WORD2VEC KULLANARAK EŞ ANLAMLILIK TEMELİNDE ANAHTAR KELİME ÇIKARIMI

KEYWORD EXTRACTION BASED ON WORD SYNONYMS USING WORD2VEC

Iskender Ulgen OGUL
Bilgisayar Mühendisliği, İzmir Yüksek Teknoloji Enstitüsü, İzmir, Türkiye
iskenderogul@gmail.com

Caner OZCAN
Bilgisayar Mühendisliği, Karabük Üniversitesi, Karabük, Türkiye
canerozcan@karabuk.edu.tr

Ozlem HAKDAGLI
Bilgisayar Mühendisliği, Uludağ Üniversitesi, Bursa, Türkiye
ozlemhakdagli@gmail.com

***Özetçe*— Günümüzde, çevrimiçi olan bireyler ile birlikte ortaya çıkarılan veriler, üstel bir şekilde artmaktadır. Artan verilerin barındırdığı ham bilgiler, makine öğrenmesi ve derin öğrenme yöntemleri kullanılarak, anlam içeren bilgi çıktılarına dönüştürülmektedir. Bilgi çıkarımı ve sınıflandırma için yaygın olarak denetimli öğrenme yaklaşımları kullanılmaktadır. Denetimli öğrenmenin temeli, sınıflandırma algoritmalarının eğitileceği eğitim setine dayanmaktadır. Önerilen yaklaşımda, metin verilerinin daha iyi sınıflandırılabilmesi için, anahtar kelime çıkarımı geliştirilmiştir. Geliştirilen yöntem, genel olarak kullanılan kelime frekansı temelli anahtar kelime çıkarımından farklı olarak, kelimelerin birbirileri ile olan semantik anlamı gözeterek çalışan Word2Vec algoritması üzerine kurulmuştur. Yeni bir yaklaşım olan word-embedding algoritması "Word2Vec", kelime frekansı, anlam ilişkisi ve vektörlerin nihai ağırlıklarını hesaplayarak çalışmaktadır. Elde edilen anahtar kelimeler, Naïve Bayes ve Karar Ağaçları yöntemleri ile eğitilmiş ve sınıflandırma örneği ile yöntemin çalışma performansı gösterilmiştir.**

Anahtar Kelimeler — Spark, Word2Vec, Word Embedding, Anahtar Kelime Çıkarımı, Metin Madenciliği.

***Abstract*— Nowadays, the data revealed by the online individuals are increasing exponentially. The raw information that increasing data holds, transformed into meaningful outputs using machine learning and deep learning methods. Generally, supervised learning methods are used for information extraction and classification. *S*upervised learning is based on the training set that classification algorithms are trained. In the proposed approach, keyword extraction solution is proposed to classify text data more convenient. The developed solution is based on the Word2Vec algorithm, which works by taking into consideration the semantic meaning of the words unlike general approaches that based on word frequency. A new approach, word embedding algorithm named "Word2Vec", works by calculating the word weights, semantic relationship, and the final weights of vectors. The obtained keywords are trained with Naïve Bayes and Decision Trees methods and the performance of the proposed method is shown by classification example.**

Keywords — Spark, Word2Vec, Word Embedding, Keyword Extraction, Text Mining.

I. GİRİŞ

Gelişen teknoloji vasıtasıyla günlük hayatımızda ortaya çıkarılan veri, her geçen yıl katlanarak artmaktadır. Teknolojiye olan kolay erişim ve çevrim için cihazların artması ile birlikte, bilgiye erişimin kolaylaşması kadar, bireylerin anlık bilgi üretimi de hızlanmıştır. Genel veri kaynakları IoT cihazlar, günlük kayıt verileri, uzaktan algılama, bilgisayarlı görme, gerçek zamanları akış ve sosyal medya verileri olarak örneklendirilebilir.

