Avrupa Bilim ve Teknoloji Dergisi
Sayı 21, S. 323-332, Ocak 2021

Derleme Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
No. 21, pp. 323-332, January 2021

Review Article

TF-IDF ve Doc2Vec Tabanlı Türkçe Metin Sınıflandırma Sisteminin Başarım Değerinin Ardışık Kelime Grubu Tespiti ile Arttırılması

Doğancan Kınık[1], Aysun Güran[2*]

[1] Doğuş Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, İstanbul, Türkiye, (ORCID: 0000-0003-0207-7194), 20172105001@dogus.edu.tr
[2*] Doğuş Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, İstanbul, Türkiye (ORCID: 0000-0001-7066-0635), adogrusoz@dogus.edu.tr

(İlk Geliş Tarihi 11 Eylül 2020 ve Kabul Tarihi 18 Ocak 2021)

(**DOI:** 10.31590/ejosat.774144)

ATIF/REFERENCE: Kınık, D., Güran, A., TF-IDF ve Doc2Vec Tabanlı Türkçe Metin Sınıflandırma Sisteminin Başarım Değerinin Ardışık Kelime Grubu Tespiti ile Arttırılması. *Avrupa Bilim ve Teknoloji Dergisi*, (21), 323-332.

Öz

TF-IDF terim ağırlıklandırma ölçümü kelimelerin metinler içinde geçme sıklığı bilgisine dayalıdır. Bu ölçüm kelimeler arasındaki anlamsal ilişkiyi barındırmamaktadır. Yapay sinir ağlarına dayalı olan Doc2Vec metodu kelimeler ve kelimeleri içeren dokümanlar arasındaki anlamsal ilişkiyi barındırmakta ve yönetilebilir boyutlu doküman vektörlerinin elde edilmesini sağlamaktadır. Ardışık kelime grubu tespitinin metin madenciliği üzerindeki olumlu etkileri literatürde sunulan pek çok çalışma tarafından belirtilmiştir. Bu çalışmada, hem geleneksel TF-IDF terim ağırlıklandırma ölçümünün, hem de YSA'lara dayalı bir yöntem olan Doc2Vec yönteminin kullanımı ile vektörleştirilen dokümanlar üzerinde temel makine öğrenmesi sınıflandırıcılarının ve topluluk öğrenmesi algoritmalarının başarım değerleri kıyaslanmıştır. Sınıflandırıcılar farklı uzunluklarda haber dokümanlarını içeren 4 farklı Türkçe veri kümesi üzerinde uygulanmıştır. Çalışmamızın literatüre olan katkısı, sınıflandırma aşamasına geçilmeden önce dokümanların içindeki ardışık kelime grubu tespitinin gerçekleştirilmesi ve dokümanların bu kelime öbeklerinin tek bir kelime gibi ele alınmasıyla vektörleştirildikten sonra, uygulanan sınıflandırıcıların başarım değerlerinin arttığının gösterilmesi olmuştur. Ardışık kelime grubu tespiti için kelimelerin birlikte geçme sıklığı prensibine dayalı olan bir prensip dışında, Türkçe Vikipedi'nin kelime bağlantıları da kullanılmış ve dokümanlar içinde az sayıda geçmesine rağmen anlamlı olan ardışık kelime öbeklerinin tespiti gerçekleştirilebilmiştir. Ardışık kelime grubu tespiti ile sınıflandırma deneylerinin hemen hemen tümünde daha yüksek başarım değerleri elde edilmiştir.

Anahtar Kelimeler: TF-IDF, Doc2Vec, Ardışık kelime grubu tespiti, Topluluk öğrenmesi, Metin sınıflama

Enhancing the Performance of TF-IDF and Doc2Vec based Turkish Text Categorization System with Phrase Modeling

Abstract

TF-IDF term weighting measure is based on frequency of words in texts. This measure doesn't capture the semantic relationship between words. Doc2Vec which is based on artificial neural networks can capture the semantic relations between the words and it enables to yield document vectors of a more manageable size. Consecutive word detection has been reported to have important effects on text mining by many studies. Consecutive word phrases are important for expressing the semantic integrity within the texts. In this study, the performances of traditional machine learning classifiers and ensemble learning algorithms are compared on four different Turkish datasets which are vectorized with both traditional TF-IDF term weighting measurement and Doc2Vec method. The classifiers have been applied on 4 different Turkish datasets containing news documents of different lengths. The contributions of our study are "to apply consecutive word detection process to the documents before the classification phase" and "to show that the performances of the applied classifiers' results have been increased after the consecutive word detection phase is applied". In addition to the approach based on frequency of words for consecutive word detection, we also use the url links of Turkish Wikipedia. By using consecutive word detection, higher performance values are presented in almost all classification experiments.

Keywords: TF-IDF, Doc2Vec, Phrase detection, Ensemble learning, Text categorization

[1] Sorumlu Yazar: adogrusoz@dogus.edu.tr

1. Giriş

Metin sınıflandırma serbest formatta yazılmış metinleri önceden belirlenmiş olan kategorilere atama işlemidir. Metin sınıflandırma sayesinde metin kümeleri için kavramsal bir görünüm oluşturulabilir. Günümüzde oldukça önemli uygulamaları vardır. Haberlerin kategorilere göre sınıflandırılması, akademik makalelerin teknik alanlarına göre ayrıştırılması, maillerin sahte ya da gerçek olarak ayrılması gibi pek çok alanda metin sınıflandırılması kullanılmaktadır.

İstatistiksel metin sınıflandırma, insanlar tarafından etiketlendirilmiş kategorilere dayanarak, metinler üzerinde otomatik sınıflandırma yapabilmek için makine öğrenimi algoritmalarını kullanır. Bu yaklaşımda serbest yazılmış bir metin x=[x(1),x(2),…x(p)] şeklinde bir özellik vektörü ile gösterilir. Burada x(i),i=1,…,p, genellikle belgedeki kelimeleri, n-gramları, sözdizimsel veya anlamsal olarak etiketlenmiş ifadeleri veya adlandırılmış varlıkları (insan isimleri, şehir adları gibi) ifade eder. Belirli bir dokümanda bu özellikleri ifade etmek için genellikle kelime torbası (bag of word) yöntemi kullanılır. Bu yaklaşım, kelimelerin metin parçaları içindeki geçme sıklığı bilgisine dayalıdır. Kelime frekansı–ters doküman frekansı (term frequency- inverse document frequency- TF-IDF) terim ağırlıklandırma yöntemi en çok kullanılan yöntemlerdendir. Bu kelime ağırlıklandırma modelinde, kelimeler birbirlerinden bağımsız değerlendirilmektedir ve veri kümesindeki doküman sayısı arttığında değerlendirilecek kelime sayısı da artacağından kelimeleri ifade eden vektörlerin boyutu da artmaktadır. Aynı zamanda kelime çantası modeline dayalı sistemler kelimeler arasındaki anlamsal ilişkiyi taşıyamamaktadır. Literatürde, kelime sıklığı tekniklerinin yaratmış olduğu bu olumsuz durumlara karşılık, kelimeler arasındaki anlamsal ilişkileri belirleyebilen, yönetilebilir boyutlu kelime vektörlerinin çıkarılmasını sağlayan, yapay sinir ağlarına (YSA) dayalı kelime gösterim metotları önerilmiştir. Bu metotlardan Word2Vec isimli çalışma 2013 yılında Mikalov ve diğerleri tarafından gerçekleştirilmiştir [1]. Bu metot ile kelimeler komşu kelimeleri ile birlikte değerlendirilmekte ve benzer anlamlı kelimeler birbirlerine yakın olacak şekilde vektörler ile temsil edilebilmektedir. Bu yöntem kelimelerin vektörleştirilme aşamasında Sürekli Kelime Çantası Modeli (Continous Bag of Words- CBOW) ve Gram Atla Modeli (Skip-gram) isimli iki farklı öğrenme mimarisine dayanmaktadır. CBOW mimarisinde, hedeflenen kelime, bir pencere boyutu içinde ele alınan bir kelimenin sağındaki ve solundaki komşu kelimelere bakılarak tespit edilmektedir. Skip-gram mimarisinde ise CBOW modelinin tersine pencere merkezine oturtulan bir kelime baz alınarak, komşu kelimelere ulaşılmaya çalışılır. CBOW ve Skip-gram mimarileri öğrenme aşamasında hesaplama maliyetini en iyilemek adına Hiyerarşik Softmax (Hierarchical Softmax-HS) veya Olumsuz Örnekleme (Negative Sampling-NS) algoritmalarından birini kullanmaktadır. Bu algoritmalardan HS düşük frekanslı kelimeleri içeren veri kümelerinde, NS ise yüksek frekanslı kelimeleri içeren veri kümelerinde daha iyi başarım değeri sergilemektedir. Mikalov ve diğerleri yayınladıkları farklı bir çalışma ile tıpkı kelimeler gibi, dokümanların da yönetilebilir boyutlu vektörler ile ifade edilebileceğini göstermişlerdir [2]. Bu yayınlarında ifade ettikleri yöntem Doc2Vec yöntemidir. Doc2Vec, Word2Vec metodunda kelime vektörlerinin yanına her bir döküman için bir döküman vektörünün eklenmesi prensibi ile çalışmaktadır. Bu çalışma prensibiyle dokümanlar da kelimeler gibi sabit boyutlu vektörler ile gösterilebilemktedir. Doc2Vec ile hem doküman, hem de kelime vektörleri üretilebilir. Doc2Vec modelinde, Doküman Vektörlerinin Dağıtılmış Bellek Modeli (Distributed Memory Model of Paragraph Vectors- PV-DM) ve Doküman Vektörünün Dağıtılmış Kelime Çantası Modeli (Distributed Bag of Words-Paragraph Vector- PV-DBOW) mimarileri kullanılmaktadır. Bu mimariler Word2Vec'in CBOW ve Skip-gram mimarilerine eştir.

