Gazi Mühendislik Bilimleri Dergisi
2021, 7(3): 243-252
Araştırma Makalesi/Research Article
https://dergipark.org.tr/gmbd

İşin Olsun Platformu İlanlarında İçerik Kontrolü

Işılay TUNCER[*,a], Şeref KESKİN[b], Mehmet APAYDIN[c]

[a,*] *Kariyer.net A.Ş., Araştırma Geliştirme Departmanı, Veri Sistemleri, İstanbul, Türkiye*
[b] *Kariyer.net A.Ş., Araştırma Geliştirme Departmanı, Veri Sistemleri, İstanbul, Türkiye*
[c] *İzmir Yüksek Teknoloji Enstitüsü., Elektrik Elektronik Mühendisliği Bölümü, İzmir, Türkiye*

MAKALE BİLGİSİ

Alınma: 25.06.2021
Kabul: 10.10.2021

Anahtar Kelimeler:
Dil modelleri,
Bert,
Transformers,
Doğal dil işleme,
Yapay zekâ,
Derin öğrenme

ÖZ

Bu makalede mavi yakalı iş arama ve bulma platformu olan İşin Olsun sitesindeki ilanların içerik yönetimi ve otomatik içerik kontrolü hakkında geliştirilen yaklaşımlar açıklanacaktır. Bu amaçla denetimli makine öğrenmesi modelleri ve Bert transformer mimarisi yöntemleri ile deneyler gerçekleştirilerek sonuçları gözlemlenmiştir. Çalışma sonunda, dizi sınıflandırma için Bert (Bert for sequence classification) kullanılarak ilan içeriklerinin otomatik olarak sınıflandırıldığı bir sistem geliştirilmiş ve bu sistem iş arama platformuna entegre edilmiştir. İşin Olsun ilan metinlerinin kontrolör görevindeki kişiler tarafından uygun veya uygun olmayan içerik şeklinde iki farklı sınıfta etiketlenmesinden başlayarak, veri kümesinin hazırlanma aşamalarından, sınıflandırma modelinin sisteme entegrasyonuna kadar olan çalışmalar bu makalede özetlenmektedir.

https://dx.doi.org/10.30855/gmbd.2021.03.07

Content Control in İşin Olsun Platform Job Texts

ARTICLE INFO

Received: 25.06.2021
Accepted: 10.10.2021

Keywords:
Language models,
Bert,
Tansformers,
Natural language processing,
Artificial intelligence,
Deep learning

ABSTRACT

In this paper, the approaches developed for content management and automatic content control of the job postings on the İşin Olsun website, which is a blue-collar job search and finding platform, will be explained. For this purpose, experiments were carried out with supervised machine learning models and Bert transformer architecture methods, and the results were observed. At the end of the study, a system was developed in which the contents of the job texts are automatically classified using Bert for sequence classification, and this system was integrated into the job search platform. Starting from labeling the job texts in two different classes as appropriate or unsuitable content by the controllers, from the preparation stages of the dataset to the integration of the classification model into the system, the studies are summarized in this paper.

https://dx.doi.org/10.30855/gmbd.2021.03.07

1. GİRİŞ *(INTRODUCTION)*

İşin Olsun platformu 2017'de kurulmuş olan tezgâhtar, garson, kurye gibi mavi yakalı iş ve işçi bulma fonksiyonu icra eden çevrimiçi bir platformdur [1]. Zaman içerisinde bu platforma verilen ilanlar içerisinde uygun olmayan içeriklere sahip ilanların da sistem üzerinden yayına alındığı tespit edilmiştir. Uygun olmayan metinler, müstehcen resim veya dolandırıcılık amaçlı içerik gibi istenmeyen senaryoların yer aldığı ilanlar olabilmektedir.

İlanların uygun ya da uygunsuz olduğunun tespiti hali hazırda İşin Olsun platformu içerisinde manuel işlemlerle gerçekleşmektedir. Bu ilanlar bir kontrol mekanizmasının önüne düşmekte, sistem içerisinde kontrolör görevindeki kişi de ilanı değerlendirerek uygun görmesi halinde ilanı sisteme otomatik olarak

*Sorumlu yazar: isilay.tuncer@kariyer.net

To cite this article: I. Tuncer, Ş. Keskin and M. Apaydın, "Content Control in İşin Olsun Platform Job Texts", *Gazi Journal of Engineering Sciences*, vol.7, no.3, pp. 243-252, 2021. doi:10.30855/gmbd.2021.03.07

dahil etmektedir. İlan iş saatleri dışında verildiği takdirde, ilanın kabul süreci daha da uzun sürebilmektedir. Özetlenen bu problemin çözümünde doğal dil işleme temelli yöntemlerin kullanılabileceği tespit edilmiştir. Doğal dil işleme, metinden anlam çıkarılmasını amaçlayan yapay zekânın alt alanlarından birisidir. Makine öğrenmesi yöntemlerinin bu alana uygulanması 1990'lı yıllarda ivme kazanmıştır. Derin öğrenme yaklaşımlarının doğal dil işleme çalışmalarında uygulanması ise 2010'lı yıllardan sonra artmıştır.

Metin sınıflandırma ve duygu analizi çalışmalarında sıklıkla doğal dil işleme yöntemleri kullanılmaktadır [2], [3], [4]. E-posta kutusundaki yaramaz (spam) postaların elenmesi veya kütüphanede kitapların kategorilere ayrılması gibi problemlerde metin sınıflandırma uygulamalarından yararlanılmaktadır. Metin sınıflandırma, doğal dil işleme metodolojilerinin kullanıldığı en bilinen uygulamalarından birisidir. Bu uygulamada verilen bir cümlenin hangi kategoriye ait olduğunun makineler tarafından tespit edilmesi amaçlanmaktadır. Metin sınıflandırmanın gerçekleştirilebilmesi için bir eğitim veri kümesi ile modelin eğitilmesi, sonrasında da ortaya çıkartılan modelin test veri kümesi ile değerlendirilmesi gerekmektedir. Bu amaçla kelime torbası, tekrarlayan sinir ağları ve son olarak 2017 itibarı ile transformer tabanlı yaklaşımlar geliştirilmiştir [5]. Bu yaklaşımların problemleri ele alış şekilleri kısaca şöyle özetlenebilir:

Kelime Torbası (Bag-of-words): Cümledeki kelimelerin sırasını dikkate almayan bir yöntemdir [6].