Sosyal medya kanallarına erişim imkanlarının kolaylaşması ile birlikte, kullanım oranları her geçen gün artmıştır, dolayısı ile metin tabanlı verilerin büyük bir kısmı, insanlar tarafından sosyal medya üzerinden oluşturulmaktadır. İnsanlar günlük hayatlarındaki tecrübeleri ve gelişen olaylar hakkında edindikleri bilgileri, sosyal medya ile arkadaşları ya da anonim olarak toplumun diğer bireyleri ile paylaşmaktadırlar [1].

Veri analizindeki en temel yaklaşım, bilgi getirimi ve getirim sağlanacak verilerin analize uygun formatlarda depolanıp saklanabilmesidir. Depolanan verilerin içerdiği ham bilgi potansiyeli yadsınamayacak kadar ciddi bir öneme sahiptir. İşlenmemiş ham veriler, doğru ve hedefe yönelik yöntemlerle analiz edildiği takdirde, elde edilecek olan sonuçlar, karar alma mekanizmaları veya bir olay – duruma verilen tepkileri ölçmek için anlamlı çıktılara dönüştürülebilmektedir.

Yıllar süren araştırmalar ve geliştirilen yöntemler sonucunda, veri analizinde kullanılan en güncel yaklaşımlar, makine öğrenmesi ve derin öğrenme yöntemleridir. Makine öğrenmesi yöntemleri genel olarak, denetimli ve denetimsiz öğrenme olarak iki gruba ayrılmaktadır. Denetimsiz öğrenme yaklaşımı, veri üzerinden, herhangi bir etiket olmadan öğrenilmesi ve analiz yapılmasıdır. Denetimli öğrenmenin temeli ise, eğitim verisi olarak belirlenen etiketlenmiş veri grubundan eğitilen model esasına dayanmaktadır. Oluşturulan model, daha sonra etiketsiz veriler üzerinde, çoğunlukla sınıflandırma problemleri için kullanılabilmektedir.

Bu çalışmada, denetimli öğrenme esasına dayanan, metin verilerinin etkin bir şekilde sınıflandırılabilmesi için eğitim verisi (anahtar kelime – keyword) çıkarımına çözüm örneği sunulmuştur.

978-1-7281-1904-5/19/$31.00 ©2019 IEEE

II. İLGİLİ ÇALIŞMALAR

Metin madenciliği alanının önem kazanmasıyla beraber, anahtar kelime merkezli eğitim seti çıkarımı da önem kazanmıştır. Sınıflandırma problemleri ve sınıflandırıcı makine öğrenmesi modellerinin temeli olan eğitim verilerinin çıkarımı için birçok yöntem geliştirilmiştir. Genel yaklaşımlar kümeleme ve kelime frekansı olarak iki grupta incelenmektedir. Kümeleme yaklaşımları TextRank – LDA – Kmeans, frekans yaklaşımları ise Term Frequency (TF) ve Inverse Document Frequency (IDF) olarak ele alınabilmektedir [2].

Klasik vektör yaklaşımı olan Term Frequency – Inverse Document Frequency (TF – IDF), bir kelimenin, bir koleksiyon ya da corpus içindeki bir belge için, ne kadar önemli olduğunu yansıtması amacıyla geliştirilmiş sayısal, istatistiksel bir yöntemdir. Metin verilerinin, makine öğrenmesi algoritmaları üzerinde kullanılabilmesi için; vektör olarak adlandırdığımız matematiksel ifadelere çevrilmesi gerekmektedir. Vektör temsillerine çevrilen metin verileri, makine öğrenmesi yöntemleri kullanılarak analiz edilmektedir [3].

Kümeleme algoritmalarında ise genel yaklaşım, kelime frekansı ile vektör temsillerine çevrilen kelimelerin, birbiri ile ulan uzaklıkları baz alınarak, iki boyutlu uzayda konumlandırılmasıdır.