Bu çalışmada dört farklı haber veri kümesi kullanılarak haberlerin sınıflandırılması işlemi gerçekleştirilmiştir. Bu sınıflandırma işlemi esnasında bir ön işlem aşaması olarak kelime grubu tespiti işleminin yapılması önerilmiş ve bu önerinin sistem başarım değerini arttırdığı gösterilmiştir. İlk olarak TF-IDF terim ağırlıklandırma ve Doc2Vec kelime gömme teknikleri kullanılarak kelime vektörleri oluşturulmuş ve sınıflandırma işlemi buna göre yapılmıştır. Daha sonra kelime gruplama yöntemi ile tespit edilen anlamlı ardışık kelimelerin (Mustafa Kemal Atatürk, Türkiye Cumhuriyeti vb...) kullanılması ile sınıflandırma işlemi tekrarlanmış ve kelime grubu tespitinin metin sınıflandırma üzerindeki yüksek değerli sonuçları paylaşılmıştır. Ayrıca kelime grubu tespiti sırasında kullanılan frekans tabanlı bir yöntemin dışında, tespit edilemeyen az sayıda geçme ihtimali olan ardışık kelime öbekleri için, TürkçeVikipedi'nin kullanılması da önerilmiştir. Word2Vec, Doc2Vec gibi kelime gömme tekniklerinin metin sınıflandırma üzerinde oldukça yüksek başarım değerleri ürettiğini gösteren çalışmalar mevcuttur [3][4]. Bu çalışmada dokümanların direkt vektörleştirilmesine olanak tanıyan Doc2Vec yöntemi de kullanılarak sınıflandırma işlemi gerçekleştirilmiştir. Doc2Vec yöntemi kullanılırken, aynen TF-IDF yönteminde olduğu gibi, veri kümlerinin öncelikle orijinal halleri kullanılmış; daha sonra kelime gruplama yöntemi ve Vikipedi'den alınan kelime grupları kullanılarak yeni bir veri kümesi oluşturulup, testler bir de bu veri kümeleri üzerinde tekrarlanmıştır. Doc2Vec yöntemi kullanıldığında TF-IDF'e oranla daha başarılı sonuçlar elde edilmiştir. Ayrıca kelime gruplama yöntemi kullanıldığında, TF-IDF ve Doc2Vec ile sayısallaştırılan dokümanların daha yüksek başarım değerleri ile sınıflandırıldığı ortaya konmuştur. Doc2Vec modeli öğrenimindeki yaklaşımlar, PV-DM ve PV-DBOW olarak belirlenmiştir. Bu yaklaşımlar sırasıyla Word2Vec'in CBOW ve Skip-gram isimli öğrenme mimarilerine benzemektedir. Bu çalışmada aynı zamanda bu farklı mimarilerin ürettiği vektörlerin birleşimiyle elde edilen kelime vektörlerinin kullanımı da önerilmiş ve birleşim işlemi ile başarım değerinin daha yüksek sonuçlara ulaşıldığı gösterilmiştir.

Çalışmamız kapsamında sınıflandırma algoritmaları olarak geleneksel makine öğrenmesi yöntemleri olan Naive Bayes, K-En yakın komşuluk (K-nearest Neighborhood-KNN), Lojistik Regresyon (LR), Karar Destek Makineleri (Support Vector machines- SVM), Karar Ağaçları (Decision Trees- DT), Çok Katmanlı Algılayıcılar (Multi-Layer Perceptrons- MLP) gibi bireysel sınıflandırıcılarının yanı sıra topluluk öğrenmesi metotlarından olan Rassal Orman (Random Forest- RF), Bagging (BG), Adaboost (AB) algoritmları da kullanılmıştır. Ayrıca son olarak en başarılı üç sınıflandırma algoritması Çoğunluk oylaması (Majority Voting) ile birleştirilmiş ve elde edilen sonuçlar paylaşılmıştır. Sonuç olarak sıklıkla topluluk öğrenmesi metotlarının bireysel sınıflandırıcılara göre daha iyi sonuçlar ürettiği gözlenmiştir.

Çalışmanın geri kalanı şu şekilde ilerlemektedir: Bölüm 2'de literatür taramasına yer verilmiştir. Bölüm 3'te TF-IDF kelime ağırlıklandırma değeri ele alınmış, Bölüm 4'te yapay sinir ağlarına dayalı olan Word2Vec ve Doc2Vec kelime gömme

teknikleri anlatılmıştır. Bölüm 5 ile ardışık kelime grubu tespiti irdelendikten sonra, Bölüm 6 ile çalışmamızda kullanılan veri kümeleri tanıtılmıştır. Bölüm 7'de nihayi sonuçlar yorumlanmış ve çalışmamız Bölüm 8 ile sonlandırılmıştır.

2. Literatür Çalışması

Şahin, G. [5], farklı kategorilere ait Türkçe metinleri Word2Vec algoritmasını kullanılarak sınıflandırmışlardır. Çalışmalarında öncelikli olarak tüm metinlerde kelime vektörleri bulunmuş, daha sonra her bir doküman içerdiği kelimelerin vektör ortalaması alınarak temsil edilmeye çalışılmıştır. Sınıflandırma işlemi SVM algoritması kullanılarak gerçekleştirilmiş ve Word2Vec algoritmasının TF-IDF baz alınarak yapılan sınıflandırma işlemlerine oranla daha başarılı sonuçlar ürettiği gösterilmiştir.

Çelenli, vd. [6], Türkçe metinler üzerinde Doc2Vec algoritmasını uygulamışlardır. Metinleri KNN ve SVM gibi algoritmalar ile sınıflandırmışlardır. Aynı sınıflandırıcılar üzerinde Doc2Vec ve TF-IDF'ten elde ettikleri sonuçları birbiriyle kıyaslamışlardır.