Tekrarlayan Sinir Ağları (Recurrent Neural Networks): Cümledeki kelimelerin sırasını dikkate alan ama kelimelerin değişik kontekslerde anlam farkına dikkat etmeyen, bir opsiyonel dikkat mekanizması ile tahminlerde cümlenin spesifik kısımlarına dikkat edebilemektedir.

Transformer: Kelimelerin sırasına dikkat eder, dikkat mekanizmasının da tekrarlayan sinir ağlarından farklı olarak çok yönlü olarak sisteme her zaman dahil olduğu, hem girdi kelimeler arasında hem de girdi-çıktı kelimeler arasındaki ilişkiyi öğrenen, dikkat mekanizması içeren, kelimelerin kontekste bağlı anlam farklarının farkında olan, 8-12-24 tabakalı derin mimari kullanan bir metodolojiye sahiptir.

Gerçekleştirilen araştırmalar sonucunda Transformer mimarilerinin ilan metinlerinde yer alan uygun olmayan içerikleri tespit etmede daha avantajlı konumda olduğu tespit edilmiştir. Transformer mimarilerinin bir diğer avantajı ise bu mimariler kullanılarak araştırmacılar tarafından oluşturulan pretrained modellerin huggingface üzerinden kullanıma sunulmuş olmasıdır [7]. Kullanıma sunulan bu pretrained modeller içerisinde Türkçe dil desteğine sahip modeller de bulunmaktadır [8], [9].

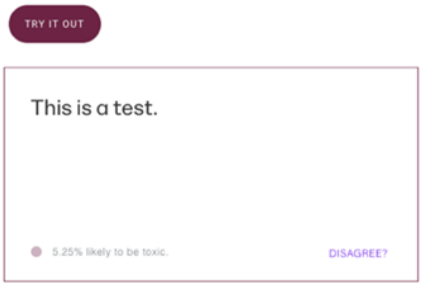
Bu makalede ilan kontrol sürecinin, doğal dil işleme konuları içerisinde günümüz en popüler yaklaşım olan transformer mimarilerine dayalı bert dil modelleri ile otomatik olarak gerçekleştirilmesi hakkında yaptığımız çalışmalar ve içerik kontrolü hakkında günümüze kadar gelen çalışmalar açıklanarak özetelenecektir. Sonrasında kullanılan teknikler ve metodolojiler anlatılacaktır. Elde edilen sonuçlar paylaşılacak ve yorumlanacaktır. Tartışma bölümü ile makale son bulacaktır.

2. ÖNCEKİ ÇALIŞMALAR (*PREVIOUS STUDIES*)

2.1. İçerik Yönetimi Hakkında Özet *(Summary About Content Management)*

İçerik yönetimi, günümüzde popüler hâle gelmeye başlayan, içeriklerin zararlı kısımlarının otomatik yakalanmasını amaçlayan bir süreçtir. Amaç metnin içeriğinde bulunan küfür, hakaret, saldırgan ifadeler gibi olumsuz içerikleri otomatik olarak filtrelemektir. Bu çalışma ile birlikte işin olsun platformunun daha güvenilir bir yer olması amaçlanmaktadır. Metinde geçen olumsuz kelimeleri filtrelemek zararlı içerik tespiti yapmanın en kolay yoludur, lakin bu yöntem yüksek seviyede olumlu anlama gelebilen cümlelerin olumsuz olarak etiketlenmesine yani "yanlış pozitif" durumuna yol açabilmektedir.

Dil modelleri ile içerik yönetimi, İngilizce başta olmak üzere daha sık kullanılan dillerde önceden geliştirilmiştir. Google şirketi, perspective.ai isimli bir şirket ile bu içerik yönetimini otomatik hâle getirmiştir, Şekil 1'de test ekranının görüntüsü yer almaktadır [10]. 2017'den itibaren bahsedilen içerik yönetimi hizmet vermeye devam etmektedir. New York Times gibi gazetelerin yorumlarının da benzer içerik yönetim sistemleri ile denetlendikten sonra sonra sistemde görünür olması söz konusudur. Türkçe doğal dil işleme çalışmaları içerisinde duygu analizi, isimli varlık tanıma, metin sınıflandırma konularında literatürde yer alan çalışmalar bulunmaktadır, yalnız içerik yönetimi üzerine bilgimiz dahilinde yapılan bir çalışma yoktur [2], [3], [11]. Kara liste yaklaşımları ile bir nebze de olsa çözüm üretebilen yaklaşımlar kullanılmaktadır.



Şekil 1. Perspective API test ekranı [10]
(Perspective API test screen)

Kara liste ile içerik yönetiminde bazı yasaklı kelimelerin tutulduğu bir liste bulunmaktadır, bu listede geçen bir kelimenin içerikte yer alması durumunda içerik doğrudan yasaklanabilmektedir. Bu kara listedeki kelimelerin içerikte varlığını kontrol etmek için bilgisayar mühendisliğinde iyi bilinen regular expression olarak isimlendirilen regex ile bu işlemler yapılabilmektedir. Bu yöntemin kullanılması sonucunda problemli durumlar ile karşılaşılabilmektedir. Örneğin, ingilizcede "Scunthorpe problem" olarak isimlendirilen [12] ve sonucunda yanlış pozitiflere neden olan bir durum söz konusu olabilmektedir. Scunthorpe şehrinden bir sitede hesap açmak isteyenlerin başvurusu otomatik olarak sistem tarafından reddedilebilmektedir. Çünkü "Scunthorpe" kelimesi içerisinde ingilizce dilinde müstehcen olarak nitelendirilen bir kelime yer almaktadır. Kara liste yaklaşımı ile filtrelenen bu kelime içerisindeki müstehcen metinden dolayı kara liste filtrelerine yakalanmaktadır.