Zhang Q. ve ekibi gerçekleştirdikleri projede, anahtar kelime çıkarımı için vektör uzay modelini ön işleme olarak kullanmış ve çince veri seti üzerinde anahtar kelime çıkarımı gerçekleştirmişlerdir. [4]

Ping Zeng ve ekibi gerçekleştirkleri projelerinde, kümeleme tabanlı algoritmaları ve Word2Vec algoritmalarını karşılaştırmışlardır. Wikipedia verisi ile eğittikleri Word2Vec algoritmasını Hulth2003, 500N ve Sem2010 veri kümeleri ile test etmiş ve Word2Vec yaklaşımının diğerlerinden daha iyi performans verdiğini belirtmişlerdir [5].

Metin verisi sınıflandırma problemi için geliştirilen çalışmalarda, makine öğrenmesi algoritmaları, literatür araştırması doğrultusunda elde edilen anahtar kelimelerin TF – IDF yöntemleri kullanılmasıyla eğitildi ve çözüm sunuldu [6,7,8]. Geliştirilen yaklaşımlarda, metin verileri frekans temeline dayanan vektör yaklaşımı ile sınıflandırıldığı için, metnin anlamsal ilişkisi göz ardı edilmiştir. Çalışmalarda, frekans tabanlı vektör yaklaşımının kullanılması, makine öğrenmesi algoritmalarının tam potansiyelini ortaya çıkaramadığı düşünülmektedir.

III. PROBLEM TANIMI VE YAKLAŞIMLAR

Denetimli makine öğrenmesi algoritmalarının performansı, algoritmaların ya da veri seçiminin ötesinde, ağırlıklı olarak eğitim verilerine bağlıdır. Hedefe yönelik elde edilen eğitim verileri ile makine öğrenmesi algoritmalarından, istenilen en yüksek performans sağlanabilmektedir.

Metin verilerinin, anahtar kelime temelinde sınıflandırma problemleri için, eğitim verisi çıkarımı genel olarak kelime frekansı yaklaşımı ile çözülmektedir. Frekans yaklaşımında, bir kelimelinin, bağlı olduğu döküman içerisindeki önemi, görülme sıklığı ve bu sıklığın döküman içindeki frekansına bakılarak elde edilir. Gerçekleştirilen bu ön işleme ile metin verileri, makine öğrenmesi algoritmaları için anlamlı birer matematiksel ifadeye çevrilmiş olurlar.

Frekans temelli yaklaşım, kelimelerin semantik birlikteliklerini göz ardı etmekte ve çıktı olarak seyrek (sparse) vektör ifadeleri vermektedir. Eğitim verisi elde edilmek istenen bir kategori için semantik anlam gözetilerek çıkarım yapıldığında, elde edilen veri seti, anlamsal olarak çıkarım yapılan kategori ile benzer olmalıdır.

Hipotez 1: Semantik olarak benzer anlam taşıyan kelimeler, benzer sentiment değerleri almalı ve vektör uzayında yakın noktalarda bulunmalıdır.

Hipotez 2: Embedding yaklaşımının kullanıldığı vektör temsillerinin (Dense), makine öğrenmesi modellerinin eğitimi ve sınıflandırma çözümlerinde, frekans temelli (Sparse) yaklaşımlardan daha iyi performans vermesi beklenmektedir.

A. Word2Vec Algoritması

Konuşma tanıma, nesne tanıma ve görüntü analizi verileri, yapılarından dolayı yoğun vektörler ile temsil edilmektedir. Ancak doğal dil işleme ve metin analizi yaklaşımları genel olarak ayrık, atomik semboller yani sparse vektör olarak ele alınmaktadır. Seyrek vektör gösterimleri ve bireysel semboller, kelimeler arasındaki semantik ilişkileri tam anlamı ile çıkaramamaktadır.

