SVM, LSTM, Word2Vec ve GloVe Algoritmaları ile 3 Farklı Veri Kümesini Sınıflandırma

Merve Karagöz   
Kocaeli Üniversitesi *Kocaeli,Türkiye* mervekaragoz124@gmail.com

*Özet*—Günümüzde makine öğrenmesi birçok problemin çözümünde kolaylık sağlayarak gelişim göstermektedir. Özellikle metin verilerindeki sınıflandırma becerileri yadsınamayacak kadar iyi sonuçlar vermektedir. Makine öğrenmesi tekniklerinden olan Support Vector Machine (Destek Vektör Makinesi-SVM), Word2Vec, LSTM (Long Short-Term Memory) ve Glove kullanılarak 3000 Tweet veri kümesini, Word2Vec, LSTM Ve Glove algoritmalarını kullanarak 17000 Tweet ve Aahaber veri kümelerini sınıflandırmak hedeflenmiştir. Projenin amacı bu veri kümelerini sınıflandırarak duygu analizi yöntemiyle etiketlerini belirlemektir.

Anahtar Kelimeler— Sınıflandırma, veri kümesi, duygu analizi, support vector machine, word2vec, lstm, glove

# GİRİŞ

Makine öğrenmesi, bilgisayarların bir başarım ölçütünü örnek veri veya geçmiş deneyim kullanarak eniyilemesi olarak tanımlanabilir [1]. Bu kapsamda belirli parametrelere sahip bir model tanımlanarak bir bilgisayar programının bir eğitim verisi veya geçmiş deneyimleri kullanarak bu parametreleri iyileştirecek şekilde çalıştırılması da öğrenme sürecini oluşturur.

Makine Öğrenmesi Bilgisayar Bilimleri, Mühendislik ve İstatistik gibi farklı bilim dallarından beslenir ve sınıflandırma, regresyon, kümeleme, yoğunluk kestirimi, örüntü tanıma, uç değer tespiti, sıkıştırma, bilgi çıkarma, vb. gibi farklı görevleri yerine getirmeye çalışır [2]. Yapılan projede sınıflandırma görevi yerine getirilmeye çalışılmıştır.

Bu projenin amacı, makine öğrenmesi tekniklerinden olan Support Vector Machine (Destek Vektör Makinesi-SVM), Word2Vec, LSTM (Long Short-Term Memory) ve Glove algoritmaları kullanılarak 3 bin tweet ‘in, Word2Vec, LSTM (Long Short-Term Memory) ve Glove algoritmaları kullanılarak 17 bin tweet ‘in ve aahaber ‘in yer aldığı veri kümesini sınıflandırmak ve duygu analizi yapmaktır. Support Vector Machine sınıflandırıcısının en sık kullanılan çekirdek fonksiyonlarına (linear, polynomial, rbf) yer verilmiştir. Bu 3 farklı Support Vector Machine çekirdek fonksiyonlarının hangisinden en iyi sonuç alınabileceği heaplanmıştır. Ayrıca eğitim kümesi yüzdeleri değiştirilerek en iyi sonuca ulaşılmak amaçlanmıştır. Sınıflandırıcı açıklamalarına ve kullanılan veri kümelerinin detaylarına ilerleyen bölümlerde yer verilmiştir.

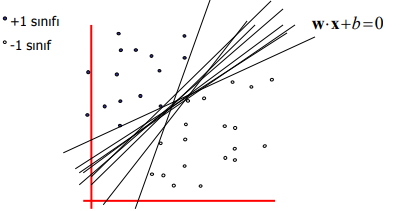
Bu rapor şu bölümlerden oluşmaktadır: 2. Bölümde yöntemler adı altında bu projede kullanılmış olan makine öğrenmesi teknikleri özetlenmektedir, 3. bölümde bu proje kapsamında sınıflandırılması yapılan veri kümeleri anlatılmaktadır, 4. bölümde sonuç başlığı altında proje içinde kullanılan makine öğrenmesi tekniklerinin sonuçlarına ve sonuçların karşılaştırılmasına yer verilmiştir. Son bölümde ise yararlanılan kaynaklara yer verilerek rapor tamamlanmıştır.

# YÖNTEM

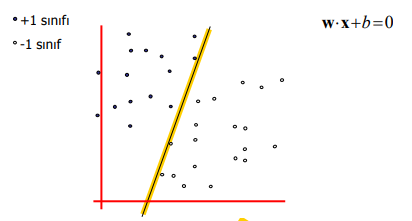
Günümüzde farklı sınıflandırma problemleri için kullanılan makine öğrenmesi yöntemilerinden olan Support Vector Machine (SVM), Word2Vec, LSTM (Long Short-Term Memory) ve Glove algoritmalarından 3 bin tweet kümesini, Word2Vec, LSTM (Long Short-Term Memory) ve Glove algoritmalarından 17 bin tweet ve aahaber veri kümesini sınıflandırmak amacıyla yararlanılmıştır. Projede kullanılan makine öğrenmesi yöntemlerine ilişkin bilgilere özet şeklinde aşağıda yer verilmektedir.

A. Support Vector Machine (Destek Vektör Makinesi) Sınıflandırıcı

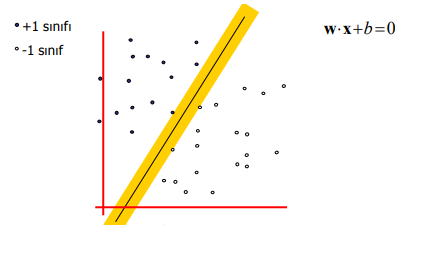
Destek Vektör Makinesi (support vector machine - SVM) veriyi sınıflandırırken sınıfların birbirlerine en yakın örneklerini bularak bu örneklerin (iki sınıfı ayıracak olan) ayırıcı yüzeye dik uzaklıklarını maksimize etmeyi amaçlar. Ayırıcı yüzeyin, veri kümesi üzerindeki başarısı değişmeden birçok farklı alternatifi olabilir. Diğer sınıflayıcılar, ayırıcı yüzey için birçok alternatif bulabilir (Resim 2.1). Bunların bir kısmı sınıflar arası uzaklığı maksimize edemez (Resim 2.2). SVM sayesinde ayırıcı yüzey her iki sınıfa da aynı mesafede ve maksimum uzaklıktadır (Resim 2.3). Bu bilgilerin görselleştirilmiş haline aşağıda yer verilmiştir [3].



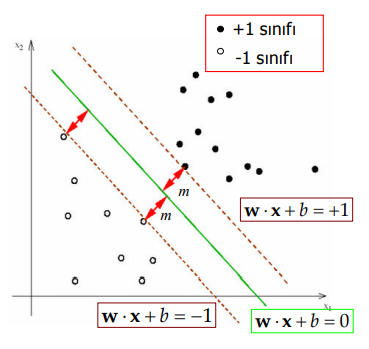
Resim 2.1



Resim 2.2

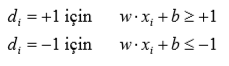


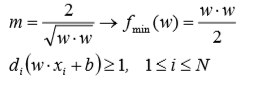
Resim 2.3



Resim 2.4

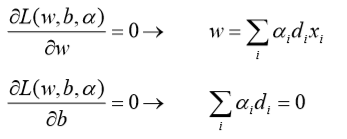
Resim 2.4 ’de görselleştirilen SVM modelinin anlaşılması için aşağıdaki (1) denklemi yazılmıştır.



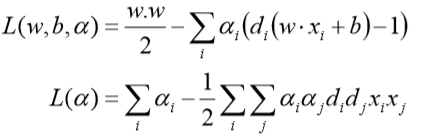


 (1)

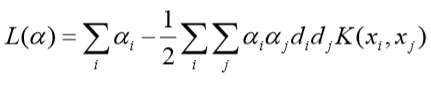
Yukarıdaki denklem (1) Lagrange denkleminin bazı veri kümeleri için klasik çözümü kolay olabilirken bu denkleme bilgisayar destekli bir çözüm üretebilmek için farklı bir yol izlenmelidir.

 (2)

Klasik çözüm uygulanarak Lagrange denklemi tek değişkene bağımlı hale getirilir (2). Bu ifadeler Lagrange denkleminde yerine yazılır (3).

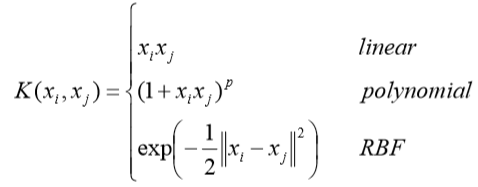
 (3)

Doğrusal olmayan veri kümeleri seçilen bir çekirdek (kernel) fonksiyonu ile bir üst uzaya haritalandıktan sonra doğrusal SVM çözümü uygulanır ve aşağıdaki sonuç elde edilir (4).



(4)

En sık ve proje içinde kullanılan çekirdek (kernel) fonksiyonları doğrusal (linear), polinom (polynomial) ve RBF fonksiyonlarıdır (5).