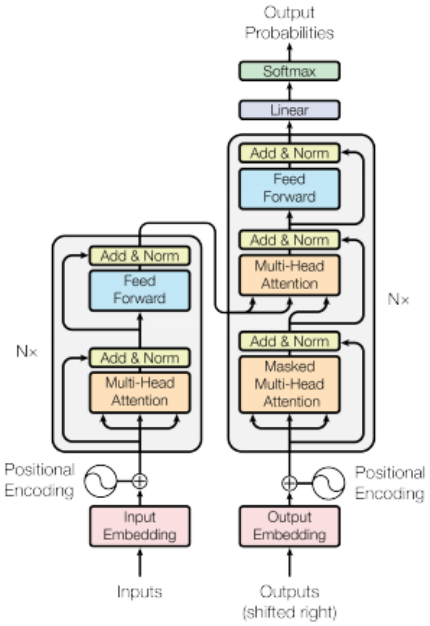
Bunun yanında daha karmaşık, anlama dayalı içerik yönetimi daha zordur. Burada içeriğin ne anlama geldiğini anlamak gerekmektedir. İşin olsun platformunun özelliği sadece işverenlerin ilan verebildiği bir platform olmasıdır. Adayların işin olsun platformunu amacı dışında kullandığı durumda, ilan veren işveren değil de iş arayan olarak sisteme yapılan girişlerin de bloklanması gerekmektedir. Bu tip girdilerin bloklanması kara liste methodolojisi ile mümkün değildir. Bu sebepten dolayı anlama da dayalı bir sistem kurulması gerekmektedir.

3. METODOLOJİ *(METHODOLOGY)*

Bu bölümde transformer mimarileri, Bert, kullanılan denetimli makine öğrenimi algoritmaları hakkında detaylı bilgiler verilmektedir.

3.1. Transformer Mimarileri ve Bert *(Transformer Architectures and Bert)*

Transformer mimarileri, Bert ile 2017 yılında literatürde yerini almıştır [13]. Bu yaklaşımda girdi ve çıktı için kodlayıcı (encoder) ve kod çözücü (decoder) mimarileri kullanılmaktadır. Kodlayıcı ve kod çözücü mimarilerinin her ikisi de derin mimari olarak tasarlanmıştır, aralarında dikkat mekanizmaları da (attention) vardır. Girdi olarak verilen kelimelerin kodlayıcıları daha sonraki katmana girdi olarak verilmekte, bu esnada bu kelimelerin diğer kelimeler ile olan etkileşimi de dikkat mekanizmaları içinde değerlendirilmektedir. En son katmanda kodlayıcının çıktısı kod çözücüye girdi olarak verilmektedir. Kod çözücü de hem girdideki hücreler ile olan dikkat mekanizmaları (Şekil 2'de görüldüğü üzere), girdi kodlayıcıda ayrı, kodlayıcı ile çıktı kodlayıcı arasında ayrı, hem de çıktı kodlayıcı içinde de ayrı dikkat mekanizmaları mevcuttur. Bu yaklaşım ile Şekil 3'te görüldüğü üzere her bir kelime bir 768 boyutlu vektör ile temsil edilmektedir.



Şekil 2. Transformer model mimarisi [13]
(Transformer model architecture)

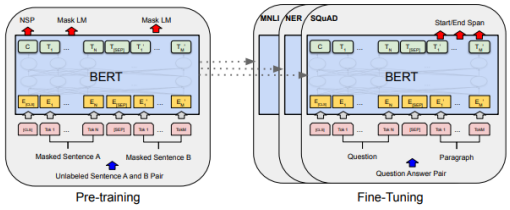
Bert, Google'un 2017 yılında sunduğu bir yöntemdir [5]. Bu yaklaşımda cümlenin farklı kısımlarına önem veren bir dil modeli eğitilmesi söz konusudur. Bu dil modeli maskelenmiş dil modeli olarak tanımlanmaktadır. Eğitim işlemi denetimsiz bir şekilde gerçekleşmektedir. Eğitim kümesinde bulunan cümlelerden rasgele bir kelime maskelenmekte ve kelimenin doğru bir şekilde tahmin edilebilmesi amacıyla dil modeli eğitilmektedir. Bunun yanında ardışık cümle çiftleri ve ardışık olmayan cümle çiftleri de girdi olarak Bert'e verilmektedir ve Bert'in ardışık kelime çiftlerini ardışık olmayanlardan ayırt etmesi de mümkündür. Bir cümlenin, daha sonra

sınıflandırılması gibi bir alt görevde transfer öğrenme ile sınıflandırma yapılması mümkündür [14].

Huggingface bu tip transformer mimarilerini implement etmiş New York tabanlı bir şirkettir. Bu şirketin uygulaması kullanarak Bert dil modeli Türkçe veri kümeleri üzerinde de eğitilmiştir [8] . Eğitimde kullanılan 35 GB boyuta sahip eğitim korpusu 44.04.976.662 küçük metin parçasından (token) oluşmaktadır. Bunlardan birisi Dbmdz isimli Almanya tabanlı bir Münih kütüphanesinin modelidir ve çalışma kapsamında bu modele transfer öğrenme (fine tuning) uygulanarak deneyler gerçekleştirilmiştir.