Word2Vec algoritması, vektör uzay modelini kullanarak, kelime vektörlerini, yoğun (dense) vektör olarak temsil etmekte ve kosinüs uzaklığını kullanarak benzer anlam taşıyan kelimeleri, vektör uzayında birbirine yakın noktalara haritalandırmaktadır. Öğrenme temelli bir yaklaşım kullanan Word2Vec, verilen dökümanları tarayarak, benzersiz kelimelerden sözlük oluşturmakta ve her bir kelime için benzersiz vektör atamaktadır. Algoritma veri kümesini iteratif şekilde tarayarak, kelimelerin anlamsal yakınlıkları öğrenmekte ve sözlüğün son vektör ağırlıklarını hesaplamaktadır. Öğrenme sonucunda semantik yönden benzer kelimeler, kosinüs uzaklığı kullanılarak, vektör uzayında yakın noktalara yerleştirilmektedir.

Geliştirilen bu yaklaşım ile metin verilerini yoğun vektör olarak temsil etmek ve anlamsal yapıyı koruyarak bilgi çıkarımının yapılması mümkün olmaktadır [9].

B. Resilient Distributed Dataset / Dataframe – Spark

Çalışma kapsamında, esnek dağıtılmış veri kümesi (RDD) yapısına sahip Apache Spark mimarisi kullanılmıştır. RDD yapısı, Hadoop dosya sistemi temel alınarak geliştirilen, verilerin RAM üzerinde, hata toleranslı bir yaklaşımla depolanarak, bilgisayar kümeleri arasında paylaşımına olanak sağlayan bir yapıdır.

Spark mimarisi, rdd yaklaşımı ile, map – reduce işlemlerinde doksan kata kadar hız sağlamakta ve verilerin RAM üzerinde depolanması sayesinde, disk girdi – çıktı işlemleri için kaybedilen zamanı minimuma indirerek, kiritik bir hız kazanımı sağlamaktadır [10].

Spark mimarisi, dahili bir makine öğrenmesi kütüphanesi bulundurmakta ve veri çözümünde olduğu gibi makine öğrenmesi algoritmalarının eğitimi aşaması içinde paralelleştirme sunmaktadır.

Sunulan, anahtar kelime çıkarımı çözümünde, veri ve işlem paralelliğinden yararlanmak için, Mikolov'un Word2Vec orijinal C dili algoritmasının, Spark üzerinde uyarlanmış versiyonu kullanılmıştır.

IV. YÖNTEM TASARIMI

Tasarlanan analiz ortamında, Word2Vec algoritmasının eğitimi için, Reddit tarafından anonim bireylerce üretilmiş ve açık kaynak olarak olarak sunulan veri seti kullanılmıştır. Reddit veri seti, farklı konular altında girilen kullanıcı yorumlarını barındırmakta ve JSON objesi olarak sunulmaktadır. Word2Vec algoritması, Reddit üzerinden elde edilen 10.1 Gb boyutundaki yorum veri kümesi ile eğitilmiştir.

Frekans tabanlı özellik çıkarımı ile öğrenme tabanlı özellik çıkarımının karşılaştırılabilmesi için seçlien beş ana kategori "*music, science, politics, gaming, cars*" üzerinden, TF-IDF ve Word2Vec modelleri kullanılarak eğitim setleri çıkarılmıştır. Çıkarılan setler ile Naïve Bayes ve Karar Ağaçları sınıflandırma algoritmaları eğitilerek, bahsedilen kategorilere ait toplamda 250.000 adet yorum içeren test veri seti oluşturulmuş ve sınıflandırma gerçekleştirilmiştir.

Naïve Bayes sınıflandırma alanında kendini kanıtlamış bir algoritmadır. Yoğun ve Seyrek vektörler ile uygun çalışabilmektedir. Karar Ağaçları, metin sınıflandırması alanında, seyrek vektörler ile optimal çalışamadığı için çok tercih edilmemektedir. Seçilen algoritmalar ile seyrek ve yoğun vektör farkının uygun şekilde verilmesi hedeflenmiştir.