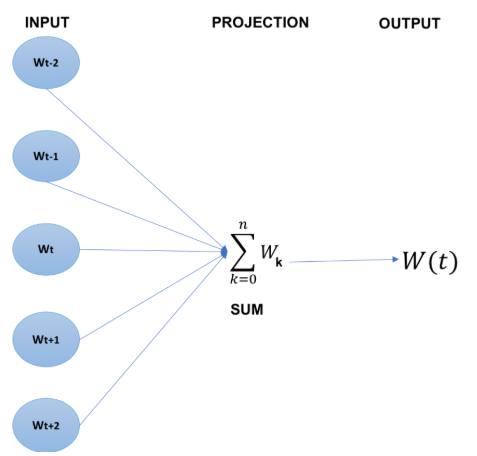


(5)

B. Word2Vec Sınıflandırıcı

Word2Vec, kelimeleri vektör uzayında ifade etmeye çalışan ***unsupervised***(no labels)ve tahmin temelli(prediction-based) bir modeldir. Word2Vec için **kelime temsil (word embedding)** yöntemidir de denilebilir. Kelime Temsil (Word Embedding),metin verisi üzerinde çalışabilmesi için öncelikle veriyi makinenin anlayacağı hale getirmeyi amaçlar. Kelime temsil yöntemleri bir dil modelleme tekniğidir, sözcükleri veya cümleleri sayısallaştırıp birer vektör haline getirir [4]. Google araştırmacı Tomas Mikolov ve ekibi tarafından 2013 yılında icat edilmiştir. 2 çeşit alt yöntemi vardır: CBOW (Continous Bag of Words) ve Skip-Gram. 2 yöntem de genel olarak birbirine benzemektedir [5].

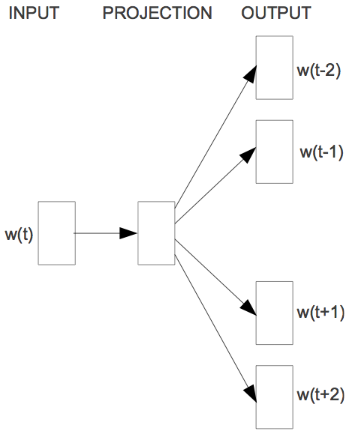
* **Continuous Bag of Words**: CBOW modelinde pencere boyutu merkezinde olmayan kelimeler girdi olarak alınıp, merkezinde olan kelimeler çıktı olarak tahmin edilmeye çalışılmaktadır. Bu durum aşağıdaki şekilde gösterilmeye çalışılmıştır (Resim 2.5). Burada w(t) ile gösterilen değer, cümlenin merkezinde bulunan ve tahmin edilmek istenen çıktı değeri iken, w(t-2) …. w(t+2) ile gösterilen değerler ise tercih edilen pencere boyutuna göre merkezde olmayan çıktı değerleridir.



Resim 2.5

* **Skip Gram**: Skip Gram modelinde pencere boyutu merkezinde olan kelimeler girdi olarak alınıp, merkezinde olmayan kelimeler çıktı olarak tahmin edilmeye çalışılmaktadır. Bu durum aşağıdaki şekilde gösterilmeye çalışılmıştır (Resim 2.6). Burada w(t) ile gösterilen değer, cümlenin merkezinde bulunan ve girdi değeri iken, w(t-2)...w(t+2) ile gösterilen değerler ise cümlenin merkezinde olmayan tercih edilen pencere boyutuna göre tahmin edilmek istenen çıktı değerleridir.

Skip Gram model ve CBOW arasındaki tek fark Skip Gram modelin CBOW’un tam tersi olmasıdır. Yani, yapay sinir ağında çıktılar ve girdilerin yeri değiştirmektedir.



Resim 2.6

 Proje içerisinde Word2Vec modelini içeren [gensim](https://radimrehurek.com/gensim/auto_examples/index.html)adındaki kütüphaneyi kullanılmıştır. Gensim, modern istatistiksel makine öğrenimini kullanarak denetlenmeyen konu modellemesi ve doğal dil işleme için açık kaynaklı bir kütüphanedir. Gensim Python ve Cython’da uygulanmaktadır. Gensim kütüphanesinde veriyi eğitebilmek için bir takım parametrelere ihtiyaç vardır, bu parametrelere ve anlamlarına aşağıda yer verilmiştir.

* Words, model için oluşturulmuş, ön işlemden geçirilmiş kelimeler listesidir. Modelin eğitilmesi için kullanılan veri kümesidir.
* Size, opsiyonel olarak integer değer alır. Kelime vektörlerinin boyutudur.
* Window, opsiyonel olarak integer değer alır. Bir cümle içindeki mevcut ve tahmin edilen kelime arasındaki maksimum mesafedir.
* Min\_count, opsiyonel olarak integer değer alır. Parametre değerinden daha düşük frekanstaki tüm kelimeler yok sayılır.
* Workers, modeli eğitmek için bu birçok işçi-iş parçacığını kullanır.
* Sample, hangi yüksek frekanslı kelimelerin rastgele altörnekleneceğinin yapılandırma eşiğidir.
* Alpha, opsiyonel olarak float değer alır. Başlangıç için öğrenme oranıdır.
* Negative, negatif örnekleme kullanılıp kullanılmayacağını belirler. Proje içerisinde 0 olarak ayarlanarak negatif örnekleme kullanılmamıştır.

C. LSTM (Long Short-Term Memory) Sınıflandırıcı

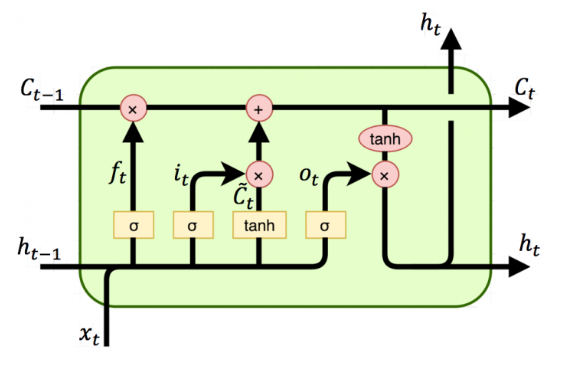
Uzun kısa süreli bellek (LSTM), değerleri rastgele aralıklarla hatırlayan bir ‘tekrarlayan sinir ağı’ (RNN) mimarisidir.

Bir LSTM, önemli olaylar arasındaki bilinmeyen boyut ve sürenin zaman gecikmeleri göz önüne alındığında zaman serilerini sınıflandırmak, işlemek ve öngörmek için oldukça uygun bir sınıflandırıcıdır.

LSTM ’ler, uzun vadeli bağımlılık sorununun önüne geçmek için açıkça tasarlanmıştır. Uzun süreler boyunca bilgileri hatırlamak pratikte varsayılan davranışlarıdır.

Bir LSTM ağı, diğer ağ üniteleri yerine veya buna ek olarak LSTM üniteleri içerir. LSTM birimi, uzun veya kısa zaman periyotlarını hatırlar. Bu kabiliyetin anahtarı, tekrarlanan bileşenlerinde hiçbir etkinleştirme işlevini kullanmamasıdır. Dolayısıyla, depolanan değer yinelemeli olarak değiştirilmez. LSTM birimleri, genellikle birkaç birim içeren “bloklar” içinde uygulanmaktadır. Bu tasarım, derin sinir ağlarında tipiktir ve paralel donanımlarla yapılan uygulamaları kolaylaştırır.

LSTM blokları, bilgi akışını kontrol eden üç veya dört “kapı” içerir. Bu kapılar, 0 ile 1 arasındaki bir değeri hesaplamak için [lojistik](https://veribilimcisi.com/2017/07/18/lojistik-regresyon/) fonksiyonu kullanarak gerçekleştirilir. Örneğin, bir “giriş” kapısı, yeni bir değerin bellekte ne derece aktığını kontrol eder. Bir “unut” kapısı, bir değerin bellekte kalma derecesini denetler. Bir “çıktı” kapısı, bloğun çıktı aktivasyonunu hesaplamak için bellekteki değerin ne kadar kullanıldığını kontrol eder [6]. LSTM ‘i kolayca anlamak amacıyla aşağıdaki görselde yapısına yer verilmiştir (Resim 2.7) [7].



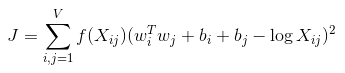
Resim 2.7

Resim 2.7 ‘de yer alan kavramların açıklaması aşağıdaki gibidir.

* h\_t, C\_t: hidden layer (gizli atman) vektörleri
* x\_t: input (giriş) vektörü
* sigma, tanh: Aktivasyon fonksiyonları

D. GloVe Sınıflandırıcı

Unsupervised algoritmaların temelinde verilerin istatistikleri yer almaktadır. Skip-gram, CBOW gibi modeller anlamsal bilgileri yakalar ama birlikte kullanılma istatistiklerini kullanmazlar. Matris ayrıştırma yöntemleri bu istatistikleri kullanmasına rağmen anlamsal ilişkileri yakalayamamaktadır. Bu tarz modellerde anlamsallık yoktur. Pennington ve diğerleri tarafından önerilen “GloVe” modeli olasılık istatistiklerinden yararlanarak yeni bir objektif fonksiyon oluşturarak bu problemi çözmeyi amaçlamaktadır [8].

(6)

Burada (Denklem 6) X\_ij, korpustaki kelime çiftinin (i, j) birlikte geçme sayısıdır. F(x) ağırlık fonksiyonu 3 gereksinime sahiptir:

* f (0) = 0 olması durumunda bütün terimler sonsuza gitme eğilimi göstermemeli.
* Birlikte az geçen kelime çiftleri için düşük ağırlık verirken ağırlık fonksiyonu azaltıcı olmamalıdır.
* X\_ij’nin büyük değerleri için bu değer biraz daha küçük olmalıdır.

# III. VERİ KÜMESİ

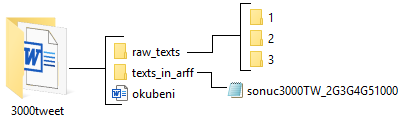
A. 3000 Tweet Veri Kümesi

Projede kullanılan veri kümesi klasörü içeriği; ‘3000tweet’ klasörü içinde bulunan ‘raw\_texts’ klasörü, ‘texts  
\_in\_arff’ klasörü ve ‘okubeni’ dosyasıdır. Duygu analizi ile metin sınıflandırma kısmında raw\_text klasörü ile işlem yapılmış, okubeni klasöründen veri kümesiyle ilgili bilgiler edinilmiştir. Veri kümesi hakkında yeterli bilgi edinilebilmesi için klasörlerin görselleştirilmiş hali resim 3.1 ‘de gösterilmiştir.