Sarı ve Özbayoğlu [7], Java programla dili ile geliştirilmiş bir kütüphane olan DeepLearning4J kütüphanesini kullanarak, köşe yazarlarından topladıkları veriler üzerinde Doc2Vec algoritmasının iki versiyonu olan PV-DBOW ve PV-DM mimarilerini uygulamışlar ve toplam 20000 verisi olan bir veri kümesinde köşe yazarlarına ait yazılardan yazarı tahmin etmeye yönelik bir çalışma yapmışlardır. Çalışmalarında bazı yazarların diğerlerine göre çok daha ayırt edici olarak tespit edilebildiğini görmüşlerdir. Bu yaklaşımın yazar profili çıkarma, intihal tespiti ve hangi yazar kalemlerinin birbirine daha yakın olduğunun tespit edilebilmesi gibi amaçlar için kullanılabileceğini savunmuşlardır.

Karcıoğlu ve Aydın [8], Twitter sosyal platformundan toplanmış İngilizce ve Türkçe kısa iletiler üzerinde metin sınıflandırma çalışması yapmışlardır. Hem İngilizce, hem de Türkçe veri kümesinde kelime çantası modeli ile oluşturulmuş model ile Word2Vec ile oluşturulmuş olan modeli kıyaslamışlardır. Ayrıca kelimeleri köklerine ayırarak ve kelimeleri köklerine ayırmadan iki farklı versiyon ile testlerini yapmışlardır. İlgili modelleri SVM ve LR sınıflandırıcılarını kullanarak sınıflandırmışlar ve sonuçları birbiriyle kıyaslamışlardır.

Deniz vd. [9], Türkçe mailler üzerinde metin sınıflandırma işlemi uygulamışlardır. Türkçe veri kümesi kullanılarak sahte mailleri tespit etmeye çalışmışlardır. Maillerin vektörler ile ifade edilebilmesi için Doc2Vec algoritmasını kullanmışlar ve farklı sınıflandırma algoritmaları kullanarak bulunan sonuçları birbiriyle kıyaslamışlardır.

Erşahin vd [10], Türkçe metinler üzerine sözlük tabanlı ve makine öğrenmesi tabanlı yaklaşımların birleşimi olan hibrit bir yaklaşım ile duygu analizi uygulaması gerçekleştirmişlerdir. Yazarlar, bu hibrit yaklaşımının ortalama başarı oranını %7 oranında arttırdığını tespit etmişlerdir.

Sel vd. [11], metin sınıflandırma işleminde boyut indirgeme ve özellik seçimi işlemleri ile sınıflandırma sisteminlerinin başarım değerlerini arttırmaya çalışmışlardır. Çalışmalarında kullanmış oldukları veri kümesini ön işlem aşamalarından geçirdikten sonra Doc2Vec metodunu kullanılarak veri kümesini oluşturan dokümanları vektörleştirerek sınıflandırmışlardır. Daha sonra özellik seçimi işlemini gerçekleştirmişler ve özellik seçiminin başarı sonuçlarını yükselttiğini göstermişlerdir.

Erdinç ve Güran [12], 5 milyon Türkçe haber dokümanından oluşan bir veri kümesini farklı ön işlem aşamalarından geçirdikten sonra veri kümelerini Word2Vec, Doc2Vec ve FastText metotları ile eğiterek kelime vektörleri elde etmişlerdir. Bu kelime vektörlerinin kullanılması ile dokümanların sayısallaştırılmasını sağlamışlardır. Ardından yaklaşık 2 milyon haber dokümanı eğitim ve test amaçlı kullanılarak farklı makine öğrenmesi metotları ile sınıflandırmışlardır. En iyi sonucu FastText metodu ile vektörleştirilen dokümanların sınıflandırılması ile elde ettiklerini belirtmişlerdir.

Güler ve Tantuğ [13], popüler olarak araştırılan diller üzerinde yapılan araştırmaların Türkçe gibi zengin dillerde ortaya çıkan problemlerin tamamını aydınlatamadığını belirtmişlerdir. Yazarlar, Word2Vec modelini kelimelerin eklerine ve köklerine ayrılma işlemlerinin farklı şekillerde yapıldığı Türkçe metinler üzerine eğitmişlerdir. Ayrıca FastText modelini de eğitip, kelime benzetmesi, duygu analizi, metin sınıflandırma gibi görevlerin sonuçlarını kıyaslamışlardır.

3. TF-IDF Kelime Ağırlıklandırma Ölçümü

Dokümanların bilgisayarlar tarafından anlaşılabilmesi için sayılar ile ifade edilmesi gerekmektedir. Bu işlem dokümanların analiz edilmesi için makine öğrenimi sürecinde temel bir adımdır. Dokümanları oluşturan temel yapı kelimelerdir. Bir dokümanı vektörler ile ifade etmek için kelimeler TF-IDF ölçümü gibi ölçümlerle ifade edilebilir. TF-IDF değeri aşağıdaki şekilde hesaplanabilir:

$$TF - IDF(t,d,D) = TF(t,d) * IDF(t,D) \quad (1)$$

$$TF(t,d) = \log(1 + frekans(t,d)) \quad (2)$$

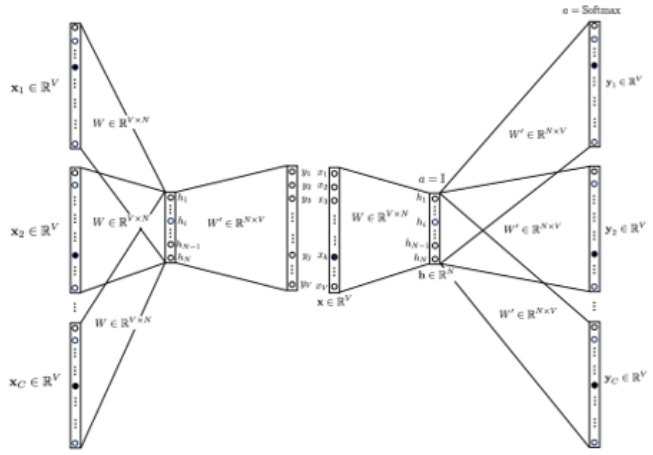
$$IDF(t,D) = \log(\frac{N}{say(d \in D : t \in d)}) \quad (3)$$

Burada frekans(t,d) kelime t'nin d.dokümanda kaç kez geçtiği bilgisidir; say(d∈D:t∈d) ifadesi ise kelime t'nin D derlemini oluşturan dokümanlardan kaç tanesinde barındırıldığı bilgisidir. TF-IDF, bir dokümandaki herhangi bir kelimenin dokümanla ne kadar alakalı olduğunu gösteren bir sayısal ölçümdür. TF-IDF kelimelerin birbirlerinden bağımsız olduğunu kabul etmekte ve kelimeler arasındaki anlamsal ilişkiyi ifade edememektedir. Ayrıca bir doküman veri kümesindeki doküman sayısının artması TF-IDF ölçüm değeri kullanılarak sayısallaştırılan dokümanların çok daha uzun vektörler ile ifade edilmesini sağlamaktadır. Son zamanlarda dokümanların yönetilebilir boyutlu vektörler ile ifade edilmesini sağlayan Yapay Sinir ağlarına dayalı olan Doc2Vec isimli bir metot önerilmiştir [2]. Çalışamızda gerçekleştirilen Türkçe metin sınıflama veri kümelerindeki dokümanlar hem TF-IDF, hem de YSA'lara dayalı olan Doc2Vec metotu ile vektörleştirildiken sonra makine öğrenmesi algoritmaları ile sınıflandırılmış ve dokümanların vektörleştirilmesi esnasında ardışık kelime tespitine dayalı bir öneri sunularak sistem başarım değerinin arttırıldığı gösterilmiştir.