A. Veri Ön İşleme

Veri analizi çözümlerinde, ön işleme aşaması ve veriyi iyi tanımak, sonuçları oldukça fazla etkilemektedir. Seçilen Reddit verisi toplamda 58 milyon adet yorum barındırmaktadır. Veri incelendiğinde, kullanıcı ya da moderatör tarafından silinen yorumların "*deleted*" ve "*removed*" gürültü yorumları kaldırılmıştır. Ardından her bir yorum, tokenization aşamasından geçirilerek kelime gruplarına ayrılmıştır. Devamında analiz sonuçlarını olumsuz etkileyen gürültü verisi olarak bilinen "*Stop Words*" temizlenmiştir. Son olarak Word2Vec algoritmasının performasını arttırmak için beş den az görülen seyrek kelimeler "*rare words*" temizlenmiştir. Genel NLP yaklaşımlarında "*spell correction*" kelime düzeltme ve "*lemmatization*" kök – ek ayrımı yapılması beklenilse de her bir kelimenin cümle içerisinde bir anlamı temsil etmesinden dolayı Word2Vec yaklaşımı için bu aşama gerekmemektedir.

TABLO I. VERİ VE MODEL ÖZELLİKLERİ

Veri Özellikleri		**Word2Vec Parametreleri**	
Yorum Sayısı	53,851,542	**Vektör Boyutu**	300
Temiz Yorum	50,580,952	**Minimum Sayı**	0
Kelime Sayısı	1,558,781,124	**Pencere Sayısı**	5
Stop Word Silindiğinde	867,410,081	**Paralellik**	16
Toplam Kelime	858,446,824	**Iterasyon**	8

Geliştirilen yöntemin sözde kodu ise şu şekildedir.

Input: Reddit Comments
Output: Keywords & Classification Results
  Reddit_df: Read Data to Dataframe
  Cleaned_df: Clean Data / Tokenize / Remove Rare Words
  Word2Vec: Cleaned Dataset
  Traning_set_embed: Acquire Keywords from w2v Model
  Training_set_TF : 5000 Comments for each category
  Naïve_Bayes_Model: Naïve_Bayes (ts_embed, ts_tf)
  Decision_Tree_Model: Decision_Tree (ts_embed, ts_tf)
  Foreach: NB_Model, DTree_Model, Cleaned Data
    predict= (pos prob > neg prob)? pos : neg
  return: Classification Results
End

B. Word2Vec Modeli

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{J=-k}^{j=k} log\ p(w_{t+j}|w_t \tag{1}$$

Word2Vec'in Skip-Gram yaklaşımı kullanılmıştır. Hedef, verilen eğitim verisi $w_1, w_2, w_3, \ldots\ldots, w_T$ üzerinden, maksimum log-benzerliğini, eğitim aralığı olan k üzerinden elde etmektir. Skip-Gram modelinde her kelime w, sırasıyla kelime ve bağlam olarak iki vektörle $u_w\ ve\ v_w$ ilişkilendirilir. Verilen kelime w_j üzerinden, doğru kelime w_i tahmin etme olasılığı, Softmax modeli tarafından kontrol edilmektedir.

$$p(w_i|w_j) = \frac{\exp(u_{wi}^T w_j)}{\sum_{l=1}^{V}\exp(u_{wi}^T w_j)} \tag{2}$$

Softmax formülünde V, sözlük boyutunu göstermektedir. Softmax kullanarak oluşturulacak Skip-Gram modeli, işlem yoğunluğu yüksek bir yaklaşımdır. $log\ p(w_i|w_j)$, sözlük V ile orantılı olarak artmaktadır. Word2Vec eğitimini hızlandırmak için hiyerarşik softmax kullanılmıştır. Bu da hesaplama karmaşıklığını $log\ p(w_i|w_j)$ den $O(\log(V))$ durumuna optimize etmiştir.