3000tweet içerisinde bulunan “texts\_in\_arff” klasörü “raw\_texts” klasöründeki 2500 metne ait text2arff yazılımıyla çıkartılmış yaklaşık 16 bin özelliği içeren “sonuc3000TW\_2G3G4G51000.arff” metin dosyasını içermektedir. Çıkarılan özellikler 2, 3, 4 ve 5 harf ngram larıdır. Ngramlardan en az 5 en çok 1000 kere geçenler listeye dahil edilmiştir.

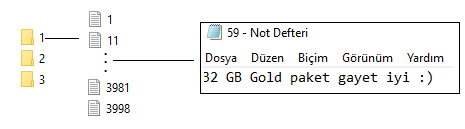
Veri kümesinde bulunan sınıf sayısı 3 ‘tür (olumlu, olumsuz ve nötr). Veri kümesindeki örnek sayısı 3000 ‘dir (3000 tane faklı tweet içerir). Veri kümesindeki özellik sayısı ise yaklaşık 16000 kadardır.

Proje içerisinde sınıflandırma amacıyla kullanılan veri kümesi raw\_texts klasöründen elde edildiğinden dolayı yazının devamında bu klasör hakkında detaylara yer verilecektir.

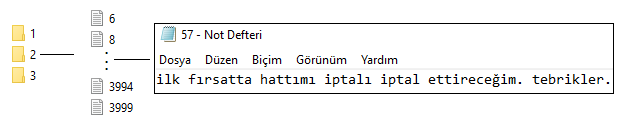


Resim 3.1

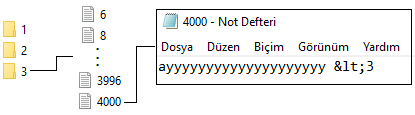
Resim 3.1 ‘de görüldüğü üzere raw\_texts klasörü içinde ‘1’, ‘2’ ve ‘3’ klasörleri yer almaktadır. ‘1’ klasörü olumlu tweetleri, ‘2’ klasörü olumsuz tweetleri ve ‘3’ klasörü ise nötr tweetleri temsil etmektedir. Klasör içeriklerinin daha iyi anlaşılması adına ‘1’ klasörü içeriğinin Resim 3.2 ‘de, ‘2’ klasörü içeriğinin Resim 3.3 ’de ve son olarak ‘3’ klasörü içeriğinin Resim 3.4 ’de görselleştirilmiş hali yer almaktadır.



Resim 3.2



Resim 3.3

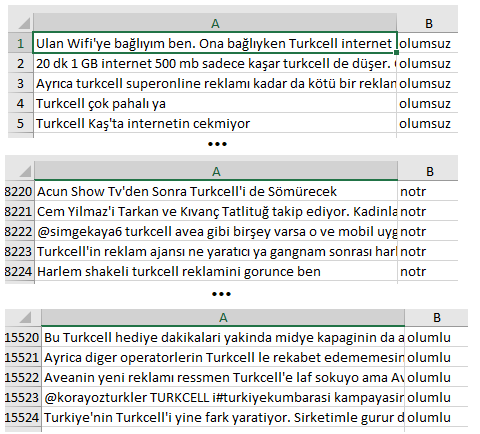


Resim 3.4

Ayrıntılı olarak tanıtılan veri kümesinde Twitter ‘dan alınmış olumlu, olumsuz ve nötr etiketleriyle klasörlendirilmiş 3000 adet tweet bulunmaktadır. Bu tweetlerin sınıflandırması duygu analizi yöntemiyle yapılmıştır.

B. 17000 Tweet Veri Kümesi

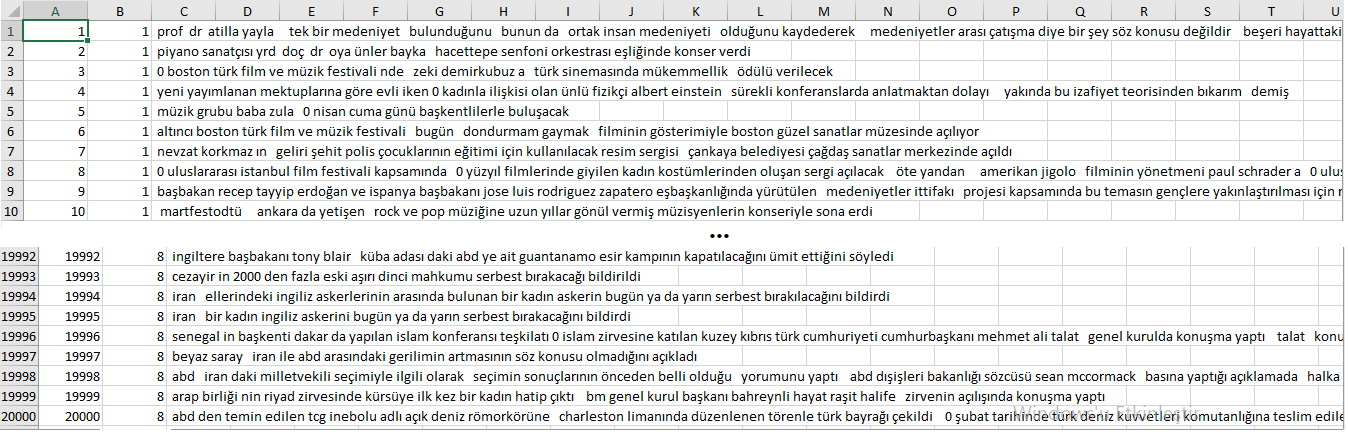
Projede kullanılan 17000 tweet veri kümesi train ve test olarak 2 farklı belge içermektedir. Proje içerisinde bu 2 farklı belge birleştirilerek tek dosya halne getirilmiştir. 17000 tweet veri kümesinde Twitter üzerinden toplanmış olumlu, olumsuz ve nötr olarak etiketlenmiş tweet ‘ler bulunmaktadır. Bu veri kümesi ile sınıflandırma yapılarak yazılan herhangi bir tweet cümlesinin etiket tahmini yapılması hedeflenmiştir. Aşağıdaki resim ile veri kümesinin daha iyi anlaşılması hedeflenmiştir (Resim 3.5).



Resim 3.5

C. Aahaber Veri Kümesi

Aahaber veri kümesi, Türkiye Ulusal Haber Ajansı olan Anadolu Ajansı tarafından yayınlanan gazete makalelerinden oluşan bir veri kümesidir. Bu veri kümesinde sekiz kategori, her kategoriye ait 2500 doküman ve toplamda 20000 adet veri bulunmaktadır. Kategoriler Türkiye, dünya, politika, ekonomi, spor, eğitim bilimi, kültür sanatı ve çevre sağlığıdır [9]. Aşağıda veri kümesi hakkında daha ayrıntılı bilgiye ulaşılması adına veri kümesinin bir kısmının görseline yer verilmiştir.



Resim 3.6

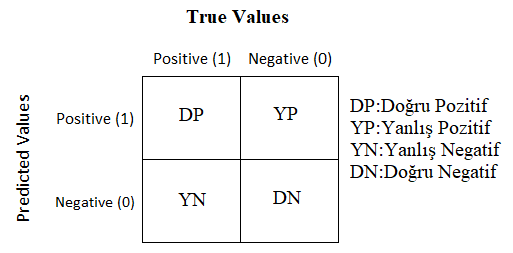
# IV. SONUÇ

3000 tweet kümesi Support Vector Machine (Destek Vektör Makinesi) ile sınıflandırılmıştır. Support Vector Machine için projede ayrıca yer verilen çekirdek (kernel) fonksiyonları doğrusal (linear), polinom (polynomial) ve RBF fonksiyonlarıdır. 3 farklı kernel (çekirdek) fonksiyonu kullanılarak en iyi sonuca ulaşmak hedeflenmiştir.

Sonuçlar başlığı altında farklı eğitim yüzdelerinden edinilen sonuçlara ve çekirdek (kernel) fonksiyonlarından elde edilen sonuçlara ‘Bulgular’ alt başlığında, proje içinde kullanılan karşılaştırma ölçütlerine ayrı bir alt başlıkta yer verilmiştir.

A. Karşılaştırma Ölçütleri

Sınıflandırma başarımını ölçmek için öncelikle hata matrisi kullanılmaktadır. Confusion matrix (hata matrisi) sınıf sayısı kadar satır ve sütundan oluşan ve her bir hücrede farklı bir değer tutan bir tablodur (Resim 4.1).



Resim 4.1 İkili sınıflandırmaya ilişkin doğru (1) ve yanlış (0) sınıflandırma sonuç sayılarını gösteren bir hata matrisi.

Hata matrisinden yararlanarak aşağıdaki ölçütler yöntemleri karşılaştırma amacıyla kullanılmaktadır:

* Doğruluk (Accuracy): Doğru olarak sınıflandırılmış girdilerin toplam girdilere oranını veren ölçüttür. Denklem (7) ile hesaplanmaktadır.

 (7)

* Kesinlik (Precision): Doğru olarak sınıflandırılmış pozitif girdilerin toplam pozitif değerlere oranıdır. Denklem (8) ile hesaplanmaktadır.

 (8)

* Geriçağırım (Recall): Doğru olarak sınıflandırılmış pozitif girdilerin gerçek pozitif değerlere oranıdır. Denklem (9) ile hesaplanmaktadır.

 (9)

* F1 ölçütü: Daha iyi bir karşılaştırma sağlamak için kesinlik ve geriçağırımın harmonik ortalaması şeklinde hesaplanan bir ölçüttür (Denklem (9)). (K:Kesinlik, G:Geriçağırım)

 (10)

* Support: Veri kümesini sınıflandırırken verilmiş olan sınıfın gerçekleşme sayısıdır.