4. Word2Vec ve Doc2Vec Modelleri

4.1. Word2Vec Modeli

Word2Vec modeli YSA temelli bir modeldir. Bu model giriş katmanı, gizli katman ve çıkış katmanından oluşmaktadır. Word2Vec iki farklı öğrenme mimarisine sahiptir. Bu mimariler CBOW ve Skip-Gram mimarileridir. Her iki öğrenme mimarisinde de veri kümelerindeki her kelime için bir-sıcak vektör adı verilen bir giriş vektörü ve kelimeler arasındaki anlamsal ilişkiyi barındıran bir çıkış vektörü (gömülü kelime vektörü) oluşturulmaktadır. CBOW mimarisinde Şekil 1'de gösterildiği gibi bir kelimenin belli bir pencere boyutu içindeki komşu kelimelerine (sağındaki ve solundaki kelimelere) bakılmaktadır. Bu komşu kelimlerin baz alınmasıyla hedef kelime tahmin edilmektedir. Skip-gram mimarisinde ise, CBOW mimarisinin tersine pencere merkezine oturtulan hedef kelimeye bakılarak komşu kelimelere ulaşılması hedeflenmektedir.



Şekil 1. CBOW ve Skip-gram Modelleri

CBOW modelinde, yarı eğiticili model için kullanılan veri kümesindeki her kelime için $x_k=1$ ve $k\neq k'$ için $x_{k'}=0$ özelliklerini sağlayan x bir sıcak vektörleri yaratılır ve sözlük boyutunun |V| olduğu varsayımı altında ağa sunulan bir-sıcak vektörler ile giriş katmanı ve gizli katman arasında bulunan W ağırlık matrisinin kullanılması ile Eş. 4 ile belirtilen durum uygulanır:

$$\mathrm{h} = W^{\mathrm{T}}\mathrm{x} \tag{4}$$

Gizli katmanın çıkış değerlerinin hesaplanması için, ağa girdi olarak verilen giriş vektörlerinin ortalaması alınmaktadır. Hem CBOW, hem de Skip-gramda giriş katmanı ile gizli katman arasında aktivasyon fonksiyonu kullanılmamaktadır. Gizli katman ve çıkış katmanı arasında $N \times V$ boyutlu $W' = \{w'_{ij}\}$ kullanılmaktadır (i=1,2,..., N; j=1,2,...,V). Bu ağırlık değerleri kullanılarak Eş.5 ile belirtilen işlem gerçekleştirilmektedir:

$$W'^{T}\mathrm{h} = \mathrm{u} \tag{5}$$

$\mathbf{v}'_{w_j}$ vektörü W' matrisinin j. kolonu olsun. Vx1 boyutlu $\mathbf{u}$ vektörünün bileşenleri ise u_j olsun. Bu durumda u_j değerleri Eş. 6 ile ifade edilebilir:

$$u_j = \mathrm{v}'^{\,T}_{w_j}\mathrm{h} \tag{6}$$

Yukarıdaki ifadede $\mathrm{h} = \mathrm{v}_{w_I}$ yerine konulursa Eş. 7 ile belirtilen ifade elde edilir:

$$u_j = v'^{\,T}_{w_j} v_{w_I} \tag{7}$$

Aslında u_j ifadesi bir skor değeri gibi düşünülebilir. Bu skor w_I içerik kelimesinin giriş ve gizli katman arasındaki W matrisi ile elde edilen (v_{w_I}) vektör gösterimi ile bu kelimeden sonra gelebilecek olan w_j kelimesinin gizli katman ile çıkış katmanı arasında bulunan W' matrisi ile elde edilen (v'_{w_j}) vektör gösterimi arasındaki noktasal çarpımdır. Diğer bir ifadeyle kelimeler arasındaki benzerliktir. Bu aşamadan sonra u_j skorları softmax fonksiyonu kullanılarak normalize edilmekte ve değerler Eş. 8 ile belirtilen şekilde olasılık değerlerine çevrilmektedir:

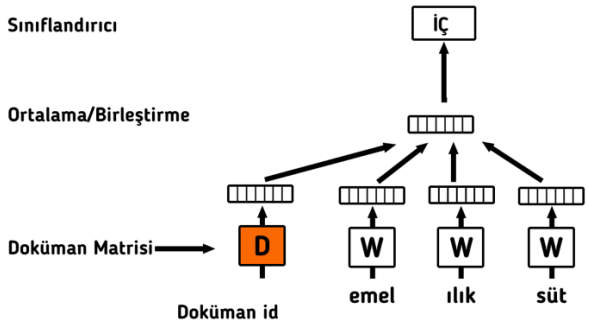
$$p(w_j|w_I) = \hat{y_j} = \frac{\exp(u_j)}{\sum_{j'=1}^{V}\exp(u_{j'})} \tag{8}$$

Hem CBOW, hem de Skip-gram geriye yayılım (backpropagation) algoritmasını kullanmaktadır. Bu algoritma, ağın çıktısının belirlendiği ileri besleme ve oluşan hatanın gradiyenti azaltacak şekilde geri yayılarak ağırlıkların güncellendiği geri besleme safhalarını içermektedir. Algoritmanın işleyişi, hedeflenen ve elde edilen çıktılar arasındaki hatanın kabul edilir düzeye inmesini sağlamak amacıyla öncelikle gizli katman ile çıkış katmanı arasındaki ağırlık değerlerinin, daha sonra ise gizli katman-giriş katmanı arasındaki tüm ağırlık değerlerinin güncellenmesi prensibine dayalıdır. Ağın eğitiminin tamamlanmasının ardından CBOW modelinde gizli katman ile çıkış katman arasındaki W ağırlık matrisi kelimelerin gömülü vektörlerini içeren matris olarak kullanılmaktadır. Bu matristeki n. sütun sözlükteki n.kelimenin N boyutlu uzaydaki vektörüdür. Skip-gramda ise ağın eğitiminin tamamlanmasının ardından giriş katmanı ve gizli katman arasında elde edilen ağırlık matrisi kullanılmaktadır.

4.2. Doc2Vec Modeli

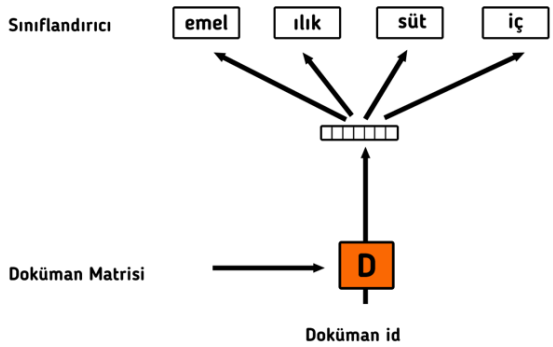
Doc2Vec modeli prensip olarak Word2Vec modeli prensiplerine dayanmaktadır. Bu model, Word2Vec mimarisine derlemdeki her bir doküman için ayrı bir doküman vektörünün eklenmesi ile yaratılmıştır. Bu doküman vektörleri ile derlemdeki dokümanlar uzunluklarından bağımsız bir şekilde sayısallaştırılabilmektedir. Doc2Vec modeli iki farklı öğrenme mimarisine sahiptir. PV-DM öğrenme modeli Word2Vec modelindeki CBOW'a karşılık gelmektedir.

Şekil 2 PV-DM öğrenme mimarisini göstermektedir. Bu şekil incelendiğinde kelime vektörlerini barındıran W matrisinin elde edilmesi sırasında, doküman vektörlerini içeren D doküman matrisinin de elde edildiği görülmektedir.



Şekil. 2. PV-DM Modeli

Şekil 3 ise Doc2Vec modelinin bir diğer öğrenme mimarisi olan ve Word2Vec modelinde Skip-gram modeline karşılık gelen PV-DBOW modelini göstermektedir. Bu modelde de Skip-gram mantığına paralel bir mantık ile dokümanlar uzunluklarından bağımsız bir şekilde vektörleştirilmektedir.



Şekil. 3. PV-DBOW Modeli

5. Kelime Grubu Modelleme

5.1 Kelime Grupları

Çalışmamızda bazı kelime gruplarının sahip olduğu kelimelerden bağımsız olarak daha özel anlamalara sahip olduğu görülmüştür. Örneğin 'Türkiye Büyük Millet Meclisi' ve 'Türkiye Cumhuriyeti Devleti' kelime grupları metin kümesinde daha özel bir anlam oluştururlar. Bu kelime gruplarının tek bir vektör halinde ifade edilmesi, vektör temsilinde kelime boyutunu azaltılmasını ve anlamlı kelime gruplarının oluşturulmasını sağlamaktadır.