Vektör boyutu, Word2Vec algoritmasını eğitirken her bir kelimenin kaç uzunlukta vektör ya da boyut ile temsil edileceğini gösterir. Minimum sayı ile ifade edilen nokta, Word2Vec'in benzersiz kelime sözlüğü oluştururken, her bir kelimenin doküman içinde en az ne kadar görünmesi gerektiğini belirler. Bu işlem veri temizleme aşamasında gerçekleştirildiği için sıfır verilmiştir. Pencere aralığı, Word2Vec modeli eğitilirken, bir kelimenin kendisinden önceki ve sonraki kelimelerle olan ilişkisine bakılır. Bahsedildiği gibi Spark, model paralelliği sunmaktadır. Model işlemcilere bölündükçe, tek seferde gözteceği veri grubu da aynı oranda bölünmektedir. Paralellik – iterasyon arasındaki oran yarı yarıya olduğu durumlarda veri bütünlüğünün korunduğu gözlemlenmiştir. Her itreasyon, bölüntülerin bir önceki iterasyon esnasında örnekleyemediği verileri, yeniden dağıtır ve bölüntülerin tüm veriyi örneklemesine olanak sağlar.

C. Naïve Bayes

Word2Vec üzerinden elde edilen eğitim setinin doğruluğunun test edilmesi ve embed tabanlı temsil yaklaşımının frekans yaklaşımdan farkının ölçülmesi için sınıflandırma çalışması gerçekleştirilmiştir. Sınıflandırma için Multinomial Naïve Bayes yaklaşımı seçilmiştir.

$$p(x\,|C_k) = \frac{(\sum_i X_i)!}{\prod_i X_i!}\prod_i p_{ki}{}^{xi} \tag{3}$$

Multinomial Naïve Bayes, $(p_1 \dots \dots p_n)$ tarafından üretilen belirli olayları temsil eder. Burada p_i, i'nin görülme sıklığını temsil eder. Çoklu sınıf durumları, k ile temsil edilmektedir. Vektör $x = (x_i \dots \dots x_n)$ 'in, dokümanda görülme sıklığı x_i ile ifade edilerek, son çıktı elde edilir.

D. Karar Ağaçları

$$\sum_{i=1}^{C} f_i(1 - f_i) \tag{4}$$

f_i düğümdeki etiket i'nin frekansı, C ise etiketlerin numarasıdır. Bilgi kazancı, ana düğüm ile iki alt düğümün toplamı arasındaki farktır.

$$\begin{aligned}\mathrm{IG(D,s)} \;=\; & Impurity(D) - \frac{N_{left}}{N} Impurity\big(D_{left}\big) \\ & - \frac{N_{right}}{N} Impurity\,(D_{right})\end{aligned} \tag{5}$$

V. Deneysel Çıktılar

Elde edilen bu anahtar kelimeler ile Naïve Bayes ve Karar Ağaçları modelleri, TF – IDF ve Word2Vec yöntemleri kullanılarak eğitilmiş ve model doğrulukları, 60-40, 70-30, 80-20 ve K = 10 fold çapraz doğrulama yöntemleri ile hesaplanmış, Tablo 2'de karşılaştırmaları ile verilmiştir. Son olarak Word2Vec yönteminin çalışma performansını test etmek için belirlenen kategorilerde her biri 50 bin olmak üzere, 250 bin yorum alınarak sınıflandırma işlemi gerçekleştirilmiş ve sonuçlar Tablo 3'de verilmiştir.