B. 3000 Tweet Veri Kümesi İçin Bulgular

***1. Support Vector Machine Sınıflandırıcısından Elde Edilen Bulgular***

Support Vector Machine algoritmasına sınıflandırmak amacıyla verilen test size=0.20 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **1** | 0.88 | 0.10 | 0.18 | 152 |
| **2** | 0.46 | 0.99 | 0.62 | 261 |
| **3** | 0.71 | 0.05 | 0.10 | 187 |
| **accuracy** |  |  | 0.47 | 600 |
| **macroavg** | 0.68 | 0.38 | 0.30 | 600 |
| **weighted avg** | 0.64 | 0.47 | 0.35 | 600 |

Tablo 1: Test Size = 0.20 iken çekirdek (kernel) fonksiyonu ‘poly’ ile elde edilen sonuç

ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.2 Test Size = 0.20 iken çekirdek (kernel) fonksiyonu ‘poly’ ile elde edilen confusion matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **1** | 0.74 | 0.22 | 0.34 | 152 |
| **2** | 0.52 | 0.95 | 0.67 | 261 |
| **3** | 0.66 | 0.26 | 0.37 | 187 |
| **accuracy** |  |  | 0.55 | 600 |
| **macroavg** | 0.64 | 0.48 | 0.46 | 600 |
| **weighted avg** | 0.62 | 0.55 | 0.49 | 600 |

Tablo2: Test Size = 0.20 iken çekirdek (kernel) fonksiyonu ‘rbf’ ile elde edilen sonuç

ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.3 Test Size = 0.20 iken çekirdek (kernel) fonksiyonu ‘rbf’ ile elde edilen confusion matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **1** | 0.62 | 0.36 | 0.46 | 152 |
| **2** | 0.59 | 0.80 | 0.68 | 261 |
| **3** | 0.51 | 0.43 | 0.46 | 187 |
| **accuracy** |  |  | 0.57 | 600 |
| **macroavg** | 0.57 | 0.53 | 0.54 | 600 |
| **weighted avg** | 0.57 | 0.57 | 0.56 | 600 |

Tablo 3: Test Size = 0.20 iken çekirdek (kernel) fonksiyonu ‘linear’ ile elde edilen sonuç

ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.4 Test Size = 0.20 iken çekirdek (kernel) fonksiyonu ‘linear’ ile elde edilen confusion matrix

ekran görüntüsü, çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.5 Test Size = 0.20 iken accuracy (doğruluk) değerlerinin karşılaştırılması

Support Vector Machine algoritmasına sınıflandırmak amacıyla verilen test size=0.50 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **1** | 0.86 | 0.09 | 0.16 | 360 |
| **2** | 0.47 | 1.00 | 0.64 | 667 |
| **3** | 0.76 | 0.07 | 0.13 | 473 |
| **accuracy** |  |  | 0.49 | 1500 |
| **macroavg** | 0.70 | 0.38 | 0.31 | 1500 |
| **weighted avg** | 0.65 | 0.49 | 0.36 | 1500 |

Tablo 4: Test Size = 0.50 iken çekirdek (kernel) fonksiyonu ‘poly’ ile elde edilen sonuç

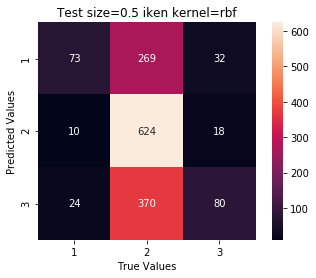
ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.6 Test Size = 0.50 iken çekirdek (kernel) fonksiyonu ‘poly’ ile elde edilen confusion matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **1** | 0.68 | 0.20 | 0.31 | 360 |
| **2** | 0.52 | 0.93 | 0.66 | 667 |
| **3** | 0.56 | 0.22 | 0.31 | 473 |
| **accuracy** |  |  | 0.53 | 1500 |
| **macroavg** | 0.58 | 0.45 | 0.43 | 1500 |
| **weighted avg** | 0.57 | 0.53 | 0.47 | 1500 |

Tablo 5: Test Size = 0.50 iken çekirdek (kernel) fonksiyonu ‘rbf’ ile elde edilen sonuç



Resim 4.7 Test Size = 0.50 iken çekirdek (kernel) fonksiyonu ‘rbf’ ile elde edilen confusion matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **1** | 0.60 | 0.37 | 0.46 | 360 |
| **2** | 0.59 | 0.78 | 0.67 | 667 |
| **3** | 0.52 | 0.43 | 0.47 | 473 |
| **accuracy** |  |  | 0.57 | 1500 |
| **macroavg** | 0.57 | 0.53 | 0.53 | 1500 |
| **weighted avg** | 0.57 | 0.57 | 0.56 | 1500 |

Tablo 6: Test Size = 0.50 iken çekirdek (kernel) fonksiyonu ‘linear’ ile elde edilen sonuç

ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.8 Test Size = 0.50 iken çekirdek (kernel) fonksiyonu ‘linear’ ile elde edilen confusion matrix

ekran görüntüsü, çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.9 Test Size = 0.50 iken accuracy (doğruluk) değerlerinin karşılaştırılması

Support Vector Machine algoritmasına sınıflandırmak amacıyla verilen test size=0.70 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **1** | 0.79 | 0.08 | 0.14 | 522 |
| **2** | 0.46 | 1.00 | 0.63 | 925 |
| **3** | 0.71 | 0.05 | 0.09 | 653 |
| **accuracy** |  |  | 0.47 | 2100 |
| **macroavg** | 0.65 | 0.37 | 0.29 | 2100 |
| **weighted avg** | 0.62 | 0.47 | 0.34 | 2100 |

Tablo 7: Test Size = 0.70 iken çekirdek (kernel) fonksiyonu ‘poly’ ile elde edilen sonuç

ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.10 Test Size = 0.70 iken çekirdek (kernel) fonksiyonu ‘poly’ ile elde edilen confusion matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **1** | 0.78 | 0.14 | 0.23 | 522 |
| **2** | 0.50 | 0.96 | 0.66 | 925 |
| **3** | 0.60 | 0.21 | 0.32 | 653 |
| **accuracy** |  |  | 0.52 | 2100 |
| **macroavg** | 0.63 | 0.44 | 0.40 | 2100 |
| **weighted avg** | 0.60 | 0.52 | 0.45 | 2100 |

Tablo 8: Test Size = 0.70 iken çekirdek (kernel) fonksiyonu ‘rbf’ ile elde edilen sonuç

ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.11 Test Size = 0.70 iken çekirdek (kernel) fonksiyonu ‘rbf’ ile elde edilen confusion matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **precision** | **recall** | **f1-score** | **support** |
| **1** | 0.60 | 0.29 | 0.39 | 522 |
| **2** | 0.57 | 0.76 | 0.65 | 925 |
| **3** | 0.48 | 0.45 | 0.46 | 653 |
| **accuracy** |  |  | 0.55 | 2100 |
| **macroavg** | 0.55 | 0.50 | 0.50 | 2100 |
| **weighted avg** | 0.55 | 0.55 | 0.53 | 2100 |

Tablo 9: Test Size = 0.70 iken çekirdek (kernel) fonksiyonu ‘linear’ ile elde edilen sonuç

ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.12 Test Size = 0.70 iken çekirdek (kernel) fonksiyonu ‘linear’ ile elde edilen confusion matrix

ekran görüntüsü, çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

Resim 4.13 Test Size = 0.70 iken accuracy (doğruluk) değerlerinin karşılaştırılması

Bulgular kısmında elde edilen accuracy (doğruluk) değerlerinin yanında etiket bilgisine ulaşım sağlanılan kod kısmına ve çıktılarına yer verilmiştir. Aşağıdaki resimlerde etiketi bulunmak istenilen sırasıyla olumlu, olumsuz ve nötr cümleler ve aldıkları etiket sonuçları yer almaktadır.



Resim 4.14 olumlu cümlenin etiket çıktısı



Resim 4.15 olumsuz cümlenin etiket çıktısı



Resim 4.16 nötr cümlenin etiket çıktısı

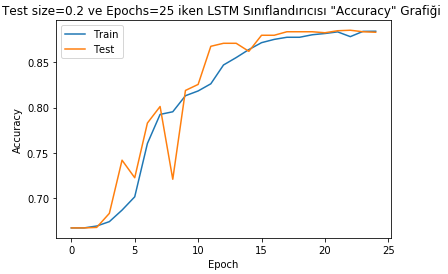
Son olarak elde edilen tüm accuracy (doğruluk) değerleri Tablo 1 ‘de gösterilmektedir. En başarılı sonuç eğitim yüzdesi %80 ve çekirdek(kernel) fonksiyonu ‘linear’ iken elde edilmiştir.

| Eğitim  Yüzdesi | Çekirdek (Kernel) Fonksiyonları ve Eğitim Yüzdesi Değişikliğine Göre Doğruluk Metriği | | |
| --- | --- | --- | --- |
| linear | polynomial | rbf |
| **TS30** | 0.54 | 0.47 | 0.52 |
| **TS50** | 0.57 | 0.48 | 0.53 |
| **TS80** | 0.57 | 0.47 | 0.55 |

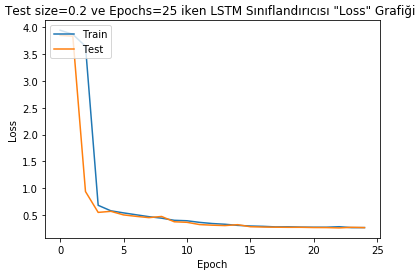
Tablo 10: Tüm accuracy (doğruluk) değerlerinin karşılaştırılması

***2. LSTM Sınıflandırıcısından Elde Edilen Bulgular***

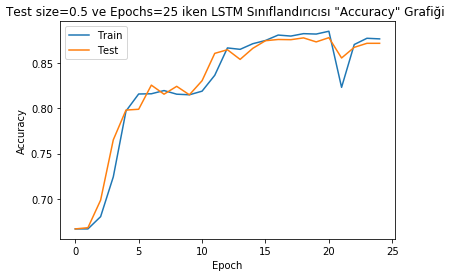
LSTM algoritmasına 3000 tweet veri kümesini sınıflandırmak amacıyla verilen test size=0.20, test size=0.50, test size=0.70 ve epochs=25 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.



