörünün kosinüs benzeri olarak ilk sırada bulunan (‘götürmek’) vektörüdür. “Götürmek” kelimesine vektörlerin toplanması ve çıkarılması ile nasıl ulaşılabilir sorusunun cevabı; “Gitmek” kelimesinden “Git” fiil kökünü çıkarıp yerine “Götür” fiil kökü eklenmesi olacağı ilk akla gelen cevaptır.

(‘gitmek’) - (‘git’) + (‘götür’) İşlemi sonuç vektörü kosinüs benzerliği sonuçları:

[('götürmek', 0.7065088748931885),

('yetişmek', 0.5844410061836243),

('götürülmek', 0.5795775651931763),

('binmek', 0.5781220197677612),

('uğurlamak', 0.561299204826355)]

“Götürmek” kelimesine, kelime vektörler üzerinde toplama ve çıkarma işlemi yaparak ulaşılabileceği bu incelemede görülmektedir.

(‘yapraklı’) vektörüne, vektörlerin toplanması ve çıkarılması ile nasıl ulaşılabildiği aşağıda incelenmiştir.

(‘çiçekli’) - (‘çiçek’) + (‘yaprak’) İşlemi sonuç vektörünün kosinüs benzerliği sonuçları:

[('yapraklı', 0.6582359671592712),

('dallı', 0.6488081812858582),

('dişbudak', 0.6367601752281189),

('otu', 0.624358594417572),

('yapraklar', 0.61882483959198)]

“Yapraklı” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “Sıfat, yaprağı olan” olarak tanımlanmaktadır. “Çiçekli” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “Sıfat, çiçeği veya çiçek resimleri olan” olarak tanımlanmaktadır.

(‘yapraklı’), (‘çiçekli’) vektörlerinin ait olduğu kelimeler isim köklüdür ve benzer ek alarak türetilmiş sıfatlardır. Yukarıdaki vektör işlemleri ile elde edilen vektörün, kosinüs benzeri olarak ilk sırada (‘yapraklı’) vektörü olduğu görülmektedir.

(‘saydamlık’) vektörüne, vektörlerin toplanması ve çıkarılması ile nasıl ulaşılabildiği aşağıda incelenmiştir.

(‘tazelik’) - (‘taze’) + (‘saydam’) İşlemi sonuç vektörünün kosinüs benzerliği sonuçları:

[('saydamlık', 0.4215427339076996),

('opak', 0.3784925937652588),

('erçivan', 0.3675283193588257),

('görüntüleme', 0.36332154273986816),

('tipindedir', 0.3586195111274719)]

“Tazelik” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde,

1. İsim, taze olma durumu, körpelik, taravet.

2. İsim, mecaz, dinç, diri, canlı olma durumu.

olarak tanımlanmaktadır.

“Saydamlık” kelimesi Türk Dil Kurumu güncel Türkçe sözlüğünde, “İsim, saydam olma durumu, şeffaflık.” olarak tanımlanmaktadır.

(‘saydamlık’), (‘tazelik’) vektörlerinin ait olduğu kelimeler isim köklüdür ve benzer ek alarak türetilmiş isimlerdir. Yukarıdaki vektör işlemi sonucu, ilk sırada kosinüs benzeri olarak bulunan (‘saydamlık’) vektörüne en yakın kosinüs benzeri vektör olarak ulaşılmıştır.

4.4 Sonuç

Bu tezde genel konu içerikli Türkçe derlem kullanılarak (Word2vec ile) elde edilen kelime vektörlerinin kümelenmesi ile ilgili inceleme, dilbilimin iki alt dalı olan anlambilim ve biçim bilimi (dil morfolojisi) göz önüne alınarak yapılmıştır. Anlambilim açısından yapılan incelemede, Türkçe derlem ile elde edilen vektörlerin ait oldukları kelimeleri ne kadar doğru temsil edebildikleri ve nasıl kümelendikleri incelenmiştir. İkinci olarak ele alınan konu, Türkçeye özgü eklere göre biçim özellikleri açısından kelime vektörlerinin nasıl kümelendiği ve ilişkilendiği incelenmiştir. Bu konularda yapılan literatür taramasında, İngilizce derlem ile elde vektörlerin anlambilim açısından gösterdikleri kümelenme performansı üzerine yeterince araştırma olmasına karşın Türkçe üzerinde yeterince çalışma olmadığı görülmüştür.

Kelime vektörlerinin kümelenmeleri ile ilgili elde edilen sonuçlar ve değerlendirmeler aşağıda özetlenmiştir.

1- Kelimelerin anlam ilişkileri göz önüne alınarak, kelime vektörlerinin nasıl kümelendikleri incelenmiştir.

- “Elma” kelimesine ait vektörün kosinüs benzerliklerinde, kelimenin anlam ilişkisi içerisinde olduğu diğer meyve adlarına ait vektörlerle kümelendiği görülmüştür.
- “İstanbul” kelimesine ait vektörün kosinüs benzerliklerinde, kelimenin anlamı ile ilgili olarak şehir adlarını temsil eden “Ankara”, “Bursa”, “Trabzon” kelimeleri anlam ilişkisi içerisindedirler. “Üsküdar” ve “Yenibosna” kelimeleri İstanbul’un semtlerini temsil etme anlam ilişkisi içerisinde kümelenmişlerdir.
- “Ahmet” kelimesi kosinüs benzerliklerine bakıldığında, Türkçede kullanılan erkek adı anlam ilişkisi içerisinde diğer erkek adları ile beraber kümelendiği görülmektedir.
- “Ayşe” kelimesi kosinüs benzerliklerine bakıldığında, Türkçede kullanılan kız adı anlam ilişkisi içerisinde diğer kız adları ile beraber kümelendiği görülmektedir.
- “Okul” kelimesi kosinüs benzerliklerinde, “İlkokul”, “Dershane”, “Lise”, “Ortaokul” kelimeleri anlamsal ilişki içerisinde kümelendikleri görülmektedir. “Okulun” kelimesi biçimsel ilişki içerisindedir.

Türkçe derlemden elde edilen vektörlerin kümelenmesi, ait oldukları kelimelerin anlam ilişkileri, biçim ilişkileri veya her iki ilişki ile olabildiği elde edilen sonuçlarda görülmüştür. Kelimelerin anlam ilişkileri göz önüne alınarak incelendiğinde, kümelenmelerin alt kırılımlara inebildiği görülmektedir. Kişi adlarının kümelenmesinde, erkek adları ile kadın adlarının ayrı ayrı kümelenmeleri alt kırılıma örnek teşkil etmiştir.

2- Kelime vektörlerin toplanması, çıkarılması ile vektörler elde edilebilmektedir. İşlem ile elde edilen vektörlerin kosinüs benzerliklerinin, yapılan işlemin anlamı ile uygunluğu açısından incelemesi yapılmıştır.

- (‘kral’) - (‘erkek’) + (‘kadın’) işleminin sonucu elde edilen vektörün, kosinüs benzeri vektörler arasında ilk sırada (‘kraliçe’) vektörü bulunmuştur. İşlem ile sonuç vektörünün, ait oldukları kelimelerin anlam ilişkileri ile uyumlu olduğu görülmektedir.
- (‘ingiltere’) - (‘londra’) + (‘ankara’) işleminin sonucu elde edilen vektörün kosinüs benzeri vektörler arasında ilk sırada (‘türkiye’) vektörü bulunmuştur. İşlem ile sonuç vektörünün, ait oldukları kelimelerin anlam ilişkileri ile uyumlu olduğu görülmektedir.
- (‘finans’) – (‘para’) + (‘altın’) işleminin sonucu elde edilen vektörün kosinüs benzeri vektörler arasında ilk sırada (‘bankacılık’) vektörü bulunmuştur. İşlem ile sonuç vektörünün, ait oldukları kelimelerin anlam ilişkileri ile uyumlu olduğu görülmektedir.
- (‘spor’) – (‘futbol’) + (‘yüzme’) işleminin sonucu elde edilen vektörün kosinüs benzeri vektörler arasında (‘havuzu’), (‘havuzları’), vektörü bulunmuştur. Bulunan vektörlere ait kelimeler “Havuz” kelimesinden türemiştir. Havuz, yüzme sporunun yapıldığı yer olarak anlamsal ilişki içerisindedir. “Binicilik” kelimesi de “Spor” kelimesinden iki spor dalının yer değişimi olarak bulunan üçüncü bir spor dalını temsil etmesi açısından anlamsal bir ilişki içerisindedir. “Sporları” kelimesi “Spor” kelimesi ile biçimsel ilişki içindedir. İşlem ile sonuç vektörlerin, ait oldukları kelimelerin anlam ve biçim açısından ilişkili olduğu görülmektedir.