Bert'in kodlayıcı mimarisi mevcuttur. Kodlayıcı mimarisinde girdi aşamasında köklerine ayrılan kelimeler daha sonra derin bir mimari ağından geçmekte, burada word2vec yaklaşımlarından farklı olarak kelimelerin bulunduğu kontekste de bağlı olarak temsil edilmesi söz konusu olmaktadır.

Bert-Transfer Öğrenme sürecinde bert mimarisinin kelimeleri kodlamak kısmında bir değişiklik söz konusu değildir. Sekans sınıflandırma işleminde pytorch ile kodlanan metnin çıktısı bir İleri Beslemeli Sinir Ağına (Feedforward Neural Network (FFNN) girdi olarak verilmekte, bu ağın ağırlıkları etiketli veri ile eğitilmektedir. Böylece bir dil modelini sıfırdan eğitmek söz konusu değildir.



Şekil 3. Bert transfer öğrenme mimarisi örneği [5]
(Bert transfer learning architecture example)

3.2. Denetimli Makine Öğrenimi Algoritmaları
(Supervised Machine Learning Algorithms)

Bu bölümde, çalışma içerisinde kullanılan denetimli makine öğrenimi algoritmaları hakkında kısaca bilgi verilmektedir.

Lojistik Regresyon, ikili sınıflandırma problemlerinde, bağımlı değişkenin iki farklı değer aldığı durumlarda yüksek başarım gösteren bir algoritmadır [15]. Doğrusal sınıflandırma problemlerinde yaygın olarak kullanılmaktadır. Bu bağımlı değişkenin; ilan içeriğinin uygun ya da uygunsuz olarak alabileceği değerlerin gerçekleşme olasılığını tespit eder. Algoritmanın matematiksel gösterimi Formül (1)'de gösterilmiştir, p başarı olasılığını temsil etmektedir.

$$logit(y) = \log(\frac{p}{1-p}) \quad (1)$$

Destek Vektör Makinesi, düzlem üzerine yerleştirilmiş iki farklı sınıfı ayırmak için uygun karar sınırını bulan doğrusal ve doğrusal olmayan verileri sınıflandırabilen algoritmadır [16]. Doğrusal olarak sınıflandırılabilen verilerde, düzlem üzerindeki noktaları ayırmak için bir doğru çizer ve çizilen doğrunun iki sınıfın noktalarına da maksimum uzaklıkta olmasını hedefler. Doğrusal olarak sınıflandırılamayan doğrusal olmayan verilerde ise, yeni 3. bir boyut oluşturarak sınıflandırma işlemini gerçekleştirmeye çalışır [17]. Çalışma içerisinde 4 farklı kernel modu ile denemeler gerçekleştirilmiştir: Rbf kernel, Sigmoid kernel, Polynomial kernel ve Linear kernel. En başarılı sonuç ise Formül (2)'de matematiksel gösterimi yer alan Rbf kernel ile elde edilmiştir.

$$f(x) = \sum_{i=1}^{n} \alpha_i g(x - x_i). \quad (2)$$

Rastgele Orman, hem sınıflandırma hem regresyon problemleri için kullanılabilen, birden fazla karar ağacı oluşturarak doğru bir tahmin yapabilmek için bu karar ağaçlarını birleştiren denetimli öğrenme algoritmasıdır [18].

Xgboost Sınıflandırıcı, temeli karar ağacına dayanan gradyan arttırma çerçevesini kullanan hibrit bir algoritmadır [19]. Xgboost'un gelişimi, karar ağaçlarından başlayarak torbalama, rasgele orman, arttırma ve gradyan arttırma olarak devam edip en son şeklini almıştır.

K-En Yakın Komşu (KNN), temeli uzaklık ve komşuluk sayısı olarak iki değer üzerine kurulu olan, tahmin edilecek olan vektörün en yakınında bulunan komşu vektörlerin sınıfına göre tahmin üreten sınıflandırma algoritmasıdır [20]. Verilen optimum k parametresi ile en yakın k komşu vektör üzerinden hesaplama yapmaktadır. Verilen k değeri çok az olduğu durumda aşırı öğrenme (overfit) oluşabilir, çok fazla olduğu durumda da çok genel tahminler üretmektedir. K komşuluk sayısı optimum seçilerek ideal tahmin sonuçlarına ulaşılmaktadır.

4. BULGULAR *(RESULTS)*

Bu bölümde farklı metodolojiler ile gerçekleştirilen deneylerin detayları, sonuçları ve gerçekleştirilen veri ön işleme operasyonları hakkında bilgi verilmektedir. Ayrıca çalışmada kullanılan veri seti ve veri setinin vektörel temsillerinin elde ediliş yöntemleri açıklanmaktadır.

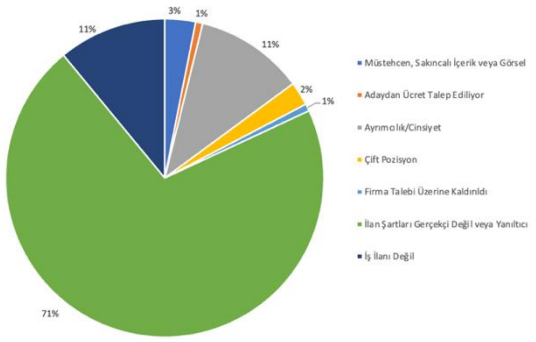
Deneyler, 12 GB GPU kapasitesine sahip GeForce RTX 2080 Ti ekran kartı kullanan Ubuntu işletim sistemine sahip bir bilgisayarda gerçekleştirildi.