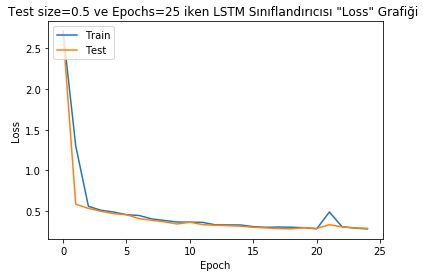
Resim 4.17 LSTM sınıflandırıcısının test size=0.2 ve 3000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



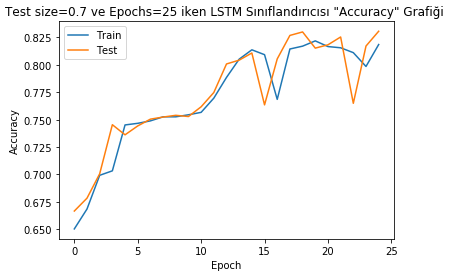
Resim 4.18 LSTM sınıflandırıcısının test size=0.2 ve 3000 tweet veri kümesi için loss grafiğinin çizimi



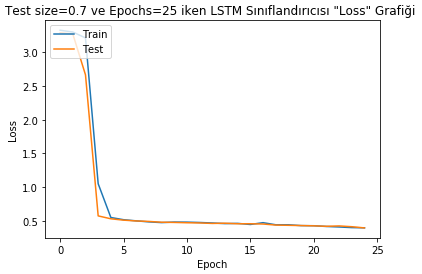
Resim 4.19 LSTM sınıflandırıcısının test size=0.5 ve 3000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.20 LSTM sınıflandırıcısının test size=0.5 ve 3000 tweet veri kümesi için loss grafiğinin çizimi



Resim 4.21 LSTM sınıflandırıcısının test size=0.7 ve 3000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



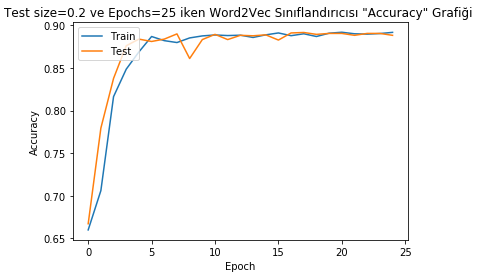
Resim 4.22 LSTM sınıflandırıcısının test size=0.7 ve 3000 tweet veri kümesi için loss grafiğinin çizimi

|  |  |
| --- | --- |
| Eğitim Yüzdesi | LSTM |
| **TS30** | 0.83 |
| **TS50** | 0.87 |
| **TS80** | 0.88 |

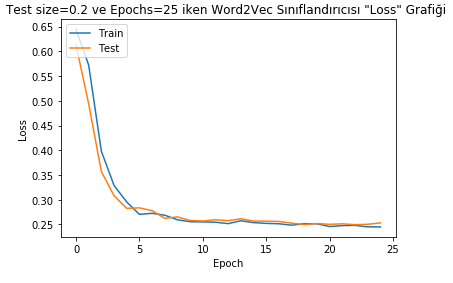
Tablo 11: LSTM sınıflandırıcısı için 3000 tweet veri kümesinin tüm accuracy (doğruluk) değerlerinin karşılaştırılması

***3. Word2Vec Sınıflandırıcısından Elde Edilen Bulgular***

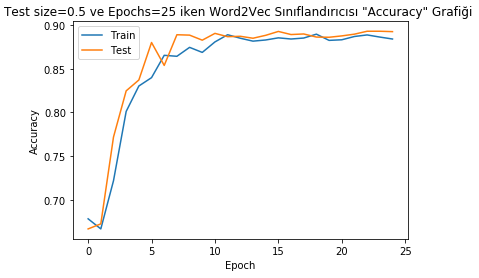
Word2Vec algoritmasına 3000 tweet veri kümesini sınıflandırmak amacıyla verilen test size=0.20, test size=0.50, test size=0.70 ve epochs=25 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.



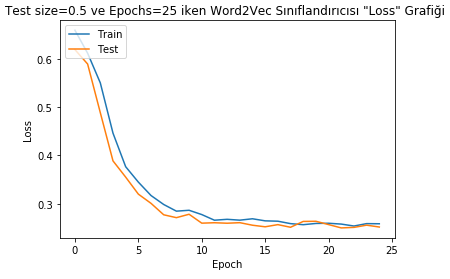
Resim 4.23 Word2Vec sınıflandırıcısının test size=0.2 ve 3000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



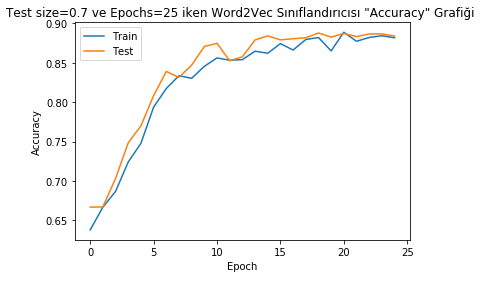
Resim 4.24 Word2Vec sınıflandırıcısının test size=0.2 ve 3000 tweet veri kümesi için loss grafiğinin çizimi



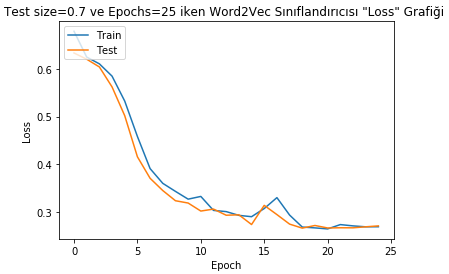
Resim 4.25 Word2Vec sınıflandırıcısının test size=0.5 ve 3000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.26 Word2Vec sınıflandırıcısının test size=0.5 ve 3000 tweet veri kümesi için loss grafiğinin çizimi



Resim 4.27 Word2Vec sınıflandırıcısının test size=0.7 ve 3000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



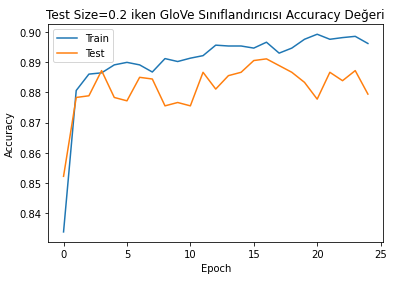
Resim 4.28 Word2Vec sınıflandırıcısının test size=0.5 ve 3000 tweet veri kümesi için loss grafiğinin çizimi

|  |  |
| --- | --- |
| Eğitim Yüzdesi | Word2Vec |
| **TS30** | 0.88 |
| **TS50** | 0.89 |
| **TS80** | 0.88 |

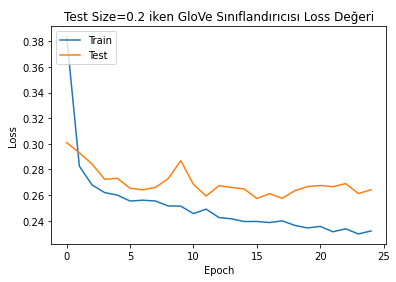
Tablo 12: Word2Vec sınıflandırıcısı için 3000 tweet veri kümesinin tüm accuracy (doğruluk) değerlerinin karşılaştırılması

***4. GloVe Sınıflandırıcısından Elde Edilen Bulgular***

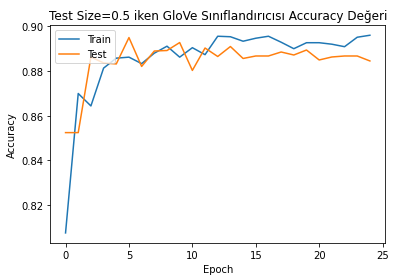
GloVe algoritmasına 3000 tweet veri kümesini sınıflandırmak amacıyla verilen test size=0.20, test size=0.50, test size=0.70 ve epochs=25 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.



