Distributed City Sentiment
Şehir Bazlı Dağıtık Duygu Analizi

Rabia Arslan, Muhammed Said Zengin
TOBB Ekonomi ve Teknoloji Üniversitesi

***Özetçe* —Sosyal medya verisi üzerinde çeşitli yapay zeka çalışmaları yapılmaktadır. Sosyal medyada artan paylaşım miktarıyla birlikte işlenmesi gereken veri miktarında büyük artış olmaktadır. Bu sebeple yapay zeka tahmin hızlarının da artması gerekmektedir. Bu çalışmada büyük sosyal medya verisi üzerinde dağıtık tahminleme ile standart tahminleme karşılaştırılmıştır. Bu karşılaştırma için Türkiye'nin 81 ilinden toplam 41.000 Twitter kullanıcısının 16 milyon tweeti çekilmiştir. Bu veri üzerinde duygu analizi çalışması yapılmıştır. Duygu analizi için veri etiketlenmiş ve önceden eğitilmiş BERT modeli ince ayar (fine tune) yapılarak sınıflandırma modeli oluşturulmuştur. Karşılaştırma amacıyla önce normal tahminleme, ardından dağıtık yapıyla tahminleme yapılmıştır. Sonucunda ise 81 il için duygu analiz skoları çıkarılmıştır. 16 milyon veriye, duygu analiz modeline, çalışma kodlarına ve mutluluk analiz skolarına https://github.com/msaidzengin/distributedCitySentiment adresinden ulaşabilirsiniz.**

***Anahtar Kelimeler*—Büyük Veri, Dağıtık Veri İşleme, Duygu Analizi, Dil Modelleri**

***Abstract*—Various artificial intelligence studies are carried out on social media data. With the increasing amount of sharing on social media, there is a great increase in the amount of data that needs to be processed. For this reason, artificial intelligence prediction speeds should also increase. In this study, distributed prediction and standard prediction were compared on large social media data. For this comparison, 16 million tweets of a total of 41,000 Twitter users from 81 provinces of Turkey were taken. A sentiment analysis study was conducted on this data. For sentiment analysis, the data were labeled and the classification model was created by fine-tuning the pre-trained BERT model. For comparison purposes, first normal prediction and then distributed prediction were made. As a result, sentiment analysis scores were obtained for 81 provinces. You can access 16 million data, sentiment analysis model, codes and sentiment analysis scores at given url.**

***Keywords*—Big Data, Distributed Data Processing, Sentiment Analysis, Language Models**

I. Giriş

Günümüzde sosyal medya kullanımı oldukça yaygınlaşmakta, pandemi sebebiyle insanların sosyal medyada vakit geçirme süresi ve paylaşım sayıları gittikçe artmaktadır. Bu sebeple sosyal medyanın büyük veri kaynağı olarak kullanımı yaygınlaşmaktadır. Pandeminin sosyal medya kullanıcıları üzerinde olumsuz etkileri de olmaktadır. Örneğin sosyal hayata katılamayan insanlar psikolojik olarak zor zamanlar geçirmektedir. Bu sebeple olumsuz düşünceler sosyal medyada gittikçe artmaktadır.

Bu çalışmada sosyal medya kullanıcıların şehir bazında duygu analizleri yapılmıştır. Duygu analizi, verilen bir metni pozitif veya negatif olarak sınıflandırmaya yarayan bir sınıflandırma görevidir. Twitter üzerinde duygu analizi yapan çalışmalar [1], [2] bulunmaktadır. Türkçe duygu analizi çalışmalarında SVM [3], LSTM [4], BERT [5] gibi yöntemler kullanılmıştır. Sınıflandırma doğal dil işlemenin en temel konularından biridir. Metin sınıflandırma çalışmalarında [6] BERT [7] oldukça başarılı sonuçlar vermektedir. Bu sebeple önceden eğitilmiş BERT modeli ince ayar yapılarak bir sınıflandırma çalışması olan duygu analiz modeli hazırlanmıştır.

Sosyal medya üzerinde gerçek zamanlı tahmin yapmak oldukça önemlidir. Bir olayın tespiti gibi anlık analiz edilmesi gereken konular olabilir. Bu sebeple her gün milyonlarca veri üretilen sosyal medyada gerçek zamanlı tahmin yapmak zor bir konudur. Bu makalede büyük bir verinin dağıtık olarak tahmin edilmesi incelenmektedir. Normal şartlarda çok uzun sürecek tahminleme işlemini Apache Spark[1] kullanarak hızlandırılmıştır. Bu alanda tahmin hızını arttırmaya çalışan [8] gibi çalışmalar bulunmaktadır. Bu hızlandırma için tahmin sayısını sabit tutarak dağıtım sayısını arttırma denemeleri yapılmıştır.