Metin içerisindeki kelime gruplarını tanıyabilmek için pek çok teknik geliştirilmiştir. Bunlardan bir tanesi, basit ve veri odaklı bir yaklaşımdır [2]. Bu yaklaşımda kelime grupları Eş. 9 ile belirtilen ifade yardımıyla tespit edilebilmektedir.

$$skor(w_i, w_j) = \frac{say(w_i,w_j) - \delta}{say(w_i) * say(w_j)} \quad (9)$$

Bu formülde $say(w_i)$ metodu w_i kelimesinin ilgili doküman içerisinde kaç kez geçtiğini belirtmektedir. Aynı şekilde $say(w_i, w_j)$ metodu iki kelimenin doküman içerisinde birlikte kaç kez geçtiğini ifade etmektedir. δ bir katsayı olarak kullanılır. İlgili kelime grubunun minimum kaç kez metin kümesinde geçtiği ile ilgilenir. Kullanıcı tarafından tanımlanan bir parametredir. Çok nadir kelime öbeklerinin geçtiği büyük bir kelime grubu listesinin oluşmasını engeller. Kendi ölçümlerimizi yaparken δ katsayısı, beş olarak seçilmiştir. Eş.9 ile belirtilen formüle göre her kelime grubuna bir skor değeri verilmektedir. Daha sonra bir eşik değeri belirlenmesi gerekir. Bu değer kullanıcı tarafından belirlenen bir değerdir. Veri kümesinin büyüklüğüne ve içeriğine göre değişiklik gösterebilir. Burada doğru değeri bulabilmek için birkaç kez test yapmak gerekebilir. Kendi testlerimizde bu değer on olarak alınmıştır. Yukardaki formülden hesaplanan skor, eşik değerinden büyükse bigram kelime grubu oluşturulabilir. Ayrıca formüle eklemeler yapılarak trigram ve diğer n-gramlar da kullanılabilir. Çalışmamızda n={1,2,3} olacak şekilde kelime grupları oluşturulmuştur.

5.2 Vikipedi Kullanımı

Vikipedi, yazılmış olan verileri toplu şekilde erişime sunmaktadır. Bu erişimle, Vikipedideki metinlere ait başlıklara, içeriklere ve içeriklerde geçen bağlantılara ulaşılabilmektedir. Buradaki bağlantılar genellikle bulundukları metin ile ilgili anlamsal ifadeler taşıyan ya da metinin hangi konuyla ilgili olduğu konusunda bilgi veren bağlantılardır.

Bu çalışmada Bölüm 5.1'deki yöntemle ardışık kelime grupları bulunduktan sonra, veri kümesininde az sayıda bulunabilecek olan ardışık kelime gruplarını tespit etmek için, Vikipedi'de bağlantı olarak ifade edilmiş kelime gruplarının veri kümemizde var olup olmadığı kontrol edilmiştir. Böylece herhangi bir kelime grubu; Vikipedi'de bağlantı olarak ifade edildiyse bu kelime grupları da tek bir vektör ile ifade edilebilecektir. Bu şekilde 'Orgeneral James Jones', 'Orgeneral Hilmi Özkök', 'Gaziantep SSK Bölge Hastanesi', 'Anabilim Dalı Öğretim Üyesi' gibi kelime grupları da tıpkı Bölüm 5.1'deki yöntem ile tespit edilebilen kelime grupları gibi bir vektör ile ifade edilebilmiştir. Bu kullanım sayesinde n>3 olduğunda karşılaşılan bellek problemine bir çözüm getirilmeye çalışılmıştır. Vikipedi'nin bağlantıları kullanılarak n>3 olan kelime grupları da tespit edilebilmiştir.

6. Veri Kümeleri

Çalışmamız da dört farklı veri kümesi kullanılmıştır. Bu veri kümleri sırasıyla '1150Haber', 'Hürriyet6c-1k', 'Milliyet-9c-1k' ve 'AA haber' veri kümeleridir. Bu veri kümeleri ile ilgili bilgiler Tablo 1'de belirtildiği gibidir.

Tablo 1. Veri Kümeleri

Veri Kümesi	Sınıf Sayısı	Doküman Sayısı
1150 Haber	5	1150
Hürriyet6c-1k	8	6000
Milliyet-9c-1k	9	9000
AA Haber	8	20000

- 1150Haber veri kümesi, içerisinde 1150 adet haber metni içeren bir veri kümesidir. Bu veri kümesinin içerisindeki haber metinleri beş farklı kategoriye ayrılmıştır. Bu kategoriler sırasıyla; 'Ekonomi', 'Magazin, 'Sağlık', 'Siyasi' ve 'Spor' kategorileridir. [14]
- Hürriyet6c-1k veri kümesi, içerisinde 6000 adet haber metni içeren bir veri kümesidir. Bu veri kümesinin içerisindeki haber metinleri altı farklı kategoriye ayrılmışlardır. Bunlar sırasıyla; 'Dünya', 'Ekonomi', 'Gündem', 'Siyaset', 'Spor', 'Yaşam' şeklindedir. [15]
- Milliyet-9c-1k veri kümesi, 9000 adet haber metni içermektedir. Bu veri kümesinde dokuz farklı kategori bulunmaktadır. Bu kategoriler sırasıyla; 'Yaşam', 'Türkiye', 'Spor', 'Siyaset', 'Güncel', 'Ekonomi', 'Ege', 'Dünya' ve 'Kafe' şeklindedir. [15]
- AA Haber veri kümesi veri kümelerimiz içerisinden 20000 adet haber metni ile en büyük veri kümemizdir. Anadolu Haber Ajansı grubunun haberlerini içeren bir veri kümesidir ve sekiz farklı kategoriden oluşmaktadır. Bunlar sırasıyla; 'Kültür ve Sanat', 'Ekonomi', 'Eğitim ve Bilim', 'Çevre ve Sağlık', 'Politika', 'Spor', 'Türkiye' ve 'Dünya' şeklindedir. [16]

7. Araştırma Sonuçları ve Tartışma

Bu bölümde araştırma sonuçları paylaşılmıştır. Tablo 2, 1150 Haber veri kümesini oluşturan dokümanların TF-IDF terim ağılırıklandırma ölçümünün kullanılması ile vektörleştirilmesinin ardından veri kümesinin %80-%20 eğitim-test oranı ile ayrılması ve 10'lu çapraz doğrulama tekniği ile sınıflandırılması sonuçlarını içermektedir. Bu tablodan görüleceği gibi ardışık kelime grubu tespitinin gerçekleştirilmesi hem bireysel sınıflandırıcıların, hem de topluluk öğrenmesi sınıflandırıcılarının başarım değerlerinin artmasını sağlamıştır. Sınıflandırma sonuçları incelendiğinde en iyi sınıflandırma metotolarının LR, SVM, MLP ve MV sınıflandırıcıları olduğu (**0.94**) görülmüştür.

Tablo 2. 1150Haber TF-IDF Sonuçları

Sınıflandırıcılar	Kelime Grubu Tespiti Yapılmadan	Kelime Grubu Tespiti Yapıldıktan Sonra
NB	0.91 (± 0.06)	**0.92 (± 0.04)**
LR	0.93 (± 0.07)	**0.94 (± 0.04)**
SVM	0.92 (± 0.06)	**0.94 (± 0.06)**
MLP	0.93 (± 0.05)	**0.94 (± 0.02)**
DT	0.81 (± 0.15)	**0.82 (± 0.18)**
KNN	0.89 (± 0.06)	**0.90 (± 0.04)**
MV	0.93 (±- 0.07)	**0.94 (± 0.06)**
RF	0.87 (± 0.05)	**0.91 (± 0.13)**
BG	0.81 (±- 0.10)	**0.83 (± 0.02)**
AB	0.88 (± 0.05)	**0.90 (± 0.12)**

Tablo 3, 1150 Haber veri kümesini oluşturan dokümanların Doc2Vec metodu ile vektörleştirilmesinin ardından veri kümesinin %80-%20 eğitim-test oranı ile ayrılması ve 10'lu çapraz doğrulama tekniği ile sınıflandırılması sonuçlarını içermektedir. Bu tablodan görüleceği gibi ardışık kelime grubu tespitinin gerçekleştirilmesi çoğunlukla sınıflandırıcılarının başarım değerlerinin artmasını sağlamıştır (Başarım değeri artan algoritmalar koyu renkle vurgulanmıştır.) Sınıflandırma sonuçları incelendiğinde en iyi sınıflandırma metodunun Doc2Vec mimarilerinden elde edilen doküman vektörlerinin birleştirilmesi ile elde edilen durumunda (DBOW+DMC), MV ve BG algoritmalarının kullanımıyla elde edildiği görülmüştür.