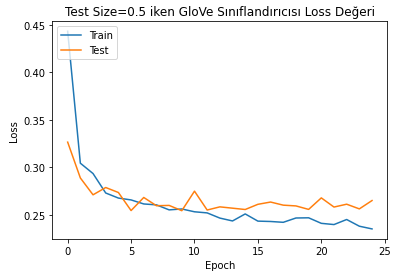
Resim 4.30 GloVe sınıflandırıcısının test size=0.2 ve 3000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



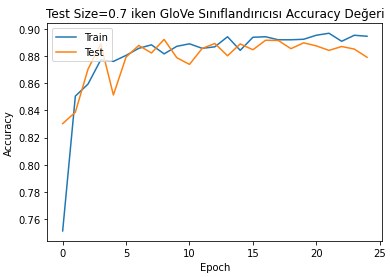
Resim 4.31 GloVe sınıflandırıcısının test size=0.2 ve 3000 tweet veri kümesi için loss grafiğinin çizimi



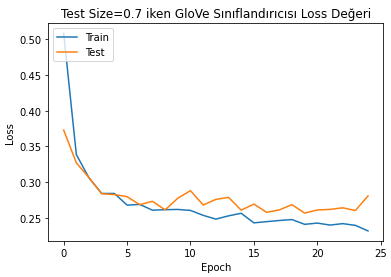
Resim 4.32 GloVe sınıflandırıcısının test size=0.5 ve 3000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.33 GloVe sınıflandırıcısının test size=0.5 ve 3000 tweet veri kümesi için loss grafiğinin çizimi



Resim 4.34 GloVe sınıflandırıcısının test size=0.7 ve 3000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.35 GloVe sınıflandırıcısının test size=0.7 ve 3000 tweet veri kümesi için loss grafiğinin çizimi

|  |  |
| --- | --- |
| Eğitim Yüzdesi | GloVe |
| **TS30** | 0.87 |
| **TS50** | 0.88 |
| **TS80** | 0.87 |

Tablo 13: GloVe sınıflandırıcısı için 3000 tweet veri kümesinin tüm accuracy (doğruluk) değerlerinin karşılaştırılması

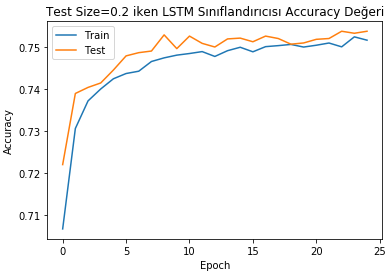
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Eğitim  **Yüzdesi** | **Sınıflandırıcıların Farklı Eğitim Yüzdelerindeki Accuracy (Doğruluk) Değerleri** | | | |
| SVM | LSTM | Word2Vec | GloVe |
| **TS30** | 0.54 | 0.83 | 0.88 | 0.87 |
| **TS50** | 0.57 | 0.87 | 0.89 | 0.88 |
| **TS80** | 0.57 | 0.88 | 0.88 | 0.87 |

Tablo 14: Projede kullanılan tüm sınıflandırıcıların 3000 tweet veri kümesi için accuracy (doğruluk) değerlerinin karşılaştırılması

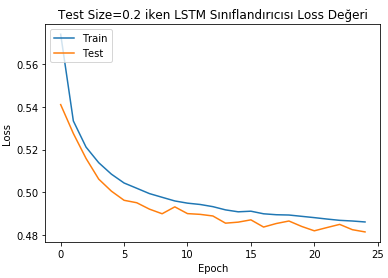
C. 17000 Tweet Veri Kümesi İçin Bulgular

***1. LSTM Sınıflandırıcısından Elde Edilen Bulgular***

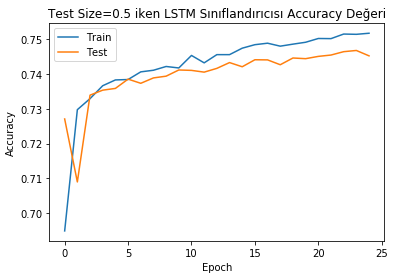
LSTM algoritmasına 17000 tweet veri kümesini sınıflandırmak amacıyla verilen test size=0.20, test size=0.50, test size=0.70 ve epochs=25 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.



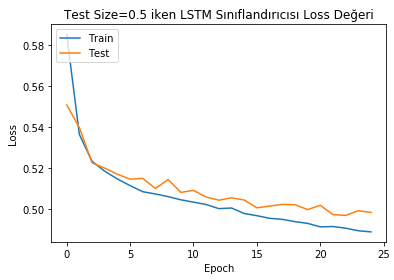
Resim 4.36 LSTM sınıflandırıcısının test size=0.2 ve 17000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



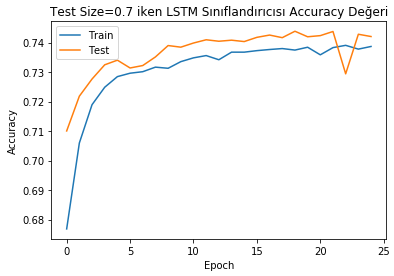
Resim 4.37 LSTM sınıflandırıcısının test size=0.2 ve 17000 tweet veri kümesi için loss grafiğinin çizimi



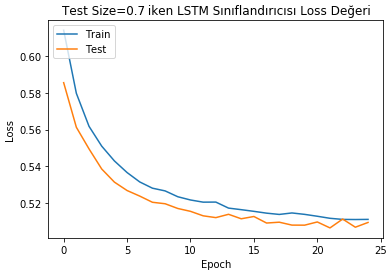
Resim 4.38 LSTM sınıflandırıcısının test size=0.5 ve 17000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.39 LSTM sınıflandırıcısının test size=0.5 ve 17000 tweet veri kümesi için loss grafiğinin çizimi



Resim 4.40 LSTM sınıflandırıcısının test size=0.7 ve 17000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



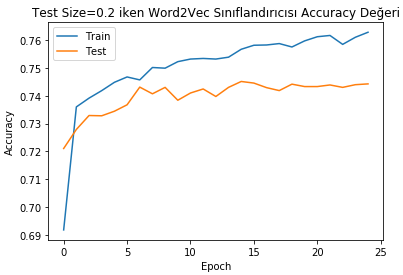
Resim 4.41 LSTM sınıflandırıcısının test size=0.7 ve 17000 tweet veri kümesi için loss grafiğinin çizimi

|  |  |
| --- | --- |
| Eğitim Yüzdesi | LSTM |
| **TS30** | 0.74 |
| **TS50** | 0.74 |
| **TS80** | 0.75 |

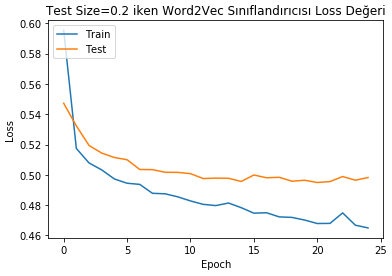
Tablo 15: LSTM sınıflandırıcısı için 17000 tweet veri kümesinin tüm accuracy (doğruluk) değerlerinin karşılaştırılması

***2. Word2Vec Sınıflandırıcısından Elde Edilen Bulgular***

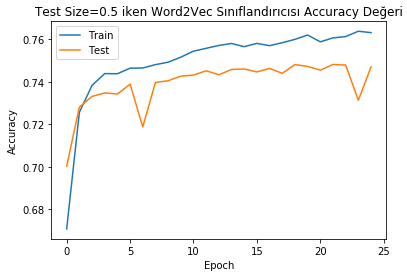
Word2Vec algoritmasına 17000 tweet veri kümesini sınıflandırmak amacıyla verilen test size=0.20, test size=0.50, test size=0.70 ve epochs=25 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.



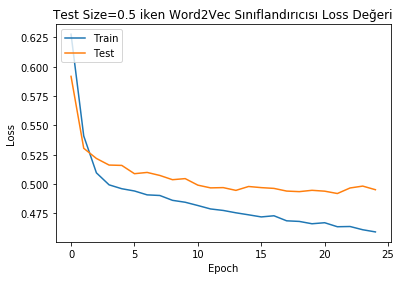
Resim 4.42 Word2Vec sınıflandırıcısının test size=0.2 ve 17000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



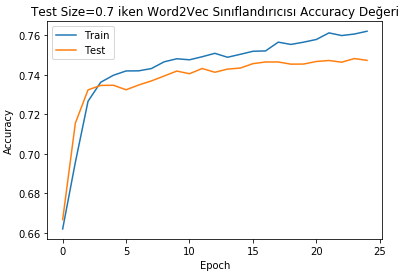
Resim 4.43 Word2Vec sınıflandırıcısının test size=0.2 ve 17000 tweet veri kümesi için loss grafiğinin çizimi



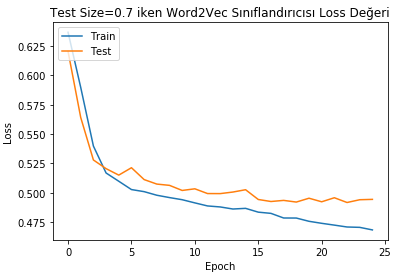
Resim 4.44 Word2Vec sınıflandırıcısının test size=0.5 ve 17000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.45 Word2Vec sınıflandırıcısının test size=0.5 ve 17000 tweet veri kümesi için loss grafiğinin çizimi



Resim 4.46 Word2Vec sınıflandırıcısının test size=0.7 ve 17000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi



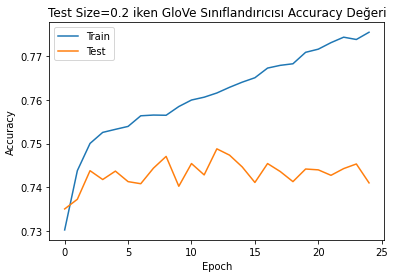
Resim 4.47 Word2Vec sınıflandırıcısının test size=0.7 ve 17000 tweet veri kümesi için loss grafiğinin çizimi

|  |  |
| --- | --- |
| Eğitim Yüzdesi | Word2Vec |
| **TS30** | 0.74 |
| **TS50** | 0.74 |
| **TS80** | 0.74 |

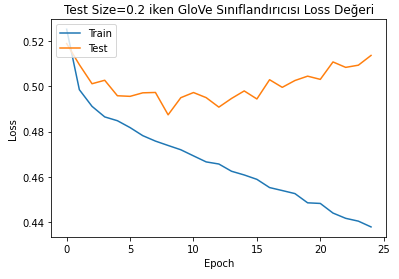
Tablo 16: Word2Vec sınıflandırıcısı için 17000 tweet veri kümesinin tüm accuracy (doğruluk) değerlerinin karşılaştırılması

***3. GloVe Sınıflandırıcısından Elde Edilen Bulgular***

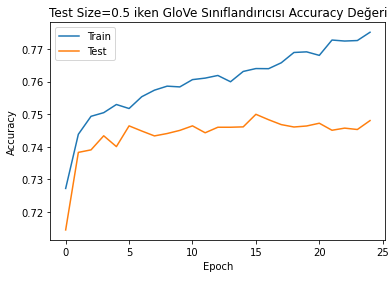
GloVe algoritmasına 17000 tweet veri kümesini sınıflandırmak amacıyla verilen test size=0.20, test size=0.50, test size=0.70 ve epochs=25 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.