Duygu analizi çalışması için veri toplanmış, veri etiketlenmiş ve önceden eğitilmiş BERT modeline ince ayar yapılarak model oluşturulmuştur. Ardından Twitter üzerinde 81 ilden toplam 41.000 kullanıcının 16 milyon tweeti çekilmiştir. Duygu analiz modeli ile, her il için özel duygu analiz skoları çıkarılmıştır. Bu çalışma yapılırken önce teker teker tahmin yapılmış, ardından Spark kullanılarak dağıtık tahminleme yapılmıştır. Burada ölçülen başarı duygu analizi değil, tahminleme hızının artış başarısıdır.

II. Problem Tanımı

Bu çalışmanın amacı, günümüzde yaygın olarak kullanılan yapay zeka algoritmalarının dağıtık veri teknolojileri kullanılarak hızlandırılmasıdır. Yapay zeka algoritmaları öncelikle modelin oluşturulması ve tahmin yapılması ile çalışmaktadır. Model oluşturma kısmında, standart kütüphaneler GPU kullanarak işlemi hızlandırırken, tahminleme sırasında bu kütüphaneler CPU ile tahminlediği için daha yavaş çalışmaktadır. Çözmeye çalıştığımız problem, gerçek zamanlı stream veri üzerinde tahminleme hızının arttırılmasıdır. Ancak bu çalışmada hızlandırma yöntemi stream veri üzerinde değil, 16 milyonluk sabit bir veri üzerinde yapılacaktır.

III. Veri Seti

Bu bölümde kullanılan veri setleri anlatılacaktır. Veri çekme, veri ön işleme ve veri istatistikleri açıklanacaktır.

[1] https://spark.apache.org/

A. Veri Çekme

Şehir bazlı sosyal medya verileri Twitter[2] üzerinden toplanmıştır. Twitter'da konum bazlı veri çekmek için iki farklı yöntem izlenebilir. İlk yöntem, eğer kullanıcı atılan tweet için konum paylaşımını açtıysa verideki bu alan kullanılarak konum bazlı veri çekilebilmektedir. Fakat bu şekilde bulunan tweet sayısı oldukça az olduğu için yeterli veri toplanamaz. Bu sebeple ikinci yöntem uygulanmıştır. İkinci yöntem, eğer kullanıcı profiline bir konum bilgisi girdiyse, bu bilgi kullanılarak kullanıcı adları ve tweet'leri toplanabilir. Eğer kullanıcı bir ilçe adı girerse, bulunduğu ilin ismi kullanıldığında bu kullanıcı da gelmektedir. Ayrıca koordinat kullanılarak ve bir daire yarıçapı uzunluğu verilerek de veri çekilmektedir.

Veri çekmek için açık kaynaklı bir Twitter veri kütüphanesi olan Twint[3] kullanılmıştır. Bu kütüphane ile öncelikle 81 il için her şehirden 5000 tweet olmak üzere toplam 405.000 tweet çekilmiştir. Bu tweetler 81 satırdan oluşan bir bash script ile çekilmiştir. Ankara şehri için örnek bash script: *"twint –near "Ankara" –limit 5000 -o Ankara.csv –csv"* şeklindedir.

Ardından bu tweetleri paylaşan kullanıcı listesi oluşturulmuştur. Toplam 104.143 kullanıcı bulunmuştur. Her şehir için ortalama 1285 kullanıcı olmaktadır.

Ardından her şehir için kullanıcı sayısı 500 olarak filtrelenmiştir. Bu 500 kullanıcının son paylaştığı 500 tweet çekilmiştir. Her şehir için 500 satırlık bir bash script çalıştırılmıştır. Bir kullanıcının 500 tweetini çeken bash script: *"twint -u username –timeline –limit 500 –output username.csv –csv"* şeklindedir.

Bunun sonucunda toplam 40.500 kullanıcı ve 16.116.035 tweet elde edilmiştir. Her kullanıcının 500 tweeti olmadığı için hesaplanandan düşük bir sayı çıkmıştır. Bu tweetlerin içerisinde tweetin atıldığı koordinat bilgisini tutan yalnızca 64.044 tweet ve 1357 kullanıcı bulunmaktadır. Koordinat bilgisi yeteri kadar olmadığı için bu veri kullanılmamıştır.