Türkçe derlemden elde edilen vektörlerin kelimenin anlamına göre kümelendiği gözlemlenmiştir. Vektörlerin toplanması, çıkartılması işlemleri ile elde edilen yeni vektörlerin, İngilizce derlemlerle yapılan çalışmalarla benzerlik gösterdiği elde edilen sonuçlardan görülmektedir.

3- Kelimelerin biçim özellikleri göz önüne alınarak, kelime vektörlerinin nasıl kümelendikleri incelenmiştir.

- (‘gitmek’) vektörü kosinüs benzerliklerinde ilk beş sırada (‘dönmek’), (‘yetişmek’), (‘götürmek’), (‘inmek’), (‘yerleşmek’) vektörleri bulunmuştur. Bulunan vektörlerin temsil ettikleri kelimeler, fiil köküne “mek” mastar eki alarak türetilmiştir. Burada kümelenmenin kelimelerin biçim özelliğine göre olduğu görülmektedir.
- (‘gittim’) vektörü kosinüs benzerliklerinde ilk beş sırada (‘gitmiştim’), (‘gittiğimde’), (‘gidiyordum’), (‘gidiyorum’), (‘gideceğim’) vektörleri bulunmuştur. Bulunan vektörlerin ait oldukları kelimeler, “Git” fiil köküne birinci tekil şahıs eki eklenerek türetilmiş çekimli fiillerdir. Burada kümelenmenin kelimelerin biçim özelliğine göre olduğu görülmektedir.
- (‘elmalı’) vektörü kosinüs benzerliklerinde ilk üç sırada (‘kumluca’), (‘akseki’), (‘ibradı’) vektörleri bulunmuştur. Bulunan vektörlerin ait oldukları kelimeler, Antalya şehrinin ilçeleri olmaları anlam ilişkisi içerisinde kümelenmişlerdir.
- (‘ağaçlık’) vektörü kosinüs benzerliklerinde ilk beş sırada (‘ormanlık’), (‘çalılık’), (‘sazlık’), (‘makilik’), (‘otluk’) vektörleri bulunmuştur. Bulunan vektörlerin ait oldukları kelimeler, isimden yer ismi yapan “lık”, “lik”, “luk” ekleri ile türetilen yer adlarıdır. Burada kümelenmenin kelimelerin biçim ve anlam ilişkisi içerisinde olduğu görülmektedir.
- (‘avukatlık’) vektörü kosinüs benzerliklerinde ilk beş sırada (‘muhasebecilik’), (‘doktorluk’), (‘hakimlik’), (‘yargıçlık’), (‘memurluk’) vektörleri bulunmuştur. Bulunan vektörlerin ait oldukları kelimeler, isimden meslek ismi yapan “lik”, “lık”, “luk” ekleri ile türetilen meslek adlarıdır. Burada kümelenmenin kelimelerin biçim ve anlam ilişkisi içerisinde olduğu görülmektedir.
- (‘temizlik’) vektörü kosinüs benzerliklerinde en yakın beş vektörden dördü (‘temizleme’), (‘temizliği’), (‘banyo’), (‘kumlama’) vektörleri olarak bulunmuştur. Burada kümelenmenin kelimelerin biçim ve anlam ilişkisi içerisinde olduğu görülmektedir.

Kelimelerin biçim özellikleri göz önüne alınarak incelendiğinde, vektörlerin kelimelerin aldıkları eklere göre kümelenebildiği elde edilen sonuçlardan görülmektedir. İsim köklü kelimelerde, eklere göre kümelenmelerin kelimelerin anlam ilişkilerini de içerebildiği görülmektedir. "Avukatlık" kelimesinin içinde bulunduğu kümelenmede "lık", "lik", "luk" ekleri bulunan kelimelerden, sadece meslek ismi olan kelimeler yer almıştır. Benzer şekilde "Ağaçlık" kelimesinin içinde bulunduğu kümelenmede "lık", "lik", "luk" ekleri bulunan kelimelerden, sadece yer isimleri olan kelimelerin yer aldığı görülmektedir.

4- İşlem ile elde edilen vektörlerin kosinüs benzerliklerinin, yapılan işlem ile eklerinin uygunluğu açısından incelenmesi yapılmıştır.

- ('gitmek') - ('git') + ('götür') işlemi ile "Gitmek" kelimesindeki "Git" fiil kökünün "Götür" fiil kökü ile değiştirilme işlemi yapılmıştır. İşleminin sonucu elde edilen vektörün kosinüs benzeri vektörler arasında ilk sırada ('götürmek') vektörü bulunmuştur.
- ('çiçekli') - ('çiçek') + ('yaprak') işlemi ile "Çiçekli" kelimesinden "Çiçek" isim kökünün "yaprak" isim kökü ile yer değiştirme işlemi yapılmıştır. İşleminin sonucu elde edilen vektörün kosinüs benzeri vektörler arasında ilk sırada ('yapraklı') vektörü bulunmuştur.
- ('tazelik') - ('taze') + ('saydam') işlemi ile "Tazelik" kelimesinden "taze" ön adının (sıfat) "saydam" ön adı ile yer değiştirme işlemi yapılmıştır. İşlemin sonucu elde edilen vektörün kosinüs benzeri vektörler arasında ilk sırada ('saydamlık') vektörü bulunmuştur.

Türkçenin ek bakımından zengin bir dil olması yapılan incelemelerde göz önüne alınmıştır. Vektörlerin ait oldukları kelimelerin biçim özelliklerine göre de kümelendiği elde edilen sonuçlarda gözlemlenmiştir.

Fiil köklü kelimelerin biçimsel özelliklerine göre, kelime vektörlerinin doğruluk kalitesinin iyi olduğu sonuçlardan görülmektedir. İsim köklü kelimeler, cümledeki anlamlarına ve aldıkları eklere göre kelime vektörlerinde kümelenmelere neden olmuşlardır. Ayrıca, isim köklü kelimeler bazen anlam özelliklerine göre, bazen biçim

özelliklerine göre, bazen de hem anlam hem de biçim özelliklerinin karışımı ile kümelendiği elde edilen sonuçlarda görülmektedir.

Derlemden elde edilen kelime vektörlerin, en yakın kosinüs benzerine sahip ilk beş vektör ile ilgili örnekler Ek-3'te verilmiştir.

EK'LER

Ek-1 Wikimedia dump Türkçe servisinin xml içerikli yapısının düzenlenmesi için Python kodları

```
#dosya adı wikipreprocesi.py

from __future__ import print_function

import os.path

import sys

from gensim.corpora import WikiCorpus

import xml.etree.ElementTree as etree

import warnings

import logging

import string

from gensim import utils

def tokenize_tr(content,token_min_len=2,token_max_len=50,lower=True):

        if lower:

                lowerMap = {ord(u'A'): u'a',ord(u'A'): u'a',ord(u'B'): u'b',ord(u'C'): u'c',ord(u'Ç'): u'ç',ord(u'D'): u'd',ord(u'E'): u'e',ord(u'F'): u'f',ord(u'G'): u'g',ord(u'Ğ'): u'ğ',ord(u'H'): u'h',ord(u'I'): u'ı',ord(u'İ'): u'i',ord(u'J'): u'j',ord(u'K'): u'k',ord(u'L'): u'l',ord(u'M'): u'm',ord(u'N'): u'n',ord(u'O'): u'o',ord(u'Ö'): u'ö',ord(u'P'): u'p',ord(u'R'): u'r',ord(u'S'): u's',ord(u'Ş'): u'ş',ord(u'T'): u't',ord(u'U'): u'u',ord(u'Ü'): u'ü',ord(u'V'): u'v',ord(u'Y'): u'y',ord(u'Z'): u'z'}

                content = content.translate(lowerMap)

        return [

        utils.to_unicode(token) for token in utils.tokenize(content, lower=False, errors='ignore')

        if token_min_len <= len(token) <= token_max_len and not token.startswith('_')

        ]


if __name__ == '__main__':

        if len(sys.argv) < 3:
```

```
        print("iki parametre giriniz, ilki wikipedia dump, ikincisi xml icerikli dosya olacaktir.")
        print("cmd: python wikipreprocesi.py trwiki-20190320-pages-articles.xml.bz2 wiki_tr.txt")
        sys.exit()

    logging.basicConfig(level=logging.INFO,
        format='%(asctime)s %(levelname)s %(message)s')
    inputFile = sys.argv[1]
    outputFile = sys.argv[2]
    wiki = WikiCorpus(inputFile, lemmatize=False,tokenizer_func = tokenize_tr)
    logging.info("Wikipedia dump is opened.")
    output = open(outputFile,"w",encoding="utf-8")
    logging.info("Output file is created.")
    i = 0
    for text in wiki.get_texts():
        output.write(" ".join(text)+"\n")
        i+=1
        if (i % 10000 == 0):
            logging.info("Saved " +str(i) + " articles.")
    output.close()
```