Tablo 3. 1150Haber Doc2Vec Sonuçları

	Kelime Grubu Tespiti Yapılmadan			Kelime Grubu Tespiti Yapıldıktan Sonra		
Sınıflandırıcı	**PV-DBOW**	**PV-DMC**	**DBOW+DMC**	**PV-DBOW**	**PV-DMC**	**DBOW+DMC**
NB	0.70(± 0.03)	0.78(± 0.02)	0.76(± 0.02)	**0.71**(± 0.03)	0.78(± 0.02)	**0.79**(± 0.02)
LR	0.80(± 0.04)	0.81(± 0.03)	0.84(± 0.03)	**0.81**(± 0.04)	**0.82**(± 0.04)	**0.85**(± 0.04)
SVM	0.81(± 0.04)	0.83(± 0.04)	0.87(± 0.04)	0.81(± **0.02**)	0.83(± 0.03)	0.87(± 0.04)
MLP	0.75(± 0.05)	0.80(± 0.04)	0.80(± 0.04)	0.75(± **0.02**)	**0.81**(± 0.03)	0.80(± 0.03)
DT	0.84(± 0.03)	0.85(± 0.02)	0.87(± 0.05)	0.84(± 0.04)	**0.86**(± 0.02)	0.87(± 0.04)
KNN	0.71(± 0.06)	0.75(± 0.03)	0.74(± 0.05)	**0.72**(± 0.06)	0.75(± 0.05)	**0.75**(± 0.04)
MV	0.92(± 0.04)	0.93(± 0.01)	0.93(± 0.01)	**0.93**(± 0.03)	0.93(± 0.01)	**0.94**(± 0.01)
RF	0.91(± 0.02)	0.92(± 0.04)	0.93(± 0.02)	0.91(± 0.02)	**0.93**(± 0.02)	0.92(± 0.01)
BG	0.92(± 0.01)	0.93(± 0.02)	0.94(± 0.01)	0.92(± 0.01)	0.93(± 0.01)	0.94(± 0.04)
AB	0.89(± 0.03)	0.91(± 0.04)	0.92(± 0.02)	0.89(± **0.02**)	0.91(± 0.02)	**0.93**(± 0.04)

Tablo 4 Milliyet haber veri kümesini oluşturan dokümanların TF-IDF terim ağılırıklandırma ölçümünün kullanılması ile vektörleştirilmesinin ardından veri kümesinin %80-%20 eğitim-test oranı ile ayrılması ve 10'lu çapraz doğrulama tekniği ile sınıflandırılması sonuçlarını içermektedir. Bu tablodan görüleceği gibi ardışık kelime grubu tespitinin gerçekleştirilmesi hem bireysel sınıflandırıcıların hem de topluluk öğrenmesi sınıflandırıcılarının başarım değerlerinin artmasını sağlamıştır. Sınıflandırma sonuçları incelendiğinde en iyi sınıflandırma metodunun SVM sınıflandırıcısı olduğu (**0.91**) görülmüştür.

Tablo 4. Milliyet TF-IDF Sonuçları

Sınıflandırıcılar	Kelime Grubu Tespiti Yapılmadan	Kelime Grubu Tespiti Yapıldıktan Sonra
NB	0.79 (± 0.03)	**0.81(±0.02)**
LR	0.85 (± 0.022)	**0.86(±0.02)**
SVM	0.90 (± 0.03)	**0.91(±0.03)**
MLP	0.89 (± 0.02)	**0.90(±0.02)**
DT	0.70 (± 0.04)	**0.71(±0.05)**
KNN	0.70 (± 0.03)	**0.71 (± 0.02)**
MV	0.88 (± 0.02)	**0.90(±0.01)**
RF	0.81 (± 0.01)	**0.82(±0.04)**
BG	0.82 (± 0.03)	**0.84(±0.01)**
AB	0.82 (± 0.01)	**0.85(±0.03)**

Tablo 5 Milliyet Haber veri kümesini oluşturan dokümanların Doc2Vec metodu ile vektörleştirilmesinin ardından elde edilen sonuçları göstermektedir. Öncelikle bu tablonun ilk yarısına bakıldığında Doc2Vec'in iki öğrenme mimarisinin birleşimi ile elde edilen durumun mimarilerin ayrı ayrı uygulanmasıyla elde edilen durumdan sıklıkla daha iyi sonuç verdiği görülmüştür. Tablonun ikinci yarısı incelendiğinde ardışık kelime grubu tespitinin yine bazı metotların başarım değerini arttırdığı tespit edilmiştir.

Tablo 5. Milliyet Doc2Vec Kelime Gruplama Kullanılmadan

	Kelime Grubu Tespiti Yapılmadan			Kelime Grubu Tespiti Yapıldıktan Sonra		
Sınıflandırıcı	**PV-DBOW**	**PV-DMC**	**DBOW+DMC**	**PV-DBOW**	**PV-DMC**	**DBOW+DMC**
NB	0.70(± 0.01)	0.74(± 0.01)	**0.78(± 0.01)**	**0.73(± 0.01)**	0.74(± 0.01)	0.78(± 0.02)
LR	0.85(± 0.01)	0.86(± 0.02)	**0.93(± 0.01)**	**0.90(± 0.01)**	0.86(± 0.01)	0.93(± 0.02)
SVM	0.90(± 0.04)	0.86(± 0.01)	**0.94(± 0.02)**	0.89(± 0.02)	0.86(± 0.02)	0.94(± 0.01)
MLP	0.89(± 0.02)	0.84(± 0.02)	**0.93(± 0.03)**	**0.90(± 0.03)**	**0.90(± 0.03)**	**0.94(± 0.02)**
DT	0.82(± 0.03)	0.83(± 0.03)	**0.88(± 0.02)**	**0.83(± 0.04)**	**0.84(± 0.01)**	**0.89(± 0.01)**
KNN	0.89(± 0.06)	0.87(± 0.04)	0.87(± 0.01)	0.89(**± 0.01**)	**0.89(± 0.02)**	**0.88(± 0.02)**
MV	0.94(± 0.02)	0.92(± 0.02)	**0.95(± 0.02)**	0.94(± 0.02)	**0.94(± 0.01)**	0.95(± 0.03)
RF	0.84(± 0.01)	0.88(± 0.03)	**0.88(± 0.03)**	0.84(± 0.05)	**0.89(± 0.03)**	0.88(± 0.05)
BG	0.93(± 0.05)	0.92(± 0.04)	**0.94(± 0.05)**	**0.94**(± 0.03)	**0.93(± 0.03)**	0.94(± 0.01)
AB	0.88(± 0.02)	0.90(± 0.01)	**0.90(± 0.02)**	**0.89**(± 0.04)	**0.91(± 0.02)**	**0.91(± 0.03)**

Tablo 6 Hürriyet haber veri kümesini oluşturan dokümanların TF-IDF terim ağırlıklandırma ölçümünün kullanılması ile vektörleştirilmesinin ardından veri kümesinin %80-%20 eğitim-test oranı ile ayrılması ve 10'lu çapraz doğrulama tekniği ile sınıflandırılması sonuçlarını içermektedir. Bu tabloda ardışık kelime tespitinin, pek çok algoritmanın başarım değerini arttırdığını gözler önüne sermiştir. Topluluk öğrenmesi metotları bireysel sınıflandırıclara göre daha iyi sonuçlar üretmişlerdir. En iyi başarım değeri (0.82) topluluk öğrenmesi metotlarından olan BG ve AB algoritmalarına ait çıkmıştır.