B. Veri Ön İşleme

Twint kütüphanesi ile tweetlerin tüm alanları çekilmiştir. Her tweetin 36 farklı bilgi alanı bulunmaktadır. Toplam veri bu alanlarla birlikte 7.6 GB boyutundadır. Bu çalışmada sadece metin verisi kullanıldığı için kalan 35 field silinmiştir.

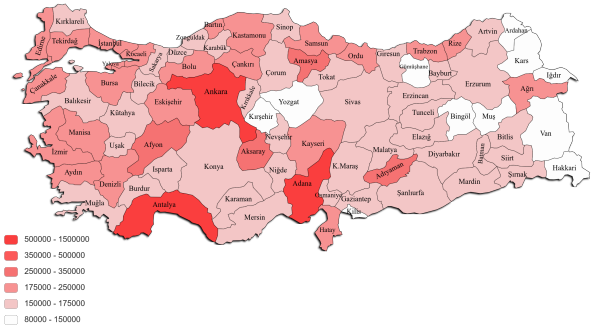
Metin verisinde tab, new line gibi tüm boşluklar space karakteri ile değiştirilmiştir. Veriden mention, url ve hashtagler silinmiştir. Ardından her il için bir adet txt dosyası olmak üzere 81 adet txt dosyasına her bir satıra bir tweet gelecek şekilde metinler atılmıştır. Toplam veri boyutu 81 il için 1.5 GB olmuştur.

C. Veri İstatistiği

Her şehir için veri sayısının ısı haritası Şekil 1'de bulunmaktadır. En az tweet 88.390 adet Ardahan'da, en çok tweet 1.054.767 adet Ankara'da bulunmaktadır. Toplam 16.116.035 tweet olduğu için bir şehirdeki ortalama tweet sayısı 198.964 olmuştur. Tüm verilere ve veri istatistiklerine paylaşılan Github adresinden erişebilirsiniz.

[2]https://twitter.com/

[3]https://github.com/twintproject/twint



Şekil 1: Şehir Bazlı Veri Sayılarının Isı Haritası

IV. Duygu Analizi

Bu bölümde duygu analizi modelinin detayları açıklanacaktır.

A. Veri Etiketleme

Duygu analizi için Twitter üzerinden gülme, kahkaha atma, üzülme ve ağlama gibi emojiler kullanılarak 5000 adet veri çekilmiştir. Bu veriler 2 kişi tarafından pozitif veya negatif olarak etiketlenmiştir. Öncelikle ilk kişi etiketlemeyi yapmıştır. Ardından ikinci kişi etiketlerin üzerinden geçmiştir. İkinci kişinin farklı düşündüğü bir veri bulunuyorsa bu etiketi kaldırmıştır. Bu sayede iki kişinin de ortak düşündüğü veriler bulunmaktadır. Toplam 1000 veri pozitif, 1000 veri negatif olarak etiketlenmiştir.

B. Duygu Analiz Modeli Oluşturma

Duygu analizi modeli oluşturmak için TensorFlow Keras[4] kullanımını kolaylaştıran ktrain [9] kütüphanesi kullanılmıştır. Bu kütüphane ile Huggingface[5] üzerinde açık kaynak olarak paylaşılan BERTurk[6] modeli ince ayar yapılmıştır. BERTurk modeli eğitilirken 35 GB ve 44 milyon token boyutundaki bir veri kullanılmıştır. BERTurk Türkçe için özel olarak hazırlanmış bir BERT modelidir. BERTurk modeli ince ayar yapılırken, 2000 adet duygu analizi için etiketlenmiş veri kullanılmıştır. Learning rate olarak 5e-5, batch size olarak 6 kullanılmıştır. 3 epoch eğitilmiştir. Tüm random seed'ler 42 olarak ayarlanmıştır. Train verisi %90, test verisi %10 olarak bölünmüştür. Sonucunda ise test verisi üzerinde %93.7 F1 skoru elde edilmiştir. Bu çalışmada duygu analizi üzerinde durulmadığı için bu model detaylarına değinilmemiştir. Duygu analiz modeline paylaşılan Github adresinden erişilmektedir.

V. Deneysel Kurulum

Bu bölümde yapılan deneylerden bahsedilecektir. Tüm deneyler aynı sunucuda yapılmıştır. Sunucuda 32 GB RAM, Intel Core i7 11800 2.70 GHz x16 işlemci, 64 bit Ubuntu 20.04 işletim sistemi bulunmaktadır.