Ek-2 Gensim kütüphanesi eğitimi için python kod örneği

```
#dosya adı word2vec.py
from __future__ import print_function
import logging
import sys
import multiprocessing
from gensim.models import Word2Vec
from gensim.models.word2vec import LineSentence
if __name__ == '__main__':
    if len(sys.argv) < 3:
        print("iki parametre giriniz, ilki metin kutuphane, ikinci model uzantili data")
        print("cmd: python word2vec.py derlem_tr.model")
        sys.exit()
    inputFile = sys.argv[1]
    outputFile = sys.argv[2]
    logging.basicConfig(level=logging.INFO,
        format='%(asctime)s %(levelname)s %(message)s')
    model = Word2Vec(LineSentence(inputFile), size=300, window=10, min_count=10, workers=multiprocessing.cpu_count())
    model.wv.save_word2vec_format(outputFile, binary=True)
```

Ek-3 Derlemden elde edilen örnek sonuçlar

Karabiber

[('kimyon', 0.8390986919403076), ('maydanoz', 0.8281662464141846), ('tarçın', 0.8228152990341187), ('kişniş', 0.8185569047927856), ('sarımsak', 0.8059334754943848)]

sinema

[('tiyatro', 0.5881890058517456), ('film', 0.5763677358627319), ('yeşilçam', 0.5630802512168884), ('sinemanın', 0.5332459211349487), ('başrol', 0.5142776370048523)]

başlatıyor

[('başlatıyoruz', 0.7328254580497742), ('başlatacağız', 0.7295175790786743), ('başlatacak', 0.7215145826339722), ('başlatmıştır', 0.7202731370925903), ('başlatmıştı', 0.7025669813156128)]

yıpranmış

[('eskimiş', 0.6591467261314392), ('bakımsız', 0.5125147700309753), ('bozulmuş', 0.5091325044631958), ('yanmış', 0.48716145753860474), ('zayıflamış', 0.4822291433811188)]

katlanır

[('katlanıp', 0.6019555330276489), ('yerleştirilir', 0.5933854579925537), ('sarılır', 0.5808990001678467), ('gerilir', 0.5586707592010498), ('takılır', 0.5494933128356934)]

kristalleri

[('tanecikleri', 0.7623960971832275), ('kristaller', 0.7342382073402405), ('çekirdekleri', 0.7289485931396484), ('erimiş', 0.7282359600067139), ('kristallerinin', 0.7214767932891846)]

mustafakemalpaşa

[('pamukova', 0.6977615356445312), ('saruhanlı', 0.6784433126449585), ('altınova', 0.6773111820220947), ('taraklı', 0.6718277931213379), ('çumra', 0.6698592901229858)]

başkanı

[('başkanlarından', 0.7590184211730957), ('başkanları', 0.7044947147369385), ('sekreteri', 0.6891998052597046), ('başkan', 0.6520320177078247), ('başkanlığı', 0.6375783085823059)]

başkan

[('sekreter', 0.7060188055038452), ('başkanı', 0.6520318984985352), ('müşteşar', 0.6212227940559387), ('başkanımız', 0.6177798509597778), ('dekan', 0.615256667137146)]

kaynak

[('kaynağın', 0.47507625818252563), ('kaynakla', 0.4629722237586975), ('kaynakların', 0.4586462378501892), ('kaynaklar', 0.44447892904281616), ('finansman', 0.4277971386909485)]

var

[('vardır', 0.7113890647888184), ('vardı', 0.7031223773956299), ('var…', 0.6211414337158203), ('mevcuttur', 0.5909255743026733), ('bulunuyor', 0.5816571116447449)]

yok

[('yoktu', 0.6466321349143982), ('yoktur', 0.6424567699432373), ('kalmamıştır', 0.6147197484970093), ('kalmadı', 0.6111384630203247), ('var', 0.5809080600738525)]

eski

[('şimdiki', 0.4224407374858856), ('efsane', 0.394623339176178), ('dönemin', 0.39252156019210815), ('tanınmış', 0.3922486901283264), ('sümer', 0.3890211582183838)]

eskimiş

[('yıpranmış', 0.659146785736084), ('bakımsız', 0.4674217402935028), ('demode', 0.46580252051353455), ('kullanılmaz', 0.4655357003211975), ('kullanışsız', 0.46417006850242615)]

eskidi

[('eleştirildik', 0.4862864017486572), ('üretilmiyor', 0.4722179174423218), ('yanıyordu', 0.46512147784233093), ('konuşulmuyor', 0.46133267879486084), ('güzelmiş', 0.46007469296455383)]

koca

[('dayı', 0.5729821920394897), ('kocanın', 0.5621395707130432), ('kocayı', 0.5405229330062866), ('amca', 0.5329803824424744), ('hancı', 0.529760479927063)]

kocakarı

[('kudurmuş', 0.4974026083946228), ('betimlenirler', 0.49034583568573), ('aksaçlı', 0.4785972237586975), ('mıhlası', 0.4501592516899109), ('sabahlık', 0.44864386320114136)]

zarar

[('zararlar', 0.7208236455917358), ('zararı', 0.6926591396331787), ('hasar', 0.6447446346282959), ('zararları', 0.5964639186859131), ('sebebiyet', 0.5760033130645752)]

hasar

[('hasarlar', 0.7399342060089111), ('hasarı', 0.6677336692810059), ('zarar', 0.6447446346282959), ('hasarlı', 0.6333843469619751), ('hasarın', 0.6019700169563293)]

kar

[('yağan', 0.6003866791725159), ('yağmur', 0.5640805959701538), ('yağmurun', 0.5627613663673401), ('karların', 0.5440748929977417), ('yağışı', 0.5394572615623474)]

yağmur

[('yağmurlar', 0.7179421782493591), ('yağmurun', 0.689769446849823), ('yağan', 0.6873151063919067), ('yağış', 0.6318959593772888), ('yağışlar', 0.6259993314743042)]

dolu

[('doludur', 0.6267504692077637), ('doluydu', 0.5715057849884033), ('kasvetli', 0.4633152484893799), ('doldurulmuş', 0.46156179904937744), ('hüzünlü', 0.45984241366386414)]

boş

[('bomboş', 0.539141058921814), ('boştur', 0.45386600494384766), ('boşaltılmış', 0.4483604431152344), ('boşalmış', 0.4431716799736023), ('yığılıp', 0.4409559965133667)]

mevsim

[('normallerinin', 0.6397776007652283), ('mevsimde', 0.6001346111297607), ('sıcaklıkların', 0.5972020626068115), ('sıcaklıklar', 0.5866837501525879), ('normalleri', 0.581498384475708)]

yeni

[('yepyeni', 0.5931414365768433), ('yeniden', 0.5171766877174377), ('yenilenen', 0.4401439428329468), ('mevcut', 0.43946534395217896), ('alternatif', 0.39903923869132996)]

şekil

[('biçim', 0.6507545709609985), ('şekiller', 0.5942325592041016), ('şeklini', 0.5771219730377197), ('geometrik', 0.5619038343429565), ('şekli', 0.538101077079773)]