******

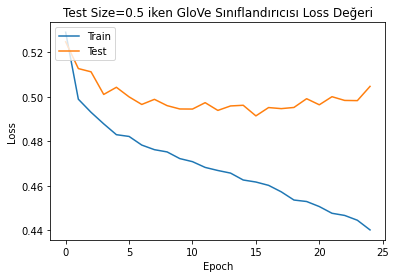
Resim 4.48 GloVe sınıflandırıcısının test size=0.2 ve 17000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi

******

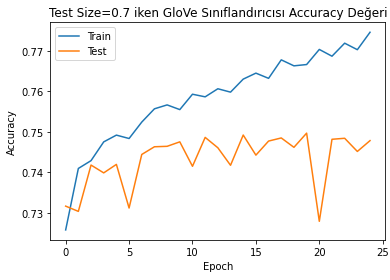
Resim 4.49 GloVe sınıflandırıcısının test size=0.2 ve 17000 tweet veri kümesi için loss grafiğinin çizimi

******

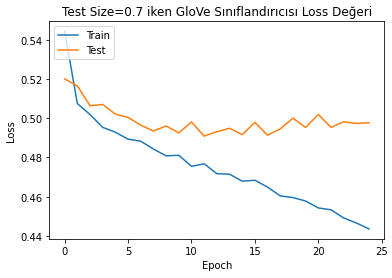
Resim 4.50 GloVe sınıflandırıcısının test size=0.5 ve 17000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi

******

Resim 4.51 GloVe sınıflandırıcısının test size=0.5 ve 17000 tweet veri kümesi için loss grafiğinin çizimi

******

Resim 4.52 GloVe sınıflandırıcısının test size=0.7 ve 17000 tweet veri kümesi için accuracy (doğruluk) grafiğinin çizimi

******

Resim 4.53 GloVe sınıflandırıcısının test size=0.7 ve 17000 tweet veri kümesi için loss grafiğinin çizimi

|  |  |
| --- | --- |
| Eğitim Yüzdesi | GloVe |
| **TS30** | 0.74 |
| **TS50** | 0.74 |
| **TS80** | 0.74 |

Tablo 17: GloVe sınıflandırıcısı için 17000 tweet veri kümesinin tüm accuracy (doğruluk) değerlerinin karşılaştırılması

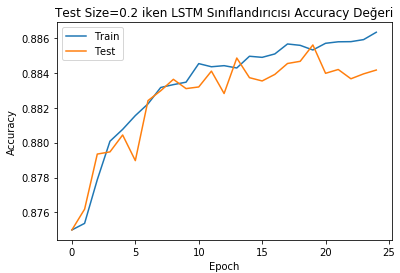
| Eğitim  Yüzdesi | **Sınıflandırıcıların Farklı Eğitim Yüzdelerindeki Accuracy (Doğruluk) Değerleri** | | |
| --- | --- | --- | --- |
| LSTM | Word2Vec | GloVe |
| **TS30** | 0.74 | 0.74 | 0.74 |
| **TS50** | 0.74 | 0.74 | 0.74 |
| **TS80** | 0.75 | 0.74 | 0.74 |

Tablo 18: Projede kullanılan tüm sınıflandırıcıların 17000 tweet veri kümesi için accuracy (doğruluk) değerlerinin karşılaştırılması

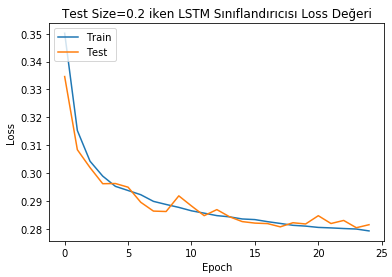
D. Aahaber Veri Kümesi İçin Bulgular

***1. LSTM Sınıflandırıcısından Elde Edilen Bulgular***

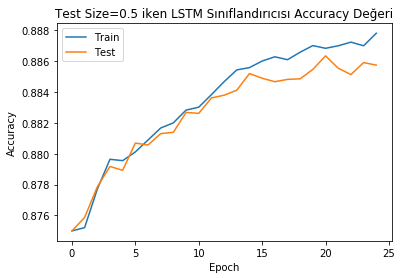
LSTM algoritmasına Aahaber veri kümesini sınıflandırmak amacıyla verilen test size=0.20, test size=0.50, test size=0.70 ve epochs=25 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.



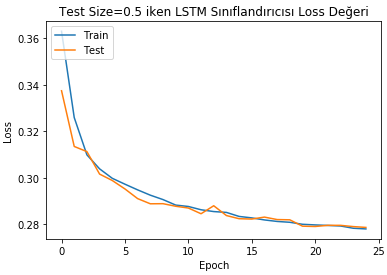
Resim 4.54 LSTM sınıflandırıcısının test size=0.2 ve Aahaber veri kümesi için accuracy (doğruluk) grafiğinin çizimi



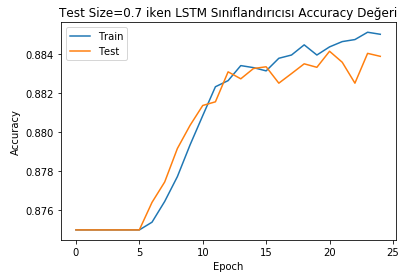
Resim 4.55 LSTM sınıflandırıcısının test size=0.2 ve Aahaber veri kümesi için loss grafiğinin çizimi



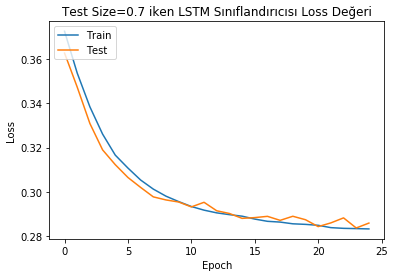
Resim 4.56 LSTM sınıflandırıcısının test size=0.5 ve Aahaber veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.57 LSTM sınıflandırıcısının test size=0.5 ve Aahaber veri kümesi için loss grafiğinin çizimi



Resim 4.58 LSTM sınıflandırıcısının test size=0.7 ve Aahaber veri kümesi için accuracy (doğruluk) grafiğinin çizimi



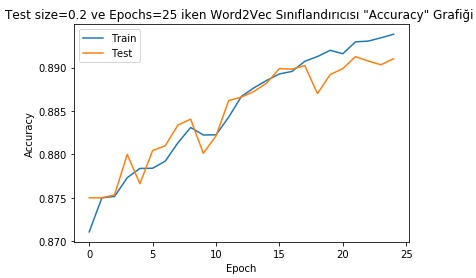
Resim 4.59 LSTM sınıflandırıcısının test size=0.7 ve Aahaber veri kümesi için loss grafiğinin çizimi

|  |  |
| --- | --- |
| Eğitim Yüzdesi | LSTM |
| **TS30** | 0.88 |
| **TS50** | 0.88 |
| **TS80** | 0.88 |

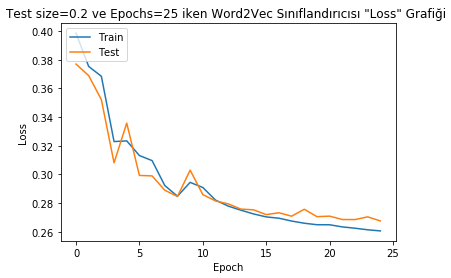
Tablo 19: LSTM sınıflandırıcısı için Aahaber veri kümesinin tüm accuracy (doğruluk) değerlerinin karşılaştırılması

***2. Word2Vec Sınıflandırıcısından Elde Edilen Bulgular***

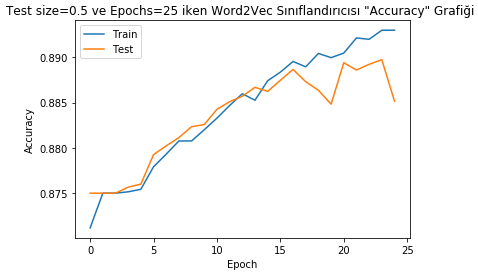
Word2Vec algoritmasına Aahaber veri kümesini sınıflandırmak amacıyla verilen test size=0.20, test size=0.50, test size=0.70 ve epochs=25 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.