[4]https://keras.io/

[5]https://huggingface.co

[6]https://huggingface.co/dbmdz/bert-base-turkish-cased

A. Baseline

Baseline için bahsedilen sunucuya 1.5 GB boyutundaki 16.1 milyon veri ve 550 MB boyutundaki duygu analizi modeli atılmıştır. Python kullanılarak, her şehrin veri dosyası teker teker okunmuş, ardından döngü kullanılarak her veri için tahmin yapılmıştır. Çalışma süresi hesaplanırken dosya okuma ve model yükleme süreleri kayda alınmamıştır. Tahmin metodu çalışmadan önce süre başlatılmış ve tahminler bittiği zaman süre durdurulmuştur.

B. Dağıtık Veri / Spark

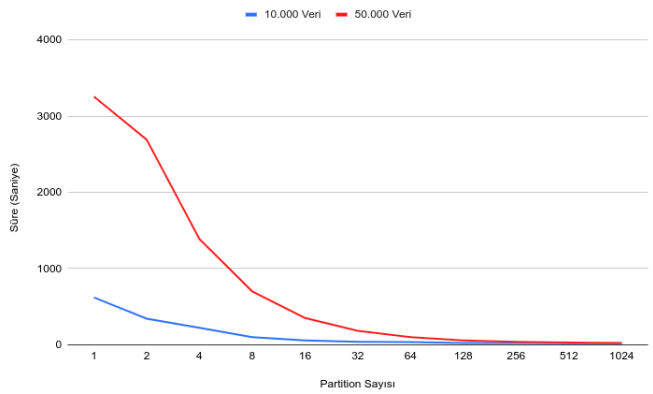
Dağıtık hesaplama yapmak için PySpark[7] kullanılmıştır. Baseline için kullanılan aynı sunucu, aynı veri ve aynı duygu analizi modeli kullanılmıştır. Tek fark olarak tahminleme dağıtık olarak yapılmıştır. Burada yapılan ilk deney, tüm veri üzerindeki maksimum hızlanma yüzdesini bulma olmuştur. İkinci deneyde ise N adet veri üzerinde, farklı dağıtma (partition) sayılarındaki hızlanma oranları incelenmiştir.

VI. Deneysel Sonuçlar

Bu bölümde yapılan deney sonuçları 3 farklı yaklaşımda incelenecektir. Bu yaklaşımlar, partition sayısına bağlı hızlanma değişimi, tüm veri üzerindeki maksimum hız artış oranı ve şehirlere bağlı duygu analiz sonuçları olacaktır.

Partition Sayısı	Süre (Saniye)	
	10.000 Veri	50.000 Veri
1	621	3256
2	342	2693
4	223	1386
8	100	701
16	57	352
32	38	182
64	35	99
128	23	58
256	22	39
512	21	28
1024	19	24

TABLO I: Deneysel Sonuçlar



Şekil 2: Deneysel Sonuçlar

[7]https://pypi.org/project/pyspark/

A. Dağıtma Sayısına Bağlı Tahminleme

Bu deney için veri setinden rastgele 10.000 ve 50.000 adet örnek tweet alınmıştır. Bu tweetler kullanılarak partition sayısına bağlı hızlanma incelenmiştir. 10.000 tweet için dağıtmadan yapılan tahminler 10 dakika 21 saniye (621 saniye) sürmüş, 1024 parçaya bölündüğünde ise tahmin süresi 19 saniyeye kadar düşmüştür. Aynı deneyler 50.000 veride yapıldığında ise dağıtık olmayan kod 54 dakika 16 saniye (3256 saniye) sürmüş, 1024 parçaya bölündüğünde 24 saniye sürmüştür. Yapılan tüm deney sonuçları Tablo I'de ve Grafik 2'de bulunmaktadır. Grafikte ve tabloda bulunan partition kısmı 1 olan sonuçlar Spark kullanılmadan baseline kodu ile hesaplanmıştır.

B. Tüm Veri Üzerinde Maksimum Hızlanma

Bu deney için öncelikle tüm veri dağıtık olmadan tahminlenmiştir. Tüm verinin tahmin süresi 12 günü geçmiştir. Saat bazında 303 saat, dakika bazında 18.226 dakika sürmüştür.

Dağıtık tahminleme için her bir şehir iteratif olarak hesaplanmıştır. Bazı sistem kısıtlamaları yüzünden veri tek parça halinde tahminlenememiştir. Tüm veri okunduğu zaman 16 milyon veriyi 1024 parçaya bölüp hesaplama yaparken sunucu kapanmıştır. Daha fazla partition sayısı da sistem sorununa yol açmıştır.