şekilde

[('biçimde', 0.8825342059135437), ('şeklide', 0.6679823398590088), ('sekilde', 0.602854311466217), ('yaklaşımla', 0.5270988941192627), ('dille', 0.5095697045326233)]

dünya

[('avrupa', 0.5590965151786804), ('ülke', 0.5222835540771484), ('dünyanın', 0.48549893498420715), ('dünyadaki', 0.45619961619377136), ('küresel', 0.4386095404624939)]

konuş

[('kocan', 0.6050739288330078), ('oyna', 0.5709276795387268), ('otur', 0.5502496361732483), ('konuşun', 0.5495389699935913), ('düşün', 0.5462361574172974)]

konuştu

[('konuşmuştu', 0.753434956073761), ('dedi', 0.6648722290992737), ('özetledi', 0.6480600833892822), ('yanıtladı', 0.6420427560806274), ('gerekçelendirdi', 0.6362746357917786)]

konuşmuş

[('konuşuyorduk', 0.5890020728111267), ('konuştum', 0.5771540403366089), ('konuşuyorlar', 0.5767780542373657), ('konuşurken', 0.5679203271865845), ('konuşuyordu', 0.5628929138183594)]

konuşur

[('konuşabilir', 0.7023253440856934), ('konuşmaktadır', 0.6770297884941101), ('konuşuyordu', 0.6620770692825317), ('konuşurken', 0.6598633527755737), ('konuşurlar', 0.6309366226196289)]

konuşma

[('konuşmalar', 0.7019051909446716), ('sunum', 0.6748086214065552), ('konuşması', 0.6289259195327759), ('konuşmayı', 0.6250210404396057), ('görüşme', 0.5803922414779663)]

konuşmak

[('görüşmek', 0.6839953064918518), ('tartışmak', 0.682774007320404), ('buluşmak', 0.6592835187911987), ('barışmak', 0.6544157266616821), ('vedalaşmak', 0.6485488414764404)]

takım

[('takımlar', 0.7178346514701843), ('takımın', 0.6820371150970459), ('takımı', 0.6008084416389465), ('takımda', 0.5772507190704346), ('takımımız', 0.5728918313980103)]

takımlar

[('takımların', 0.727157711982727), ('takım', 0.7178347110748291), ('takımları', 0.6593475937843323), ('maçlar', 0.6073297262191772), ('takımlarla', 0.6061135530471802)]

bağ

[('bağının', 0.564887285232544), ('bağın', 0.5609394907951355), ('bağı', 0.5599401593208313), ('bağların', 0.5229852795600891), ('kovalent', 0.502244234085083)]

bağlı

[('bağlıdır', 0.7237387895584106), ('bağlanmıştır', 0.6306580901145935), ('baglı', 0.6252346038818359), ('bağlıydı', 0.6160918474197388), ('bağlandı', 0.6151700019836426)]

üzüm

[('şaraplık', 0.7475355863571167), ('incir', 0.7322126626968384), ('narenciye', 0.7317790985107422), ('dut', 0.7157198190689087), ('üzümü', 0.7150685787200928)]

film

[('filmi', 0.6879197359085083), ('filminin', 0.6510778069496155), ('filmin', 0.6393141746520996), ('filmlerinin', 0.6042102575302124), ('filmlerin', 0.5981171727180481)]

tiyatro

[('tiyatroda', 0.6815170049667358), ('kabare', 0.6389065384864807), ('tiyatronun', 0.6303055286407471), ('tiyatrosu', 0.6017594337463379), ('tiyatrosunda', 0.5957787036895752)]

nisan

[('haziran', 0.9484496712684631), ('kasım', 0.9278901815414429), ('temmuz', 0.9195787310600281), ('mart', 0.9186493158340454), ('ekim', 0.9088301658630371)]

kısa

[('uzun', 0.7266738414764404), ('uzunca', 0.6068498492240906), ('uzatımı', 0.5024981498718262), ('kisa', 0.49001753330230713), ('uzatımının', 0.4868367910385132)]

oyun

[('oyunun', 0.7255239486694336), ('oyunu', 0.7138859033584595), ('oyunlar', 0.6707362532615662), ('oynanış', 0.6414002180099487), ('oyununun', 0.6327226161956787)]

oyuncu

[('futbolcu', 0.7434569597244263), ('oyuncular', 0.6801397204399109), ('oyuncudur', 0.6281636953353882), ('oyuncunun', 0.5983906984329224), ('oyuncusu', 0.5967354774475098)]

alman

[('bavyera', 0.6459997892379761), ('avusturyalı', 0.5972336530685425), ('ingiliz', 0.5927706956863403), ('prusya', 0.5904443264007568), ('fransız', 0.5870177745819092)]

almanya

[('avusturya', 0.6440938115119934), ('fransa', 0.6200663447380066), ('hollanda', 0.6104475259780884), ('polonya', 0.5885715484619141), ('belçika', 0.5808234214782715)]

almak

[('alabilmek', 0.8280186653137207), ('almamak', 0.7632754445075989), ('aldırmak', 0.7054969072341919), ('alıp', 0.7015649080276489), ('alabilmesi', 0.7012680768966675)]

yapmak

[('yapabilmek', 0.769079864025116), ('yapmamak', 0.735366940498352), ('yaptırmak', 0.7179299592971802), ('gerçekleştirmek', 0.681445300579071), ('yapıp', 0.6639840602874756)]

yazmak

[('okumak', 0.7006826400756836), ('dinlemek', 0.6826525330543518), ('yayınlamak', 0.6749341487884521), ('hatırlamak', 0.6522912979125977), ('saklamak', 0.6516871452331543)]

varmak

[('varabilmek', 0.7453747987747192), ('ulaşmak', 0.7280110716819763), ('ulaşabilmek', 0.7145501375198364), ('ulaştırmak', 0.7028128504753113), ('dayanmak', 0.6959953308105469)]

varabilmek

[('ulaşabilmek', 0.7908282279968262), ('ulaşabilmesi', 0.7628706693649292), ('varmak', 0.7453747987747192), ('ulaşılabilmesi', 0.7139999866485596), ('ulaşmak', 0.6976339817047119)]

ressam

[('ressamı', 0.8050781488418579), ('heykeltıraş', 0.731287956237793), ('portre', 0.717700719833374), ('ressamın', 0.7042849063873291), ('izlenimci', 0.7024046778678894)]

resim

[('suluboya', 0.5450290441513062), ('çizim', 0.5336084365844727), ('portre', 0.5332532525062561), ('portreler', 0.5105584859848022), ('karakalem', 0.5073819160461426)]

görsel

[('işitsel', 0.6767352819442749), ('sanatsal', 0.5635128617286682), ('interaktif', 0.5609725117683411), ('sözsüz', 0.5243401527404785), ('multimedya', 0.5240640044212341)]

görmek

[('görebilmek', 0.6552025079727173), ('hissetmek', 0.6485501527786255), ('hatırlamak', 0.644001841545105), ('seyretmek', 0.6418559551239014), ('unutmak', 0.6377195119857788)]

yazı

[('yazılar', 0.599353551864624), ('yazısı', 0.588475227355957), ('yazıyı', 0.5604897737503052), ('yazıların', 0.5456936359405518), ('mektup', 0.5378454923629761)]

yazılı

[('basılı', 0.42433029413223267), ('pankartları', 0.42385923862457275), ('afiş', 0.4209150969982147), ('imzalı', 0.41174066066741943), ('pankartı', 0.39210444688796997)]

yazmak

[('okumak', 0.7006826400756836), ('dinlemek', 0.6826525330543518), ('yayınlamak', 0.6749341487884521), ('hatırlamak', 0.6522912979125977), ('saklamak', 0.6516871452331543)]

müzik

[('caz', 0.6106172800064087), ('müziği', 0.5863064527511597), ('muzik', 0.5790199041366577), ('müzikler', 0.5719791054725647), ('beste', 0.5659106373786926)]

keman

[('piyano', 0.8477121591567993), ('viyola', 0.8367656469345093), ('çello', 0.826356053352356), ('viyolonsel', 0.8196173906326294), ('klarnet', 0.8041539192199707)]

davul

[('davullar', 0.7831085920333862), ('perküsyon', 0.7612545490264893), ('gitar', 0.7218811511993408), ('vurmalı', 0.7178071737289429), ('bateri', 0.7169550061225891)]