Tablo 7 Hürriyet Haber veri kümesini oluşturan dokümanların Doc2Vec metodu ile vektörleştirilmesinin ardından ulaşılan başarım değerlerini içermektedir. Yine bu tablonun ilk yarısına bakıldığında Doc2Vec'in iki öğrenme mimarisinin birleşimi ile elde edilen durumun mimarilerin ayrı ayrı uygulanmasıyla elde edilen durumdan sıklıkla daha iyi sonuç verdiği görülmüştür. Tablonun ikinci yarısı incelendiğinde ardışık kelime grubu tespitinin yine bazı metotların başarım değerini arttırdığı tespit edilmiştir. Topluluk öğrenmesi metotları bu veri kümesi üzerinde daha yüksek sınıflandırma sonuçları üretmiştir.

Tablo 6. Hürriyet TF-IDF Sonuçları

Sınıflandırıcılar	Kelime Grubu Tespiti Yapılmadan	Kelime Grubu Tespiti Yapıldıktan Sonra
NB	0.78 (± 0.04)	0.78 (±0.02)
LR	0.79 (± 0.04)	**0.80 (±0.02)**
SVM	0.78 (± 0.03)	**0.79 (±0.04)**
MLP	0.78 (± 0.04)	**0.80 (±0.02)**
DT	0.76 (± 0.05)	0.76 (±0.02)
KNN	0.71 (± 0.03)	**0.72(±0.04)**
MV	0.79(±0.04)	**0.80 (±0.01)**
RF	0.78 (± 0.02)	**0.80 (±0.04)**
BG	0.79(±0.03)	**0.82(±0.02)**
AB	0.80 (± 0.02)	**0.82(±0.01)**

Tablo 7. Hürriyet Doc2Vec Kelime Gruplama Kullanılmadan

	Kelime Grubu Tespiti Yapılmadan			Kelime Grubu Tespiti Yapıldıktan Sonra		
Sınıflandırıcı	**PV-DBOW**	**PV-DMC**	**DBOW+DMC**	**PV-DBOW**	**PV-DMC**	**DBOW+DMC**
NB	0.69(± 0.05)	0.75(± 0.02)	**0.78(± 0.02)**	0.75(± 0.01)	**0.76(± 0.01)**	0.78(± 0.02)
LR	0.78(± 0.01)	0.74(± 0.01)	**0.81(± 0.01)**	0.78(± 0.02)	**0.78(± 0.02)**	0.81(± 0.02)
SVM	0.78(± 0.02)	0.76(± 0.02)	**0.83(± 0.03)**	0.77(± 0.02)	0.76(± 0.02)	0.82(± 0.01)
MLP	0.81(± 0.01)	0.77(± 0.03)	**0.82(± 0.01)**	**0.82(± 0.02)**	0.81(± 0.02)	0.82(± 0.01)
DT	0.79(± 0.01)	0.78(± 0.02)	**0.85(± 0.02)**	0.78(± 0.01)	**0.80(± 0.01)**	0.85(± 0.01)
KNN	0.77(± 0.01)	0.80(± 0.01)	0.78(± 0.02)	0.76(± 0.01)	**0.81(± 0.03)**	0.78(± 0.01)
MV	0.92(± 0.03)	0.88(± 0.02)	0.92(± 0.02)	**0.94(± 0.02)**	0.88(± 0.01)	0.92(± 0.01)
RF	0.87(± 0.02)	0.85(± 0.01)	**0.88(± 0.01)**	0.87(± 0.01)	**0.86(± 0.01)**	0.88(± 0.01)
BG	0.93(± 0.01)	0.86(± 0.02)	**0.94(± 0.02)**	0.93(± 0.03)	**0.87(± 0.01)**	0.94(± 0.02)
AB	0.87(± 0.02)	0.89(± 0.03)	0.89(± 0.02)	**0.88(± 0.02)**	**0.90(± 0.02)**	**0.90(± 0.03)**

Tablo 8 TF-IDF terim ağırlıklandırma değerinin kullanılması ile sayısallaştırılan dokümanların sınıflandırılması sonuçlarını yansıtmaktadır. Ardışık kelime grubu tespiti hem bireysel hem de topluluk öğrenmesi metotlarının tümünün başarım değerlerini arttırmıştır. Topluluk öğrenmesi metotlarından olan AB ve MV algoritmaları 0.84 değeri ile en iyi başarım değerlerine sahip olan algoritmalar olmuştur.

Tablo 8. AAHaber TF-IDF Sonuçları

Sınıflandırıcılar	Kelime Grubu Tespiti Yapılmadan	Kelime Grubu Tespiti Yapıldıktan Sonra
NB	0.80 (± 0.02)	**0.81 (±0.02)**
LR	0.81 (± 0.02)	**0.82(±0.03)**
SVM	0.80 (± 0.04)	**0.81 (±0.03)**
MLP	0.77 (± 0.02)	**0.78 (±0.02)**
DT	0.75 (± 0.02)	**0.77 (±0.04)**
KNN	0.75 (± 0.01)	**0.76(±0.01)**
MV	0.82 (±0.03)	**0.84(±0.05)**
RF	0.79 (± 0.02)	**0.81 (±0.01)**
BG	0.80 (± 0.02)	**0.82(±0.03)**
AB	0.82 (± 0.03)	**0.84 (±0.05)**

Tablo 10. AA Haber veri kümesini oluşturan dokümanların Doc2Vec metodu ile vektörleştirilmesinin ardından ulaşılan başarım değerlerini içermektedir. Yine bu tablonun ilk yarısına bakıldığında Doc2vec'in iki öğrenme mimarisinin birleşimi ile elde edilen durumun, mimarilerin ayrı ayrı uygulanmasıyla elde edilen durumdan sıklıkla daha iyi sonuç verdiği görülmüştür. Tablonun ikinci yarısı incelendiğinde ardışık kelime grubu tespitinin yine bazı metotların başarım değerini arttırdığı tespit

edilmiştir. Topluluk öğrenmesi metotları bu veri kümesi üzerinde daha yüksek sınıflandırma sonuçları üretmiştir. En iyi başarım değeri 0.94 değeriyle topluluk öğrenmesi metotlarından BG algoritmasına aittir. Doc2Vec, TF-IDF değerlerinden daha yüksek değerler elde edilmesini sağlamıştır.

Tablo 10. AA Haber Doc2Vec Kelime Gruplama Kullanılmadan

	Kelime Grubu Tespiti Yapılmadan			**Kelime Grubu Tespiti Yapıldıktan Sonra**		
Sınıflandırıcı	**PV-DBOW**	**PV-DMC**	**DBOW+DMC**	**PV-DBOW**	**PV-DMC**	**DBOW+DMC**
NB	0.70(± 0.03)	0.71(± 0.02)	**0.71(± 0.02)**	**0.76(± 0.01)**	**0.72(± 0.02)**	**0.72(± 0.02)**
LR	0.78(± 0.05)	0.71(± 0.03)	0.78(± 0.03)	**0.79(± 0.01)**	0.71(± 0.02)	**0.79(± 0.01)**
SVM	0.78(± 0.04)	0.72(± 0.04)	**0.79(± 0.04)**	0.77(± 0.01)	**0.80(± 0.03)**	0.78(± 0.02)
MLP	0.81(± 0.05)	0.73(± 0.04)	**0.82(± 0.04)**	0.81(± 0.02)	0.72(± 0.01)	**0.83(± 0.03)**
DT	0.75(± 0.05)	0.70(± 0.08)	**0.85(± 0.05)**	0.75(± 0.03)	**0.71(± 0.01)**	0.85(± 0.04)
KNN	0.82(± 0.06)	0.70(± 0.05)	0.82(± 0.05)	0.81(± 0.01)	**0.73(± 0.02)**	0.81(± 0.02)
MV	0.90(± 0.02)	0.84(± 0.01)	0.90(± 0.01)	0.90(± 0.01)	0.84(± 0.02)	**0.91(± 0.01)**
RF	0.80(± 0.02)	0.70(± 0.02)	0.77(± 0.01)	0.77(± 0.03)	**0.72(± 0.02)**	**0.78(± 0.03)**
BG	0.93(± 0.01)	0.92(± 0.01)	0.93(± 0.02)	0.92(± 0.01)	**0.93(± 0.04)**	**0.94(± 0.02)**
AB	0.92(± 0.03)	0.85(± 0.03)	0.82(± 0.04)	**0.93(± 0.02)**	0.85(± 0.03)	**0.84(± 0.03)**