4.1. Veri Seti *(Dataset)*

Çalışmada kullanılan veri seti İşin olsun ilan bilgilerinin tutulduğu, uygun ve sebebiyle birlikte uygun olmayan olarak etiketlenmiş ilan metinlerinden oluşmaktadır. İlanlar, Kariyer.net Müşteri Çözüm Merkezi tarafından kontrol edilip, uygun veya uygunsuz olarak etiketlenmektedir. Uygunsuz olarak etiketlenen ilanların kategorileri başlıkları:

- Müstehcen, Sakıncalı İçerik veya Görsel,
- Adaydan Ücret Talep Ediliyor,
- Ayrımcılık/Cinsiyet,
- Çift Pozisyon,
- Firma Talebi Üzerine Kaldırıldı,
- İlan Şartları Gerçekçi Değil veya Yanıltıcı,
- İş İlanı Değil.

Olumsuzolarak etiketlenen ilanların dağılımları Şekil 4'te yer almaktadır.



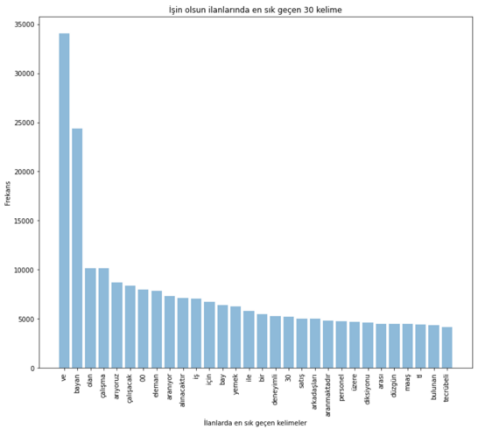
Şekil 4. Uygun olmayan ilan etiketlerinin dağılım *(Distribution of inappropriate job text labels)*

Farklı sebeplerden dolayı uygun görülmeyerek olumsuz olarak etiketlenen ilanlar, çalışma metodolojisinin ikili sınıflandırma problemi olarak ele alınması sebebi ile sadece "uygun olmayan" ilan etiketi altında birleştirilmiştir. Uygun olmayan etiketine sahip "Bekar beyin evinin temizlik ve yemek işiyle ilgilenecek güleryüzlü çalışkan samimi bayan aranıyor yatılı" örneğinde, ilan içeriğinde yer alan "bekar", "samimi", "bayan" ve "yatılı" kelimelerinin birlikte bulunması risk teşkil ettiğinden dolayı kontrolörler tarafından "Müstehcen, Sakıncalı İçerik veya Görsel" kategorisi altında uygun olmayan olarak etiketlenmiştir. Uygun ilana örnek olarak; "Manav mağazamızda çalıştırılmak üzere kasiyer aramaktayız" ilanının içeriği verilebilir.

Son durumda veri seti içerisinde bulunan etiketlerin dağılımı:
- Uygun etiketli ilan sayısı 49681,
- Uygun olmayan olarak etiketlenmiş ilan sayısı 49639,
- Toplam veri sayısı da 99320.

Şekil 5. Veri seti içerisinde bulunan etiketlerin dağılımı *(Distribution of labels in the dataset)*

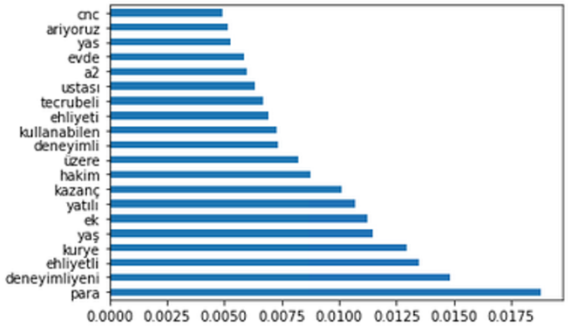
Gerçekleştirilen keşifsel veri analizi çalışmalarında, ilan metinlerinin içerisinde en sık geçen kelimeler tespit edilmiştir. En sık geçen 30 kelimenin grafiği Şekil 6'da yer almaktadır.



Şekil 6. İlan metinleri içinde en sık kullanılan 30 kelime *(30 most frequently used words in job texts)*

İlan metinleri Tf-Idf ile vektörize edilip ve en çok kullanılan kelime grupları ile matrisler oluşturulmuş, ayrı ayrı denemelerde farklı kelime sayıları ile sonuçlar gözlemlenmiştir. 10.000 kelime ile oluşturulan matrislerde en yüksek başarı elde edilmiş, 10.000'den yüksek kelime sayılarında başarının düştüğü gözlemlenmiştir. Tf-Idf, sözcüğün dokümanda ortaya çıkma sıklığını ve dokümandaki

temsilini gösteren istatistiksel bir ağırlık faktörüdür. Tf-Idf ile metin verileri kullanılarak oluşturulan vektörlerden bir matris inşa edilir. Oluşturulan matris üzerinden Xgboost algoritması ile öznitelik önemi analizi yapılmıştır ve Şekil 7'deki grafik incelendiğinde metinler içinde tahminde en çok dikkat edilen kelimenin 'para' olduğu görülmektedir, ardından 'deneyimli', 'yeni', 'ehliyetli', 'kurye' ve 'yaş' kelimeleri önem sırasını takip etmektedir.



Şekil 7. Modellerin tahmin sırasında en çok dikkat ettiği kelimeler *(The words that the models paid the most attention during the prediction)*

Deneylerde kullanılan veri setinin makine öğrenimi metodolojisine uygun hale getirilmesi amacıyla veri farklı kesitlere ayrıldı. Tüm veri eğitim seti, doğrulama seti ve test seti olmak üzere 3 parçaya ayrıldı. Bu çalışmada kullanılan oranlar:

70 % eğitim seti,
15 % test seti,
15 % doğrulama seti.