perküsyon

[('saksofon', 0.7935644388198853), ('saksafon', 0.7722662091255188), ('vurmalılar', 0.7687438130378723), ('gitarlar', 0.7673279047012329), ('trompet', 0.7665313482284546)]

kale

[('kalenin', 0.741398811340332), ('kalesinin', 0.6603646278381348), ('surlar', 0.6351524591445923), ('surların', 0.6242281198501587), ('kalesi', 0.6048299074172974)]

kaleci

[('kalecisi', 0.7350190281867981), ('kaleciler', 0.6444498300552368), ('futbolcu', 0.6184730529785156), ('stoperi', 0.5807757377624512), ('kalecimiz', 0.5754915475845337)]

korner

[('korneri', 0.7212183475494385), ('penaltıyı', 0.7175613641738892), ('atışını', 0.6917427778244019), ('frikik', 0.688570499420166), ('kornerde', 0.6743531227111816)]

mal

[('malların', 0.5057497024536133), ('taşınmaz', 0.49777328968048096), ('malın', 0.4976872205734253), ('malvarlığı', 0.48216158151626587), ('mallar', 0.48018568754196167)]

hava

[('havanın', 0.5273669958114624), ('havadan', 0.48980581760406494), ('basınç', 0.4791474640369415 3), ('radar', 0.4752149283885956), ('uçakların', 0.4751928150653839)]

deniz

[('denizlerin', 0.5518547892570496), ('ponente', 0.5150766372680664), ('denizlerde', 0.5083355903625488), ('denizler', 0.49895209074020386), ('denizin', 0.498821496963501)]

kara

[('deniz', 0.4860230088233948), ('havlarken', 0.44418495893478394), ('kukal', 0.44167613983154297), ('ficedula', 0.4330074191093445), ('panter', 0.4260662794113159)]

siyah

[('lacivert', 0.6622433662414551), ('kırmızı', 0.620586633682251), ('yeşil', 0.6153668761253357), ('turuncu', 0.5931656360626221), ('yeşi', 0.5919057726860046)]

mavi

[('turuncu', 0.6666314601898193), ('mor', 0.63327956199646), ('kırmızı', 0.606213390827179), ('lacivert', 0.5987485647201538), ('mavisi', 0.5976587533950806)]

kavun

[('karpuz', 0.7959728240966797), ('salatalık', 0.7869861125946045), ('ıspanak', 0.7790176868438721), ('armut', 0.7772818803787231), ('marul', 0.7767424583435059)]

kavuniçi

[('kırmızıdır', 0.6563795208930969), ('grimsi', 0.6556352972984314), ('gömlekli', 0.6542229652404785), ('renktedir', 0.6530982851982117), ('yanaklı', 0.6521332263946533)]

doktor

[('hemşire', 0.6338780522346497), ('doktorun', 0.607218861579895), ('doktorlar', 0.5696977972984314), ('nardole', 0.5590835809707642), ('hemşirenin', 0.552852213382721)]

sağlık

[('hastane', 0.5316234827041626), ('saglık', 0.5312386751174927), ('poliklinik', 0.519740641117096), ('ağuçan', 0.4980170726776123), ('kamu', 0.4803188443183899)]

hasta

[('hastalar', 0.7149340510368347), ('hastanın', 0.6928601264953613), ('hastaların', 0.6840862035751343), ('hastaları', 0.6594066619873047), ('yatalak', 0.651007890701294)]

veba

[('salgını', 0.8496317863464355), ('salgınında', 0.7769126892089844), ('salgın', 0.7352733612060547), ('salgınları', 0.7019565105438232), ('kolera', 0.6784012913703918)]

nezle

[('öksürük', 0.7737613320350647), ('ishal', 0.772787868976593), ('bronşit', 0.7555110454559326), ('dermatit', 0.7530518174171448), ('kaşıntı', 0.7474228143692017)]

kanser

[('kanseri', 0.784548282623291), ('alzheimer', 0.7401527762413025), ('lösemi', 0.7313854694366455), ('tüberküloz', 0.7252051830291748), ('akciğer', 0.7238896489143372)]

akciğer

[('karaciğer', 0.8069057464599609), ('pankreas', 0.7967849373817444), ('prostat', 0.7877275943756104), ('böbrek', 0.7801327705383301), ('mesane', 0.7766883969306946)]

kalp

[('böbrek', 0.654578447341919), ('karaciğer', 0.5995526909828186), ('akciğer', 0.5783783793449402), ('beyin', 0.5761678218841553), ('koroner', 0.5631855130195618)]

kedi

[('kediler', 0.7377564907073975), ('kedinin', 0.7025551199913025), ('köpek', 0.7010214924812317), ('yavrusu', 0.6795971393585205), ('fare', 0.6702134609222412)]

inek

[('keçi', 0.7531143426895142), ('eşek', 0.7311121225357056), ('ineği', 0.7111126780509949), ('etinden', 0.6830472350120544), ('tavuk', 0.6798238158226013)]

tavuk

[('etinden', 0.7110422849655151), ('kızarmış', 0.6923484802246094), ('etinin', 0.6870529651641846), ('etleri', 0.6849868297576904), ('yumurtası', 0.6828206181526184)]

kartal

[('halkalı', 0.5103697180747986), ('bağcılar', 0.49922096729278564), ('aslan', 0.49903541803359985), ('üsküdar', 0.48558467626571655), ('tuzla', 0.4818558692932129)]

aslan

[('arslan', 0.6251629590988159), ('kaplan', 0.5893155336380005), ('çoban', 0.5740116238594055), ('altuntaş', 0.5574602484703064), ('özbay', 0.5569024682044983)]

kanarya

[('balear', 0.6281352043151855), ('faroe', 0.5688276290893555), ('mayorka', 0.5672847032546997), ('minorka', 0.5415995121002197), ('komor', 0.5389535427093506)]

serçe

[('kırlangıç', 0.6592361927032471), ('turna', 0.6371328234672546), ('tilki', 0.6363838911056519), ('bıldırcın', 0.6213533282279968), ('sansar', 0.6181567907333374)]

fenerbahçe

[('galatasaray', 0.7052602171897888), ('göztepe', 0.6168362498283386), ('beşiktaş', 0.61566162109375), ('trabzonspor', 0.5802591443061829), ('tff', 0.5710484981536865)]

galatasaray

[('beşiktaş', 0.7433558702468872), ('fenerbahçe', 0.7052602171897888), ('trabzonspor', 0.6206802725791931), ('bursaspor', 0.5907373428344727), ('eskişehirspor', 0.5708142518997192)]

beşiktaş

[('galatasaray', 0.7433558702468872), ('fenerbahçe', 0.61566162109375), ('trabzonspor', 0.5636096000671387), ('bursaspor', 0.5401079654693604), ('kalamış', 0.5292030572891235)]

üsküdar

[('beykoz', 0.6984947919845581), ('eyüpsultan', 0.6926299333572388), ('kanlıca', 0.6454073190689087), ('ayazağa', 0.6429719924926758), ('bağcılar', 0.6403182744979858)]

yıllık

[('senelik', 0.7622642517089844), ('aylık', 0.6603463888168335), ('yılık', 0.6361757516860962), ('günlük', 0.5396149158477783), ('haftalık', 0.5319913029670715)]

yıl

[('sene', 0.8275400400161743), ('ay', 0.8092975616455078), ('gün', 0.7108216285705566), ('hafta', 0.6874784827232361), ('sezon', 0.5636488199234009)]

para

[('paralar', 0.6712222695350647), ('paranın', 0.6475979089736938), ('parayı', 0.6372689008712769), ('paraları', 0.6282253265380859), ('paraların', 0.5913631319999695)]

banka

[('bankanın', 0.7625132203102112), ('mevduat', 0.647213339805603), ('bankaların', 0.6290206909179688), ('bankalar', 0.6250463128089905), ('kredi', 0.622198760509491)]

bankacılık

[('finans', 0.7163158059120178), ('sigortacılık', 0.6225701570510864), ('banka', 0.5964824557304382), ('bankaların', 0.5871492028236389), ('mevduat', 0.5647120475769043)]

sigorta

[('kasko', 0.6695733070373535), ('sigortası', 0.6381502151489258), ('poliçe', 0.6331453919410706), ('sigortanın', 0.6206455826759338), ('sigortaları', 0.5842118263244629)]