Bu sebeple, her şehir verisi ayrı ayrı okunmuş ve her biri kendi içinde dağıtık hesaplanmıştır. Örnek 8 şehrin veri sayısı ve tahmin süresi Tablo II'de gösterilmiştir.

Şehir	**Tweet Sayısı**	**Tahmin Süresi (Saniye)**
Düzce	171288	30.66
Kütahya	159606	26.23
Kocaeli	184652	28.41
Edirne	189477	27.11
Erzurum	155769	25.43
İstanbul	222141	29.80
Ankara	1054822	86.73

TABLO II: Hız Artışı

81 il için ayrı ayrı dağıtık tahminleme yapıldığında, ortalama tahmin süresi 28.17 saniye, toplam tahmin süresi ise 2282 saniye sürmüştür. Baseline algoritmada toplam 1.093.560 saniye süren tahmin işlemi, dağıtık algoritmada 2282 saniyeye düşürülmüştür. Yani yaklaşık olarak işlem 479 kat daha hızlı hale gelmiştir.

C. Şehirlerin Mutluluk Analizi

Tüm tahminler yapıldıktan sonra şehir bazlı duygu analiz sonuçları çıkarılmıştır. Bu sonuçları almak için her ilden rastgele 10.000 tweet alınmıştır. Bu sonuçlara göre en pozitif şehir Bitlis, en negatif şehir ise Aydın çıkmıştır. Şekil 3'de şehir bazlı duygu analizlerinin ısı haritası görülmektedir. Tüm şehirlerin detaylı sonuçlarına paylaşılan Github adresinden ulaşılabilir.

Duygu analizi sonuçları 16 milyon veriden rastgele seçilmiş 800.000 veriden elde edilmiştir. Veri sayısı arttıkça sosyal medya kullanıcılarının duyguları daha başarılı ifade edilebilir. Bununla birlikte çekilen 16 milyon verinin tarih aralığı daha geniş olursa daha doğru bir duygu analizi yapılabilir. Fakat duygu durumu hızlı bir şekilde değişebildiği için, günlük, haftalık ve aylık hesaplamalar daha doğru olacaktır.

Şekil 3: Şehir Bazlı Duygu Analizi

VII. Sonuç

Sonuç olarak, duygu analizi için veri çekilmiş, veri etiketlenmiş ve bir model eğitilmiştir. Ardından Twitter üzerinden 16 milyon veri çekilmiştir. Bu veriler duygu analizi modeli kullanılarak hem normal bir şekilde hem de dağıtık bir şekilde tahmin edilmiştir. Bu sonuçlara göre dağıtık veri tahminlemenin normal tahminlemeye göre oldukça hızlı çalıştığı gözlemlenmiştir. Ayrıca duygu analizi sonuçlarına göre şehir bazlı mutluluklar ölçülmüştür. Gelecek çalışmalarda dağıtık tahminleme yaparken GPU kullanımı ve Spark kullanımının farkları ele alınabilir. Gerçek zamanlı Twitter stream verisi üzerinde yapay zeka modelleri çalıştırmanın bu sayede kolaylaştığı görülmüştür.

Kaynaklar

[1] Go, Alec, Lei Huang, and Richa Bhayani. "Twitter sentiment analysis." Entropy 17 (2009): 252.

[2] Sarlan, Aliza, Chayanit Nadam, and Shuib Basri. "Twitter sentiment analysis." Proceedings of the 6th International conference on Information Technology and Multimedia. IEEE, 2014.,

[3] Kaya, Mesut, Güven Fidan, and Ismail H. Toroslu. "Sentiment analysis of Turkish political news." 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology. Vol. 1. IEEE, 2012.

[4] Ciftci, Basri, and Mehmet Serkan Apaydin. "A deep learning approach to sentiment analysis in Turkish." 2018 International Conference on Artificial Intelligence and Data Processing (IDAP). IEEE, 2018.

[5] Acikalin, Utku Umur, Benan Bardak, and Mucahid Kutlu. "Turkish Sentiment Analysis Using BERT." 2020 28th Signal Processing and Communications Applications Conference (SIU). IEEE, 2020.

[6] Sun, Chi, et al. "How to fine-tune bert for text classification?." China National Conference on Chinese Computational Linguistics. Springer, Cham, 2019.

[7] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[8] Wang, Kewen, and Mohammad Maifi Hasan Khan. "Performance prediction for apache spark platform." 2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems. IEEE, 2015.

[9] Maiya, Arun S. "ktrain: A low-code library for augmented machine learning." arXiv preprint arXiv:2004.10703 (2020).