Veri Ön İşleme Adımları: Makine öğrenimi çalışmalarında veri temizliği problemden probleme göre farklılık göstermektedir. Çalışma içerisinde veri üzerinde gerçekleştirilen ön işleme aşamaları aşağıda sıralı bir şekilde verilmiştir:

- Metin temizleme aşamasında beautifulsoup kütüphanesi kullanıldı,

- Metnin içeriğinin dili textpipe isimli kütüphane ile tespit edildi,

- Dili Türkçe olmayan içerikler eğitime dahil edilmedi,

- Beautifulsoup kütüphanesi ile html etiketlerini ortadan kaldırıldı,

- Kelimelerin hepsi küçük harflerle temsil edilir hale getirildi.

4.2. Sonuçlar *(Results)*

Gerçekleştirilen deneylerde Python programlama dili kullanıldı. Denetimli makine öğrenimi çalışmalarında Scikit-learn kütüphanesi, Bert çalışmalarında ise PyTorch-Transformers kütüphanesi tercih edildi.

Makine öğrenimi çalışmalarında gerçekleştirilen deneylerin, çalışma metodolojisine uygun başarı ölçüm metrikleri ile değerlendirilmesi gerekmektedir. Bu çalışmada sonuçların gözlemlenmesi için birden fazla metrik kullanıldı. Bu metrikler doğruluk, duyarlılık, kesinlik ve F puanıdır.

Doğruluk: Yapılan tahminler içerisindeki doğru cevapların, tüm cevaplara oranını temsil eder. En yaygın kullanılan değerlendirme yöntemidir. Formül 3'te gösterilmektedir.

$$Doğruluk = \frac{DoğruPozitif + DoğruNegatif}{ToplamTahmin} \quad (3)$$

Duyarlılık: Tahminlerin içerisinde doğru tespit edilen pozitif sınıfların, tüm pozitiflere oranıdır. Formül 4'te gösterilmektedir.

$$Duyarlılık = \frac{DoğruPozitif}{YanlışNegatif + DoğruPozitif} \quad (4)$$

Kesinlik: Pozitif olarak tahmin edilen değerlerin gerçekten kaç adedinin pozitif olduğunu temsil eder. Formül 5'te gösterilmektedir.

$$Kesinlik = \frac{DoğruPozitif}{YanlışPozitif + DoğruPozitif} \quad (5)$$

F Puanı: Kesinlik ve Duyarlılık değerlerinin harmonik ortalamasıdır. Metriğin hesaplanışı Formül 6 'da gösterilmektedir.

$$FPuanı = 2 \times (\frac{Duyarlılık \times Kesinlik}{Duyarlılık + Kesinlik}) \quad (6)$$

4.2.1. Denetimli makine öğrenimi algoritmaları sonuçları *(Supervised machine learning algorithms results)*

İlan içeriklerinden oluşan veri kümesi kullanılarak denetimli makine öğrenimi algoritmaları kullanılarak deneyler yapılmıştır. Deneylerde XGboost sınıflandırıcı, rastgele orman, farklı kernel modlarında destek vektör makinesi ve KNN algoritmaları kullanılmıştır. En başarılı sonuç 'rbf' kernel'i ile destek vektör makinesi algoritmasında elde edilmiştir. Modellerin test verisi üzerinde verdiği sonuçlar Tablo 1'de verilmiştir.

Tablo 1. Denetimli makine öğrenimi algoritmaları sonuçları *(Supervised machine learning algorithms results)*

Algoritma	Test Doğruluk	F Puanı	Duyarlılık	Kesinlik
Destek Vektör Makinesi(kernel='rbf')	75%	75%	75%	76%
Lojistik Regresyon	75%	75%	75%	75%
Destek Vektör Makinesi(kernel='sigmoid')	74%	74%	74%	74%
XGboost Sınıflandırıcı	74%	74%	74%	74%
Destek Vektör Makinesi(kernel='linear')	74%	74%	74%	74%
Destek Vektör Makinesi(kernel='poly')	73%	73%	73%	73%
Rastgele Orman	68%	68%	68%	68%
KNN (k=8)	65%	63%	65%	67%

Çalışmanın ilk aşamasında ilan metinlerini vektörler ile temsil edilerek sınıflandırma algoritmalarına eğitim ve test işlemleri için girdi olarak verilmiştir. Metinleri vektörize etmek için farklı farklı metodolojiler kullanılmıştır, en başarılı sonuç ise TF-IDF ile elde edilmiştir. Metinleri sayılarla temsil edilebilir hale, vektöler temsillerin elde edilmesini sağlayan TF-IDF en sık tercih edilen algoritmaların başında gelmektedir.

Tf-Idf, metnin içerisindeki her bir farklı kelimeyi bir sayı ile temsil eder. Her bir cümle de bu sayıların ağırlıklarından oluşan bir vektördür. TF, terimin bir belgede görünme sayısının, belgedeki toplam kelime sayısına bölümü ile hesaplanır. IDF, terimin göründüğü belge sayısının toplam belge sayısına bölümünün logaritması alınarak hesaplanır. TF-IDF ağırlıklandırması, TF ve IDF değerlerinin çarpılarak hesaplanması sonucunda hesaplanır [21]. Matematiksel gösterimi Formül 7'de yer almaktadır/

$$w_{i,j} = tf_{i,j} \times \log(\frac{N}{df_i}) \quad (7)$$

4.2.2. Bert sonuçları *(Bert results)*

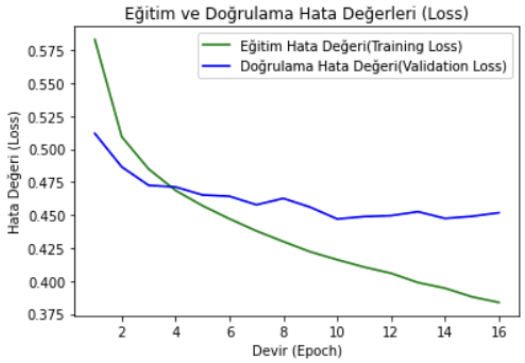
Çalışma içerisinde Bert'in huggingface'te bulunan "bert-base-turkish-uncased" modeli ve tokenizer'i kullanılmıştır [9]. Elimizde bulunan veri üzerinde gerçekleştirdiğimiz keşifsel veri analizi sonucunda, metinlerin kelime sayısının 350'yi geçmediği tespit edilmiştir. Bundan dolayı modelin girdi uzunluğu 350 olarak seçilmiştir. Optimize edici olarak "Adam Optimizer" seçilmiş olup, batch size değeri 16 olarak ayarlanmıştır.