sigortacılık

[('bankacılık', 0.6225701570510864), ('sigortacılığı', 0.5826493501663208), ('finans', 0.5777316093444824), ('işletmecilik', 0.5507126450538635), ('perakendecilik', 0.548945426940918)]

finans

[('bankacılık', 0.7163158059120178), ('sigortacılık', 0.5777316093444824), ('finansman', 0.5638695359230042), ('finansın', 0.5412291884422302), ('finansal', 0.5187184810638428)]

ekonomi

[('maliye', 0.5881654620170593), ('iktisat', 0.5735281705856323), ('makroekonomi', 0.5181373357772827), ('tarım', 0.5045198798179626), ('finans', 0.5032663941383362)]

ekonomik

[('finansal', 0.6130458116531372), ('makroekonomik', 0.6085808873176575), ('mali', 0.5988625884056091), ('sosyoekonomik', 0.5865596532821655), ('politik', 0.5765371322631836)]

tarım

[('hayvancılık', 0.6708149313926697), ('tarımsal', 0.6632221937179565), ('tarim', 0.6362239122390747), ('tarıma', 0.6282455921173096), ('bağcılık', 0.6034808158874512)]

arpa

[('buğday', 0.7755395174026489), ('ayçiçeği', 0.7619951963424683), ('nohut', 0.7537549734115601), ('fiğ', 0.7374528646469116), ('yulaf', 0.704293966293335)]

dağ

[('dağlar', 0.6768279671669006), ('dağın', 0.6734789609909058), ('dağlarının', 0.6721676588058472), ('dağının', 0.6705952882766724), ('dağların', 0.6618483066558838)]

ova

[('ovalar', 0.6555230021476746), ('düzlükler', 0.6362810134887695), ('vadileri', 0.6319870352745056), ('düzlük', 0.6308502554893494), ('ovanın', 0.6278839111328125)]

herhangi

[('hiçbir', 0.6637300252914429), ('başkaca', 0.61786949634552), ('başka', 0.5619020462036133), ('kesinlikle', 0.5479609966278076), ('hiç', 0.5409584641456604)]

köy

[('köyün', 0.7329426407814026), ('köyünün', 0.6695514917373657), ('mahalle', 0.6574668884277344), ('köyümüz', 0.6559015512466431), ('yöre', 0.6439163684844971)]

köylü

[('köylüler', 0.5378906726837158), ('köylünün', 0.5153449177742004), ('çiftçi', 0.4950219690799713), ('köylüleri', 0.4948466122150421), ('köylüsü', 0.4791218042373657)]

coğrafya

[('görenekleri', 0.5648725032806396), ('yemekleri', 0.5523366928100586), ('gelenek', 0.5258656740188599), ('tarhana', 0.4734100103378296), ('çorbası', 0.47330302000045776)]

tarih

[('tarihi', 0.6450934410095215), ('tarihimiz', 0.502015233039856), ('tarihler', 0.4556664228439331), ('tarihini', 0.4405662417411804), ('târih', 0.42663025856018066)]

matematik

[('geometri', 0.7490521669387817), ('matematiğin', 0.6688320636749268), ('fizik', 0.6672666072845459), ('matematiği', 0.6651403903961182), ('matematikte', 0.6339671611785889)]

geometri

[('trigonometri', 0.801455020904541), ('matematik', 0.7490521669387817), ('öklid', 0.7437362670898438), ('kalkülüs', 0.7380688786506653), ('geometrisi', 0.7346000075340271)]

adet

[('adedi', 0.590612530708313), ('adeti', 0.5861308574676514), ('adedinin', 0.551019549369812), ('tane', 0.5203489661216736), ('tonluk', 0.51872718334198)]

tane

[('tanesini', 0.5548651814460754), ('tanede', 0.5310530662536621), ('kere', 0.5298384428024292), ('tanesi', 0.5251513719558716), ('adet', 0.5203489065170288)]

eder

[('ederler', 0.9012336730957031), ('edebilir', 0.8398336172103882), ('etmez', 0.8275231122970581), ('ettirir', 0.8267771005630493), ('etmesidir', 0.8220250010490417)]

edebilir

[('edebiliyor', 0.8915978074073792), ('edebilirler', 0.8681643009185791), ('edebilirsiniz', 0.8667546510696411), ('edebilmektedir', 0.8593286871910095), ('edebilecek', 0.8522103428840637)]

antalya

[('alanya', 0.7901171445846558), ('ısparta', 0.6958643198013306), ('muğla', 0.6934857368469238), ('adana', 0.6815516352653503), ('gazipaşa', 0.6634168028831482)]

ankara

[('istanbul', 0.6938591003417969), ('konya', 0.6883273124694824), ('adana', 0.6718769073486328), ('bursa', 0.670285701751709), ('antalya', 0.6292842030525208)]

ankaralı

[('adanalı', 0.5799195766448975), ('bursalı', 0.5540145039558411), ('kayserili', 0.5522699952125549), ('vanlı', 0.5458414554595947), ('konyalı', 0.5397796630859375)]

ankaradan

[('misafirler', 0.4716934561729431), ('misafirleri', 0.45390474796295166), ('misafirlerin', 0.4216572344303131), ('tatilciler', 0.4167283773422241), ('misafirlerimiz', 0.4159708023071289)]

madrid

[('betis', 0.7413105964660645), ('madrıd', 0.7117573022842407), ('zaragoza', 0.6670569181442261), ('sociedad', 0.6641050577163696), ('madridli', 0.644468367099762)]

paris

[('ecole', 0.5595484972000122), ('montparnasse', 0.5366639494895935), ('strazburg', 0.5360931158065796), ('amiens', 0.5309586524963379), ('fontainebleau', 0.5290527939796448)]

londra

[('edinburgh', 0.6116883754730225), ('croydon', 0.5986671447753906), ('glasgow', 0.5665401220321655), ('london', 0.5651886463165283), ('birmingham', 0.5640472769737244)]

berlin

[('dresden', 0.7425036430358887), ('leipzig', 0.6909198760986328), ('köln', 0.6666303277015686), ('essen', 0.6531990170478821), ('heidelberg', 0.6368750929832458)]

sanayi

[('sanayii', 0.6576613187789917), ('endüstri', 0.6316375732421875), ('sanayinin', 0.5922573804855347), ('sanayisi', 0.5582289099693298), ('imalat', 0.5357668399810791)]

palamut

[('hamsi', 0.8741074204444885), ('istavrit', 0.8735874891281128), ('lüfer', 0.8543953895568848), ('hamsinin', 0.8227134943008423), ('çinekop', 0.8158628940582275)]

mevcut

[('varolan', 0.5468926429748535), ('sistemdeki', 0.49357926845550537), ('belirlenmiş', 0.4925376772880554), ('işlevsel', 0.486431774772644), ('uygulanabilir', 0.4830266237258911)]

salı

[('perşembe', 0.9319789409637451), ('çarşamba', 0.9232860803604126), ('pazartesi', 0.9209227561950684), ('cumartesi', 0.8854666352272034), ('cuma', 0.8602866530418396)]

pazar

[('perşembe', 0.7587610483169556), ('cumartesi', 0.7362455129623413), ('çarşamba', 0.7355575561523438), ('salı', 0.7256036996841431), ('pazartesi', 0.7149773836135864)]

pazarlık

[('pazarlıklar', 0.6575157642364502), ('pazarlığı', 0.6419224739074707), ('görüşme', 0.6414493918418884), ('görüşmeler', 0.6380214691162109), ('müzakere', 0.6244779825210571)]

eşarp

[('şal', 0.7524909973144531), ('ceket', 0.7337703704833984), ('bluz', 0.7315865755081177), ('çorap', 0.7270676493644714), ('pantolon', 0.719951331615448)]

düzleme

[('düzlemine', 0.7228225469589233), ('eksene', 0.7041047811508179), ('eksenine', 0.6926788687705994), ('düzleminde', 0.690942645072937), ('düzlemde', 0.6867058277130127)]

düzlemi

[('düzlem', 0.7837026119232178), ('kutupsal', 0.7532223463058472), ('ekseni', 0.7298905849456787), ('eksenleri', 0.7241221070289612), ('tanjant', 0.7183334231376648)]

buzullar

[('buzul', 0.7907873392105103), ('buzulların', 0.7871832251548767), ('buzulları', 0.7608705759048462), ('buzulun', 0.726142168045044), ('vadiler', 0.717514157295227)]