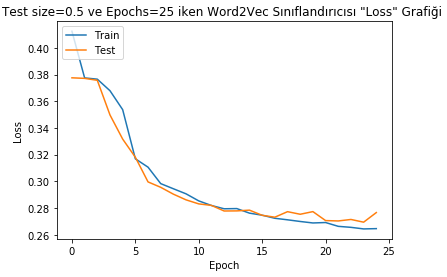
Resim 4.60 Word2Vec sınıflandırıcısının test size=0.2 ve Aahaber veri kümesi için accuracy (doğruluk) grafiğinin çizimi



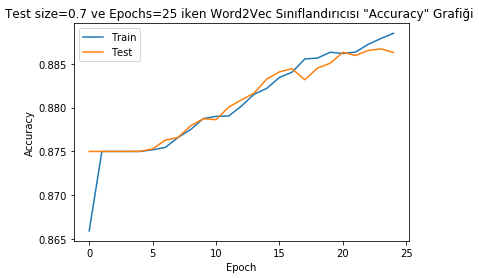
Resim 4.61 Word2Vec sınıflandırıcısının test size=0.2 ve Aahaber veri kümesi için loss grafiğinin çizimi



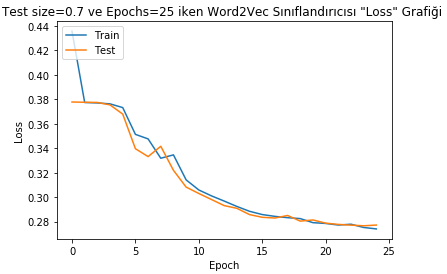
Resim 4.62 Word2Vec sınıflandırıcısının test size=0.5 ve Aahaber veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.63 Word2Vec sınıflandırıcısının test size=0.5 ve Aahaber veri kümesi için loss grafiğinin çizimi



Resim 4.64 Word2Vec sınıflandırıcısının test size=0.7 ve Aahaber veri kümesi için accuracy (doğruluk) grafiğinin çizimi



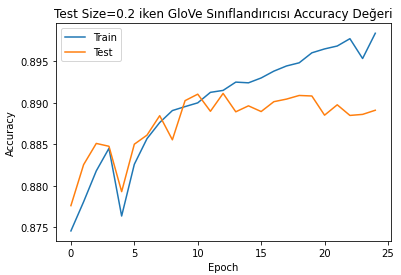
Resim 4.65 Word2Vec sınıflandırıcısının test size=0.7 ve Aahaber veri kümesi için loss grafiğinin çizimi

|  |  |
| --- | --- |
| Eğitim Yüzdesi | Word2Vec |
| **TS30** | 0.88 |
| **TS50** | 0.88 |
| **TS80** | 0.89 |

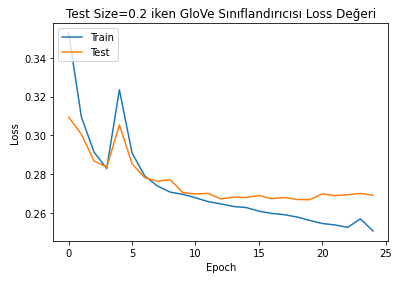
Tablo 20: Word2Vec sınıflandırıcısı için Aahaber veri kümesinin tüm accuracy (doğruluk) değerlerinin karşılaştırılması

***3. GloVe Sınıflandırıcısından Elde Edilen Bulgular***

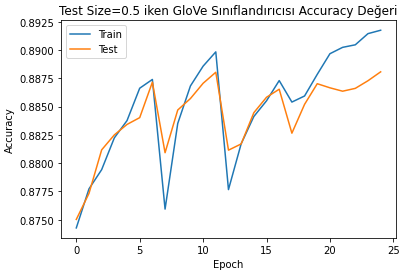
GloVe algoritmasına Aahaber veri kümesini sınıflandırmak amacıyla verilen test size=0.20, test size=0.50, test size=0.70 ve epochs=25 iken elde edilen bulgulara aşağıdaki görsellerde yer verilmiştir.



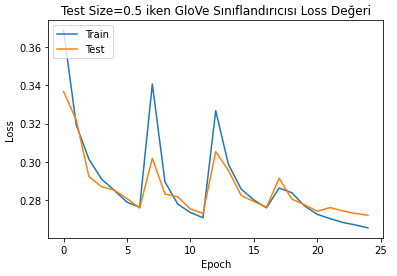
Resim 4.66 GloVe sınıflandırıcısının test size=0.2 ve Aahaber veri kümesi için accuracy (doğruluk) grafiğinin çizimi



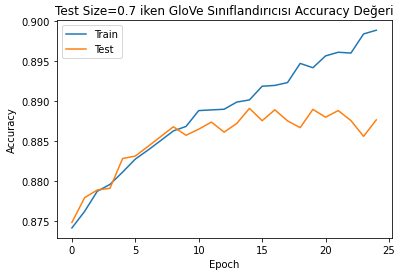
Resim 4.67 GloVe sınıflandırıcısının test size=0.2 ve Aahaber veri kümesi için loss grafiğinin çizimi



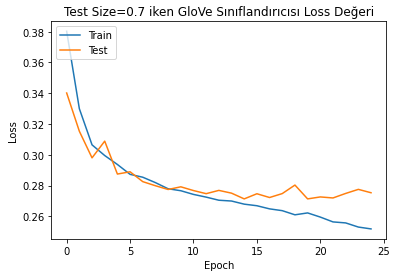
Resim 4.68 GloVe sınıflandırıcısının test size=0.5 ve Aahaber veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.69 GloVe sınıflandırıcısının test size=0.5 ve Aahaber veri kümesi için loss grafiğinin çizimi



Resim 4.70 GloVe sınıflandırıcısının test size=0.7 ve Aahaber veri kümesi için accuracy (doğruluk) grafiğinin çizimi



Resim 4.71 GloVe sınıflandırıcısının test size=0.7 ve Aahaber veri kümesi için loss grafiğinin çizimi

|  |  |
| --- | --- |
| Eğitim Yüzdesi | GloVe |
| **TS30** | 0.88 |
| **TS50** | 0.88 |
| **TS80** | 0.88 |

Tablo 21: GloVe sınıflandırıcısı için Aahaber veri kümesinin tüm accuracy (doğruluk) değerlerinin karşılaştırılması

| Eğitim  Yüzdesi | **Sınıflandırıcıların Farklı Eğitim Yüzdelerindeki Accuracy (Doğruluk) Değerleri** | | |
| --- | --- | --- | --- |
| LSTM | Word2Vec | GloVe |
| **TS30** | 0.88 | 0.88 | 0.88 |
| **TS50** | 0.88 | 0.88 | 0.88 |
| **TS80** | 0.88 | 0.89 | 0.88 |

Tablo 22: Projede kullanılan tüm sınıflandırıcıların Aahaber veri kümesi için accuracy (doğruluk) değerlerinin karşılaştırılması

E. Sınıflandırıcıların Tüm Veri Kümeleri İçin Accuracy (Doğruluk) Değerleri

Aşağıda yer alan tablolarda 3000 tweet, 17000 tweet ve aahaber veri kümeleri için her bir sınıflandırıcının accuracy (doğruluk) değerlerine yer verilmiştir.

| Eğitim  Yüzdesi | **LSTM Sınıflandırıcısının Farklı Veri Kümelerindeki Accuracy (Doğruluk) Değerleri** | | |
| --- | --- | --- | --- |
| 3000 Tweet | 17000 Tweet | Aahaber |
| **TS30** | 0.83 | 0.74 | 0.88 |
| **TS50** | 0.87 | 0.74 | 0.88 |
| **TS80** | 0.88 | 0.75 | 0.88 |

Tablo 23: Projede kullanılan tüm veri kümeleri için LSTM sınıflandırıcısının accuracy (doğruluk) değerlerinin karşılaştırılması

| Eğitim  Yüzdesi | **Word2Vec Sınıflandırıcısının Farklı Veri Kümelerindeki Accuracy (Doğruluk) Değerleri** | | |
| --- | --- | --- | --- |
| 3000 Tweet | 17000 Tweet | Aahaber |
| **TS30** | 0.88 | 0.74 | 0.88 |
| **TS50** | 0.89 | 0.74 | 0.88 |
| **TS80** | 0.88 | 0.74 | 0.89 |

Tablo 24: Projede kullanılan tüm veri kümeleri için Word2Vec sınıflandırıcısının accuracy (doğruluk) değerlerinin karşılaştırılması

| Eğitim  Yüzdesi | **GloVe Sınıflandırıcısının Farklı Veri Kümelerindeki Accuracy (Doğruluk) Değerleri** | | |
| --- | --- | --- | --- |
| 3000 Tweet | 17000 Tweet | Aahaber |
| **TS30** | 0.87 | 0.74 | 0.88 |
| **TS50** | 0.88 | 0.74 | 0.88 |
| **TS80** | 0.87 | 0.74 | 0.88 |

Tablo 25: Projede kullanılan tüm veri kümeleri için GloVe sınıflandırıcısının accuracy (doğruluk) değerlerinin karşılaştırılması

##### KAYNAKLAR

1. Alpaydın E. Yapay Öğrenme. 3. Baskı. İstanbul, Türkiye, Boğaziçi Üniversitesi Yayınevi, 2017
2. Mitchell T. Machine Learning. New York, USA, McGraw Hill, 1997.
3. Support Vector Machine konu anlatımı ve görselleri:

Yrd. Doç. Dr. Umut ORHAN ders notlarından alıntı yapılmıştır.

1. <https://www.veribilimiokulu.com/word2vec/>
2. <https://medium.com/@muhammedbuyukkinaci/word2vec-nedir-t%C3%BCrk%C3%A7e-f0cfab20d3ae>
3. <https://veribilimcisi.com/2017/09/26/uzun-kisa-sureli-bellek-long-short-term-memory/>
4. <http://dprogrammer.org/rnn-lstm-gru>
5. <https://medium.com/codable/word2vec-fasttext-glove-d4402fa8cce0>
6. <http://dspace.kocaeli.edu.tr:8080/xmlui/bitstream/handle/11493/1032/521130.pdf?sequence=1>
7. Veri kümesi referansı:

" Eğiticili ve Geleneksel Terim Ağırlıklandırma Yöntemleriyle Duygu Analizi", Mahmut Çetin, M.Fatih Amasyalı, SIU 2013, KKTC

“Duygudurum Analizinde Kelimeler, Anlamlar, Karakterler”, Mehmet Fatih Amasyali, Hakan Taşköprü, Kübra Çalışkan, Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu (ASYU), 4-6 Ekim 2018, Adana

1. Karşılaştırma Ölçütleri Bölümünde yararlanılan kaynak:

<http://oaji.net/articles/2017/1486-1538481206.pdf>

1. Karşılaştırma Ölçütleri Resim 4.1 kaynak:

<https://medium.com/@sengul_krdrl/hata-matrisini-anlamak-7035b7921c0f>