TABLO II. Model Doğruluk Tablosu

	60-40	**70-30**	**80-20**	**Cross Validation**
Naïve Bayes Word2Vec	74,06	73,20	75,09	74,87
Naïve Bayes TF – IDF	67,90	65,52	68,37	64,26
DTree Word2Vec	72,06	75,09	74,03	74,05
DTree TF- IDF	65,20	67,33	68,89	66,59

TABLO III. Sınıflandırma sonuçları

	Music	**Science**	**Politics**	**Gaming**	**Cars**
NB Tf – Idf	%26,64	%16,42	%16,56	%22,15	%18,23
NB W2V	%22,04	%21,95	%18,87	%19,39	%17,75
DTree Tf-Idf	%14,63	%27,07	%17,08	%24,02	%17,20
DTree W2V	%21,65	%18,96	%19,76	%22,05	%17,58

VI. Sonuçlar

10.1 Gb Reddit yorumları ile Word2Vec modeli eğitilmiştir. Model toplamda 800 milyon kelime grubu ile eğitilmiş ve 450 bin benzersiz kelimeden sözlük oluşturulmuştur. Model eğitimi 16 çekirdek/paralellik ve 8 iterasyon ile toplamda 4 saat 36 dakika sürmüştür. Eğitilen modelden, önceden belirlenen "*music*", "*science*", "*politics*", "*gaming*", "*cars*" kategorilerinden, her biri 1000 adet olmak üzere anahtar kelimeler elde edilmiştir. Elde edilen bu kelimelerden hazırlanan veri seti ile Naïve Bayes ve Karar Ağaçları algoitmalarıyla sınıflandırma gerçekleştirilerek, sunulan yaklaşımın çalışma performansı test edilmiştir. Elde edilen sonuçlara göre Word2Vec, yoğun vektör yaklaşımı ile TF – IDF 'in seyrek vektör yaklaşımından daha iyi performans göstermektedir. Yapılan karşılaştırmalarda, karar ağaçlarının yoğun vektör ile daha iyi çalıştığı gözlenmektedir. Gelecek çalışmalarda, anahtar kelime çıkarımının, daha geniş kapsamlı incelenmesi ve GPU destekli algorimalar kullanılarak, performans karşılaştırılması yapılması planlanmaktadır.

Kaynaklar

[1] Hutto. C and Bell. C., "Social Media Gerontology: Understanding Social Media Usage among a Unique and Expanding Community of Users", 2014 47th Hawaii International Conference on System Science

[2] Blei. D., Ng. Andrew., Jordan. M., "Latent Dirichlet Allocation" Journal of Machine Learning Research 3 (2003) 993-1022

[3] Robertson. S., "Understanding inverse document frequency: On theoretical arguments for IDF". Journal of Documentation. 60 (5): 503–520.

[4] Qingguo. Z., Chengzhi. Z., "Automatic Chinese Keyword Extraction Based on KNN for Implicit Subject Extraction", 2008 International Symposium on Knowledge Acquisition and Modeling

[5] Zeng. P, Tan. Q, Yan. Y, Xie. Q, Xu. J, Cao. W, "Automatic Keyword Extraction Using Word Embedding and Clustering" 2017 International Conference on Computer Systems, Electronics and Control (ICCSEC)

[6] Ogul. I.U., Ozcan. C., Hakdagli. O., "Fast Text Classification With Naïve Bayes Method On Apache Spark" 2017 25th Signal Processing and Communications Applications Conference (SIU)

[7] Ogul. I.U., Ozcan. C., Hakdagli. O., "Text Classification with Spark Support Vector Machine", 1. Ulusal Bulut Bilişim Ve Büyük Veri Sempozyumu B3s'17

[8] Hakdagli. O., Ozcan. C., Ogul. I.U., " Stream Text Data Analysis On Twitter Using Apache Spark Streaming", 2018 26th Signal Processing and Communications Applications Conference (SIU)

[9] T. Mikolov, I. Sutslever, K. Chen, G. Corrado, J.Dean "Distributed Representations of Words and Phrases and their Compositionality ", https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf

[10] Zaharia. M., Chowdhury. M., Das. T., Ankur. D., Ma. J., "Resilient distributed datasets: a fault-tolerant abstraction for in-memory cluster computing", NSDI'12 Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation Pages 2-2