solladı

[('aşıyor', 0.6382592916488647), ('kurtaramadı', 0.6279990673065186), ('korkutuyor', 0.623784065246582), ('aşarken', 0.6098227500915527), ('endişelendirmiyor', 0.606666088104248)]

batı

[('doğu', 0.7343217730522156), ('güneybatı', 0.6237481236457825), ('kuzeybatı', 0.6216622591018677), ('kuzeydoğu', 0.6103015542030334), ('güneydoğu', 0.557834267616272)]

meyan

[('bitkisinin', 0.7643072605133057), ('çöven', 0.7546170949935913), ('anason', 0.7473166584968567), ('zerdeçal', 0.744636058807373), ('rezene', 0.7432877421379089)]

hüznü

[('hüzün', 0.7368108034133911), ('hüznünü', 0.7001739740371704), ('acıyla', 0.6542378664016724), ('mutluluğu', 0.6450059413909912), ('keder', 0.6370415687561035)]

kasadan

[('dükkandan', 0.6948496103286743), ('kasadaki', 0.6522074937820435), ('kuyumcudan', 0.6500157117843628), ('marketten', 0.6394171714782715), ('işyerinden', 0.6236411333084106)]

kiraladık

[('kiraladı', 0.7408924102783203), ('kiralayan', 0.6611113548278809), ('kiralayarak', 0.6507939696311951), ('sattık', 0.6365664005279541), ('kiralayıp', 0.6083949208259583)]

kalmadı

[('kalmıyor', 0.7657291889190674), ('kalmamıştır', 0.7500746250152588), ('kalmamıştı', 0.748589277267456), ('kalmayacak', 0.7411359548568726), ('kalmaz', 0.7169051170349121)]

KAYNAKÇA

[1] A. M. Turing (1950) “Computing machinery and intelligence.”
İnternet adresi: https://www.csee.umbc.edu/courses/471/papers/turing.pdf

[2] Alan Turing “a short biography by Andrew Hodges” by Andrew Hodges
İnternet adresi: https://www.turing.org.uk/publications/dnb.html

[3] The Editors of Encyclopaedia Britannica “Turing test ARTIFICIAL INTELLIGENCE”
İnternet adresi: https://www.britannica.com/technology/Turing-test

[4] Copeland, Jack(19 June 2012)."Alan Turing: The codebreaker who saved 'millions of lives
İnternet adresi: https://www.bbc.com/news/technology-18419691

[5] C.E. Shannon A Mathematical Theory of Communication (1948)
İnternet adresi: http://math.harvard.edu/~ctm/home/text/others/shannon/entropy/entropy.pdf

[6] C.E. Shannon Prediction and Entropy of Printed English (1950)
İnternet adresi: https://www.princeton.edu/~wbialek/rome/refs/shannon_51.pdf

[7] Peter F. Brown at al (1990) “A Statistical approach to machine translation”
İnternet adresi: https://www.aclweb.org/anthology/J90-2002

[8] Mikolov, Tomas; et al (2013) “Efficient Estimation of Word Representations in Vector Space”
İnternet adresi: https://arxiv.org/abs/1301.3781

[9] Prof. Dr. Zeynep Korkmaz, Türkiye Türkçesi Grameri (Şekil Bilgisi), Türk Dil Kurumu Yayınları: 827, Ankara 2003

[10] T. Mikolov, W.T. Yih, G. Zweig. (2013) “Linguistic Regularities in Continuous Space Word Representations.”
İnternet adresi: https://www.aclweb.org/anthology/N13-1090

[11] Bağlam kelimesi Türk Dil Kurumu
İnternet adresi:
http://www.tdk.gov.tr/index.php?option=com_bts&arama=kelime&guid=TDK.GTS.5cacdc23713070.64368406

[12] T. Mikolov ve ekibi “Distributed Representations ofWords and Phrases and their Compositionality”
İnternet adresi: https://arxiv.org/pdf/1310.4546.pdf

[13] David E Rumelhart et al (1986); Learning representations by back-propagating errors. Nature, 323(6088):533–536.

[14] T. Mikolov et al; “Exploiting Similarities among Languages for Machine Translation”
İnternet adresi: https://arxiv.org/pdf/1309.4168.pdf

[15] Python ile Türkçe derlem (corpus) hazırlama

İnternet adresi: https://github.com/ahmetax/derlemtr

[16] P.F. Brown, S.A. Della Pietra, V.J. Della Pietra, R.L. Mercer, (1993) "The Mathematics of Statistical Machine Translation: Parameter Estimation"
İnternet adresi: https://www.aclweb.org/anthology/J93-2003

[17] Gensim Word2vec eğitim paketi
İnternet adresi: https://radimrehurek.com/gensim/models/word2vec.html

[18] The Georgetown-IBM experiment of 1954 Paul L. Garvin
İnternet adresi: http://www.mt-archive.info/Garvin-1967.pdf

[19] ALPAC (Automatic Language Processing Advisory Committee) komitesi makine tercümesi hakkındaki raporu. (1966)
İnternet adresi: https://www.nap.edu/read/9547/chapter/1

[20] Joseph Weizenbaum "Computatioanl Linguistics 1966"
İnternet adresi: https://web.stanford.edu/class/linguist238/p36-weizenabaum.pdf

[21] Thomas R. Gruber "Toward Principles for the Design of Ontologies Used for Knowledge Sharing"
İnternet adresi:
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.89.5775&rep=rep1&type=pdf

[22] IBM T. J. Watson Research Center "Class-Based n-gram Models of Natural Language "
İnternet adresi: https://www.aclweb.org/anthology/J92-4003

[23] Speech and Language Processing. Daniel Jurafsky & James H. Martin "N-gram Language Models"
İnternet adresi: https://web.stanford.edu/~jurafsky/slp3/3.pdf

[24] B. A. Shawar and E. Atwell, "A comparison between Alice and Elizabeth Chatbot Systems," Univ. Leeds, 2002.
İnternet adresi: http://eprints.whiterose.ac.uk/81930/1/AComparisonBetweenAliceElizabeth.pdf

[25] Yoshua Bengio, P. Simard (1994) "Learning Long-Term Dependencies with Gradient Descent is Difficult"
İnternet adresi: http://ai.dinfo.unifi.it/paolo/ps/tnn-94-gradient.pdf

[26] Y. Bengio, Y. LeCun (2007) "Scaling learning algorithms towards AI."
İnternet adresi: http://yann.lecun.com/exdb/publis/pdf/bengio-lecun-07.pdf

[27] Salah El Hihi, Y. Bengio (1996) "Hierarchical Recurrent Neural Networks for Long-Term Dependencies."
İnternet adresi: https://papers.nips.cc/paper/1102-hierarchical-recurrent-neural-networks-for-long-term-dependencies.pdf

[28] Remi Lebret,Ronan Collobert (2017) "Word Embeddings through Hellinger PCA"
İnternet adresi: https://arxiv.org/pdf/1312.5542.pdf

[29] T. Brants, A. C. Popat, P. Xu, F. J. Och, and J. Dean. (2007)"Large language models in Machine Translation"
İnternet adresi: https://www.aclweb.org/anthology/D07-1090

[30] T. Mikolov. 2007 “Language Models for Automatic Speech Recognition of Czech Lectures”
İnternet adresi:
http://www.fit.vutbr.cz/research/groups/speech/publi/2008/mikolov_eeict2008.pdf

[31] T. Mikolov, J. Kopecky´, L. Burget, O. Glembek and J. C˘ ernocky´. (2009) “Neural network based language models for higly inflective languages”
İnternet adresi:
http://www.fit.vutbr.cz/research/groups/speech/publi/2009/mikolov_ic2009_nnlm_4.pdf

[32] Peter F. Brown at al (1992) “Class-Based n-gram Models of Natural Language”
İnternet adresi: https://www.aclweb.org/anthology/J92-4003

[33] D. Hiemstra (1996) "Using statistical methods to create a bilingual dictionary"
İnternet adresi: https://djoerdhiemstra.com/wp-content/uploads/hiemstra96.pdf

[34] Sanjeev R. Kulkarni, Gilbert Harman (2011) “Statistical learning theory: a tutorial”
https://İnternet adresi: www.academia.edu/28726463/Statistical_learning_theory_a_tutorial

[35] T. Hastie, R. Tibshirani, J. Friedman, “The Elements of Statistical Learning”
Springer Series in Statistics. New York, NY, USA: Springer New York Inc. 2001.