Eğitim işlemi sırasında, 10 devirden (epoch) sonra modelin eğitim ve doğrulama hata değerleri (loss) arasındaki farkın giderek artarak modeli aşırı öğrenmeye götürdüğü tespit edilmiştir. Çalışma içerisinde 10 devir boyunca eğitilen Bert modeli kullanılmıştır. Tablo 2'de 16 devir boyunca eğitilen modelin hata değerleri toplamı (loss) ve metrik değerleri yer almaktadır.

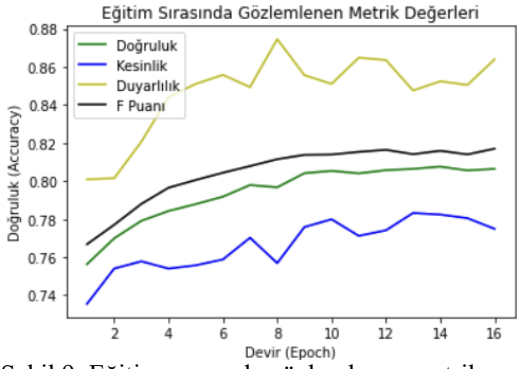
Tablo 2. Bert modelinin eğitim sırasında verdiği çıktılar *(Outputs of the Bert model during training)*

Devir	Eğitim Hata Değerleri (Training Loss)	Doğrulama Hata Değerleri(Validation Loss)	Eğitim Süresi	F Puanı	Kesinlik	Duyarlılık	Doğruluk
1	0.583090	0.511867	0:26:04	0.766820	0.735482	0.800947	0.756341
2	0.509378	0.486514	0:26:04	0.777043	0.754030	0.801504	0.769928
3	0.484861	0.472516	0:26:04	0.788017	0.757814	0.820727	0.779125
4	0.468627	0.471260	0:26:04	0.796530	0.754013	0.844129	0.784281
5	0.457046	0.465230	0:26:03	0.800577	0.755720	0.851093	0.787904
6	0.447109	0.464239	0:26:04	0.804399	0.758800	0.855830	0.791806
7	0.437926	0.457756	0:26:03	0.807896	0.770241	0.849422	0.797938
8	0.429855	0.462643	0:26:03	0.811502	0.756871	0.874634	0.796753
9	0.422104	0.455918	0:26:04	0.813804	0.775827	0.855690	0.804139
10	0.416082	0.446991	0:26:06	0.813961	0.779934	0.851093	0.805393
11	0.410511	0.448857	0:26:06	0.815364	0.771209	0.864884	0.804069
12	0.405800	0.449549	0:26:06	0.816434	0.774129	0.863630	0.805741
13	0.398771	0.452556	0:26:04	0.814156	0.783241	0.847611	0.806438
14	0.394478	0.447423	0:26:03	0.815946	0.782409	0.852486	0.807623
15	0.387972	0.449016	0:26:03	0.814025	0.780519	0.850536	0.805602
16	0.383679	0.451735	0:26:05	0.817044	0.774891	0.864048	0.806438

Tablo 2'de yer alan hata değerlerinin ve metriklerin görselleştirilmiş sonuçları sırasıyla Şekil 8 ve Şekil 9'da yer almaktadır.



Şekil 8. Eğitim ve doğrulama loss değerleri *(Training and validation loss values)*

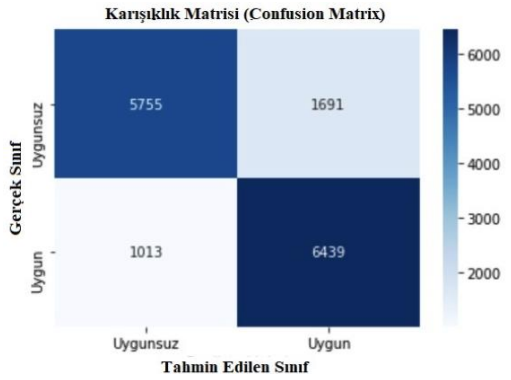


Şekil 9. Eğitim sırasında gözlemlenen metrik değerleri *(Metric values observed during training)*

Eğitim işleminin sonlandırılması sonucunda, transfer öğrenme ile elde edilen en ideal Bert modeli üzerinde test verisi kullanılarak modelin uygun ve uygunsuz etiketler üzerindeki başarısı Tablo 3'te gösterilmiştir. Karışıklık matrisi Şekil 10'da verilmiştir.

Tablo 3. Bert modelinin test verisi üzerindeki çıktıları *(Outputs of the Bert model on the test data)*

	Kesinlik	Duyarlılık	F Puanı
Uygunsuz İlan	0.850325	0.772898	0.809765
Uygun İçerikli İlanlar	0.792005	0.864063	0.826466



Şekil 10. Karışıklık matrisi değerleri *(Confusion matrix values*)

Doğal dil işleme problemlerinde sıklıkla kullanılan ve türkçe dilinde de eğitilmiş Electra ve DistilBert modelleri ile Bert modelinin karşılaştırılması yapılmıştır [22], [23]. Türkçe dil desteğine sahip Electra ve DistilBert modelleri huggingface üzerinden kullanıma sunulmuş durumdadır [24], [25]. Çalışmada transfer öğrenme ile bu modeller eğitilerek performansları gözlemlenmiştir. Bert, Electra ve DistilBert'e göre daha başarılı sonuç vermiştir, bu sonuçlar Tablo 4'te yer almaktadır.