8. Sonuç

Bu çalışmada, geleneksel kelime çantası modeli (TF-IDF terim ağırlıklandırma ölçümü) ve YSA'lara dayalı bir yöntem olan Doc2Vec yöntemi ile vektörleştirilen dokümanlar üzerinde bireysel sınıflandırıcılar ve topluluk öğrenmesi algoritmalarının başarım değerleri kıyaslanmıştır. Bu amaçla farklı uzunluklarda haber dokümanlarını içeren 4 farklı Türkçe veri kümesi kullanılmıştır.

Geleneksel kelime çantası modeline dayalı olan TF-IDF terim ağırlıklandırma ölçümü kelimeler arasındaki anlamsal ilişkileri yansıtmamaktadır. Üstelik kelime çantası modelinde veri kümesinin boyutunun artması dokümanların vektörleştirilmesi sırasında vektör uzayının boyutunun artmasına da neden olmaktadır. YSA'lara dayalı bir metot olan Doc2Vec metodu ise kelimeler ve dolayısıyla kelimeleri içeren dokümanlar arasındaki anlamsal ilişkiyi barındırmakta ve yönetilebilir boyutta doküman vektörlerinin elde edilmesini sağlamaktadır. Çalışmamızda TF-IDF ve Doc2Vec metotları ile vektörleştirilen dokümanlar üzerinde uygulanan sınıflandırıcıların başarım değerleri kıyaslanmıştır. Deneylerimiz sonunda sıklıkla Doc2Vec ile vektörleştirilen doküman vekörlerinin sınıflandırma aşamasında daha iyi başarım değerlerinin üretilmesini sağladığı gözlemlenmiştir. Ayrıca genelde topluluk öğrenmesi metotları bireysel algoritmalardan daha yüksek sonuçlar elde edilmesini sağlamıştır. Doc2Vec modeli iki farklı öğrenme mimarisine sahiptir. PV-DM öğrenme modeli Word2Vec modelindeki CBOW'a karşılık gelmektedir. PV-DBOW öğrenme modeli ise Word2Vec modelindeki Skip-gram modeline karşılık gelir. Bu çalışmada bu iki öğrenme modelinin üretmiş olduğu vektörler birleştirilmiş ve metinler bu şekilde sayısallaştırıldıktan sonra hem bireysel, hem de topluluk öğrenmesi metotları ile sınıflandırılmıştır. Sonuç olarak iki öğrenme mimarisinden elde edilen doküman vektörlerinin birleşimimin sıklıkla daha iyi sonuçlar ürettiği gözlemlenmiştir.

Bu çalışmanın literatüre olan katkısı; sınıflandırma aşamasına geçilmeden önce dokümanların içindeki ardışık kelimelerin tespitinin gerçekleştirilmesi ve dokümanların bu kelime öbeklerinin tek bir kelime gibi ele alınmasıyla vektörleştirildikten sonra uygulanan sınıflandırıcıların başarım değerlerinin arttığının gösterilmesi olmuştur. Ardışık kelime tespiti sırasında kullanılan kelimelerin birlikte geçme sıklığı prensibine dayalı olan bir prensip dışında, Türkçe Vikipedi'nin kelime bağlantıları da kullanılmış ve dokümanlar içinde az sayıda geçmesine rağmen anlamlı ardışık kelime öbeklerinin tespiti gerçekleştirilebilmiştir. Ardışık kelime grubu tespiti ile sınıflandırma deneylerinin hemen hemen tümünde daha yüksek başarım değerleri elde edilmiştir.

Kaynakça

[1] Mikolov T, Chen K, Corrado G, Dean J. (2013). Efficient estimation of word representations in vector space. Proceedings of Workshop at ICLR. Scottsdale, Arizona 2-4 Mayıs 2013.

[2] Quoc Le and Tomas Mikolov. (2014). Distributed representations of sentences and documents. In Proceedings of the 31st International Conference on Machine Learning (ICML 2014), pages 1188–1196, Beijing, China.

[3] Ay Karakuş, B., Talo, M., Hallaç, İ. R., & Aydin, G. (2018). Evaluating deep learning models for sentiment classification. Concurrency and Computation: Practice and Experience, e4783.

[4] Karasoy, O., Ballı, S. (2017). Classification Turkish SMS with deep learning tool Word2Vec. In Computer Science and Engineering (UBMK), 2017 International Conference on (pp. 294-297). IEEE.

[5] Şahin, G. (2017). Turkish document classification based on Word2Vec and SVM classifier. In 2017 25th signal processing and communications applications conference (SIU) (pp. 1-4). IEEE.

[6] Çelenli, H. İ., Öztürk, S. T., Şahin, G., Gerek, A., Ganiz, M. C. (2018). Document Embedding Based Supervised Methods for Turkish Text Classification. In *2018 3rd International Conference on Computer Science and Engineering (UBMK)* (pp. 477-482). IEEE.

[7] Sarı, M., Özbayoğlu, A. M. (2018). Classification of Turkish Documents Using Paragraph Vector. In 2018 International Conference on Artificial Intelligence and Data Processing (IDAP) (pp. 1-5). IEEE.

[8] Karcioğlu, A. A., Aydin, T. (2019). Sentiment Analysis of Turkish and English Twitter Feeds Using Word2Vec Model. In 2019 27th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.

[9] Deniz, E., Erbay, H., Coşar, M. (2019). Classification of Turkish E-Mails with Doc2Vec. In 2019 1st International Informatics and Software Engineering Conference (UBMYK) (pp. 1-4). IEEE.

[10] Erşahin, B., Aktaş, Ö., Kilinc, D., Erşahin, M. (2019). A hybrid sentiment analysis method for Turkish. Turkish Journal of Electrical Engineering & Computer Sciences, 27(3), 1780-1793.

[11] Sel, İ., Karci, A., Hanbay, D. (2019, September). Feature Selection for Text Classification Using Mutual Information. In 2019 International Artificial Intelligence and Data Processing Symposium (IDAP) (pp. 1-4). IEEE.

[12] Erdinç, H. Y., Güran, A. (2019). Semi-supervised Turkish Text Categorization with Word2Vec, Doc2Vec and FastText Algorithms. In *2019 27th Signal Processing and Communications Applications Conference (SIU)* (pp. 1-4). IEEE.

[13] Güler, G., Tantuğ, A. C. (2020). Comparison of Turkish Word Representations Trained on Different Morphological Forms. arXiv preprint arXiv:2002.05417.

[14] Amasyalı M. F., Beken, A. (2009). "Türkçe kelimelerin anlamsal benzerliklerinin ölçülmesi ve metin sınıflandırmada kullanılması," In IEEE signal processing and communications applications conference, Antalya, Turkey, 2009.

[15] Torunoğlu, D., Çakirman, E., Ganiz, M. C., Akyokuş, S., Gürbüz, M. Z. (2011). Analysis of preprocessing methods on classification of Turkish texts. In 2011 International Symposium on Innovations in Intelligent Systems and Applications (pp. 112-117). IEEE.

[16] Tantuğ A. C. (2010), "Document categorization with modified statistical language models for agglutinative languages," International Journal of Computational Intelligence Systems, vol. 3, no. 5, pp. 632–645, 2010.