[36] Mete Özay doktora tezi (2013) "Decision fusion for supervised, unsupervised and semi-supervised learning" Tez No 338501.

[37] Chris Potts, Ling 236/Psych 236c: Representations of meaning, “Spring 2013Distributional approaches to word meanings”
İnternet adresi: https://web.stanford.edu/class/linguist236/materials/ling236-handout-05-09-vsm.pdf

[38] John Rupert Firth
İnternet adresi: https://www.britannica.com/biography/John-R-Firth

[39] Aloise, D. & Hansen, P. (2007) “On the complexity of minimum sum-of-squares clustering.”
İnternet adresi: https://www.gerad.ca/~aloise/G-2007-50.pdf

[40] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu and P. Kuksa.(2011) Natural Language Processing (Almost) from Scratch.
İnternet adresi: http://www.jmlr.org/papers/volume12/collobert11a/collobert11a.pdf

[41] Toutanova et al. (2003) “Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network”
İnternet adresi: https://nlp.stanford.edu/pubs/tagging.pdf

[42] O. Bousquet, S. Boucheron, and G. Lugosi, “Introduction to statistical learning theory.” in Advanced Lectures on Machine Learning, ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures, ser. Lecture Notes in Computer Science, O. Bousquet, U. von Luxburg, and G. Rätsch, Eds., vol. 3176. Springer, 2003 “Feature Selection and Feature Reduction: Removing Excess Data”

İnternet adresi:
https://www.researchgate.net/publication/238718428_Advanced_Lectures_on_Machine_Learning_ML_Summer_Schools_2003_Canberra_Australia_February_2-14_2003_Tubingen_Germany_August_4-16_2003_Revised_Lectures

[43] T. Mikolov at al. Brno University of Technology 2011 "Extensions of Recurrent Neural Network Language Model"
İnternet adresi:
http://www.fit.vutbr.cz/research/groups/speech/publi/2011/mikolov_icassp2011_5528.pdf

[44] Christopher D. Manning Prabhakar Raghavan Hinrich Schütze "An Introduction to Information Retrieval" Cambridge University Press Cambridge, England Online edition (c) 2009 Cambridge UP

[45] Y. Chen, B. Perozzi, R. Al-Rfou', S. Skiena. 2013. The expressive power of word embeddings.
İnternet adresi: https://arxiv.org/pdf/1301.3226.pdf

[46] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin. (2003) "A neural probabilistic language model. Journal of" Machine Learning Research, 3:1137–1155, 2003a.
İnternet adresi: http://www.jmlr.org/papers/volume3/bengio03a/bengio03a.pdf

[47] Y. Bengio, J.S. Sen´ecal, et al. (2003) "Quick training of probabilistic neural nets by importance sampling." In AISTATS Conference, 2003b.
İnternet adresi: http://www.iro.umontreal.ca/~lisa/pointeurs/senecal_aistats2003.pdf

[48] J.B. McQueen. Some methods for classification and analysis of multivariate observations. In Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, volume 2, pages 281–297, Berkeley, CA, 1967.
İnternet adresi:
https://pdfs.semanticscholar.org/a718/b85520bea702533ca9a5954c33576fd162b0.pdf

[49] Hinrich Schütze and Jan O. Pedersen.Xerox Palo Alto Research Center 1995. "Information retrieval based on word senses."
İnternet adresi:
https://pdfs.semanticscholar.org/0430/5cc88e1b55365d9bcb5039d26ba5d4595cfd.pdf

[50] A.K. Jain, M. Narasimha Murty, Patrick Flynn. "Data clustering: A review."
İnternet adresi: http://users.eecs.northwestern.edu/~yingliu/datamining_papers/survey.pdf

[51] Tomas Mikolov, Martin Karafiat, Jan Cernocky, and Sanjeev Khudanpur. (2010) "Recurrent neural network based language model."
İnternet adresi:
https://www.fit.vutbr.cz/research/groups/speech/publi/2010/mikolov_interspeech2010_IS100722.pdf

[52] David Jurgens, Saif Mohammad, Peter Turney, and Keith Holyoak. 2012. Semeval-2012 task 2. "Measuring degrees of relational similarity. In *SEM 2012: The First Joint Conference on Lexical and Computational Semantics (SemEval 2012), pages 356–364. Association for Computational Linguistics."
İnternet adresi: http://reasoninglab.psych.ucla.edu/KH%20pdfs/Jurgens_etal.2012.pdf

[53] G.Salton, A.Wong, C.S.Yang (1975) "A vector space model for automatic indexing"

İnternet adresi:
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.446.5101&rep=rep1&type=pdf

[54] Grigori Sidorov et al. (2014) Soft Similarity and Soft Cosine Measure:Similarity of Features in Vector Space Model
İnternet adresi: http://www.scielo.org.mx/pdf/cys/v18n3/v18n3a7.pdf

[55] Edgar Altszyler, Mariano Sigman, Sidarta Ribeiro, Diego Fernández Slezak (2017) "Comparative study of LSA vs Word2vec embeddings in small corpora: a case study in dreams database"
İnternet adresi: https://arxiv.org/pdf/1610.01520.pdf

[56]Edgar Altszyler,Mariano Sigman,Diego Fernandez Slezak(2018) "Corpus specificity in LSA and word2vec: the role of out-of-domain documents"
İnternet adresi: https://www.aclweb.org/anthology/W18-3001

[57] Peter J. Rousseeuw. (1987) "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics" 20:53–65.
İnternet adresi:
https://www.researchgate.net/publication/222451107_Rousseeuw_PJ_Silhouettes_A_Graphical_Aid_to_the_Interpretation_and_Validation_of_Cluster_Analysis_Comput_Appl_Math_20_53-65

[58] Prathusha K Sarma, Yingyu Liang and William A Sethares (2018) "Domain Adapted Word Embeddings for Improved Sentiment Classification"
İnternet adresi: https://www.aclweb.org/anthology/W18-3407

[59] Sekitoshi Kanai,Yasuhiro Fujiwara,Yuki Yamanaka,Shuichi Adachi Sigsoftmax: Reanalysis of the Softmax Bottleneck
İnternet adresi: https://papers.nips.cc/paper/7312-sigsoftmax-reanalysis-of-the-softmax-bottleneck.pdf

[60] WordNet
İnternet adresi: https://wordnet.princeton.edu/

[61] Frederic Morin and Yoshua Bengio.(2005) "Hierarchical probabilistic neural network language model."
İnternet adresi: https://www.iro.umontreal.ca/~lisa/pointeurs/hierarchical-nnlm-aistats05.pdf

[62] Andriy Mnih and Geoffrey (2009)E Hinton. "A scalable hierarchical distributed language model."
İnternet adresi: https://www.cs.toronto.edu/~amnih/papers/hlbl_final.pdf

[63] Handout by Julie Zelenski with minor edits by Keith Schwarz and Marty Stepp "Huffman Encoding and Data Compression"
İnternet adresi:
http://web.stanford.edu/class/archive/cs/cs106x/cs106x.1192/resources/minibrowser2/huffman-encoding-supplement.pdf

[64] Michael U Gutmann and Aapo Hyvarinen.(2012) "Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics"
İnternet adresi: http://www.jmlr.org/papers/volume13/gutmann12a/gutmann12a.pdf

[65] Andriy Mnih and Yee Whye Teh.(2012) “A fast and simple algorithm for training neural probabilistic language models.”
İnternet adresi: https://www.cs.toronto.edu/~amnih/papers/ncelm.pdf

[66] Wikimedia dump Türkçe servisi
İnternet adresi: https://dumps.wikimedia.org/trwiki/20190320/

[67] Gensim Wikimedia dump servisi için temizleme paketi “corpora.wikicorpus”
İnternet adresi: https://radimrehurek.com/gensim/corpora/wikicorpus.html

[68] Word2Vec gibi işlemlerde kullanılmaya uygun Türkçe metin dosyaları
İnternet adresi: https://drive.google.com/drive/folders/0B_iRLUok9_qqOFozeHNFMjRHTVk