Tablo 4. Bert, electra ve distilbert modellerinin test verisi üzerindeki çıktıları (Outputs of bert, electra and *distilbert models on test data)*

	Kesinlik	**Duyarlılık**	**F Puanı**	**Doğruluk**
Bert	0.803455	0.802580	0.802446	0.802591
DistilBert	0.709462	0.693662	0.687774	0.693662
Electra	0.719607	0.707746	0.703747	0.707746

5. TARTIŞMALAR VE SONUÇ (*DISCUSSION AND CONCLUSION*)

Çalışma ile İşin Olsun platformunda yayına alınacak ilanlarının uygun veya uygunsuz olduğunun tespitinin kontrolör kişilerce manuel olarak yapılması yerine derin öğrenme tabanlı bir metodoloji ile kontrolün sağlanması hedeflenmiştir. İşin olsun platformunda her gün ortalama 1500 ilan yayınlanmaktadır. Gerçekleştirilen çalışma ile her gün kontrol edilen ilan sayısının arttırılması ve uygunsuz içeriğe sahip ilanların yayında kalma süresinin kısaltılması hedeflenmektedir.

Tf-Idf kelimelerin cümleler içindeki sıklığı ile ilgilenirken, transformer dil modelleri kelimenin bağlamına da önem vermektedir. Birden fazla anlama gelen kelimelerin vektörlerini birbirinden ayırmak için kelimenin bağlamını dikkate almaktadır. Bu nedenle çalışmada cümlelerin vektörize edilmesi aşamasında Bert tercih edilmiştir. Bert ile vektörize edilen cümleler Bert For Sequence Classification model ile eğitilmiştir. Deneylerin gerçekleştirildiği farklı farklı yöntemler içerisinde en başarılı sonuçlar Bert ile elde edilmiştir.

Eğitilen Bert model İşin Olsun ilanlarının, yayına alınmadan denetlenmesi amacıyla sistem içerisinde gerçek zamanlı olarak çalışmaktadır. Model her yayınlanan ilan için bir risk skoru üretmektedir. 80-100 arası bir skor almış ilan yüksek riskli içeriğe sahip olarak kabul edilmektedir. Risk skorları, ilan bilgileri ile kontrolörün önüne düşmekte ve risk skorlarına göre ilanın yayınlanıp, yayınlanmayacağına karar verilmektedir.

Çalışma içerisinde kullanılan veri setinde uygunsuz kategorileri içerisinde ayrı ayrı etiketlere sahip veriler, tek bir olumsuz etiket altında birleştirilerek ikili sınıflandırma metodolijisine uygun bir hale getirilmiştir. Ancak her uygunsuz kategori içerisindeki ilan metinleri ayrı özellikler taşımaktadır. Çalışmanın devamında uygunsuz kategorisinde yer alan her bir etikete ait veri sayısının arttırılarak dengeli bir veri seti elde edilmesi ve bu veri seti ile çoklu sınıflandırma yapabilen bir modelin oluşturulması planlanmaktadır. Ek olarak transformer modeller denetimli/denetimsiz öğrenme algoritmaları ile kıyaslandığında oldukça fazla boyuta sahiptir. Eğitilen yüksek boyutlu modelleri canlı ortamda kullanmak kaynakların fazla tüketimine neden olmaktadır. Hafif (lightweight) çözümlerle, hem dosya okuma ve ara işlemlerin hızlandırılması hem de modellerin boyutlarının azaltılarak kaynak tüketiminin minimuma indirilmesi hedeflenmektedir.

ÇIKAR ÇATIŞMASI BİLDİRİMİ *(CONFLICT OF INTEREST STATEMENT)*

Yazarlar tarafından herhangi bir çıkar çatışması bildirilmemiştir.

KAYNAKLAR *(REFERENCES)*

[1] [Online]. Available: https://isinolsun.com. [Accessed: Haz. 23, 2021].

[2] A. C. Tantuğ, "Metin sınıflandırma (text classification)" *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi*, vol. 5, no. 2, pp. 1-3, Haziran 2012.

[3] A. Köksal and A. Özgür, "Twitter dataset and evaluation of transformers for turkish sentiment analysis" in *2021 29th Signal Processing and Communications Applications Conference (SIU),*

Haziran 9-11, 2021, İstanbul, Türkiye. IEEE, 2021, pp. 1–4. doi: 10.1109/SIU53274.2021.9477814

[4] U. U. Acikalin, B. Bardak, and M. Kutlu, "Turkish sentiment analysis using bert" in *2020 28th Signal Processing and Communications Applications Conference (SIU), 2020, Gaziantep, Türkiye*. 2020, pp. 1–4. doi: 10.1109/SIU49456.2020.9302492

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, June 2019, Minneapolis, Minnesota, USA, 2019*, Association for Computational Linguistics, 2019. pp. 1-9. doi:10.18653/v1/N19-1423

[6] Y. Zhang, R. Jin, and Z.-H. Zhou, "Understanding bag-of-words model: A statistical framework" *International Journal of Machine Learning and Cybernetics*, vol. 1, pp. 43–52, Aralık 2010. doi: 10.1007/s13042-010-0001-0

[7] [Online]. Available: https://huggingface.co/transformers/model_doc/bert. [Accessed: May. 20, 2021].

[8] S. Schweter, "Berturk - Bert Models for Turkish" Nisan 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3770924. [Accessed: May. 20, 2021].

[9] [Online]. Available: https://huggingface.co/dbmdz/bert-base-turkish-cased. [Accessed: May. 20, 2021].

[10] [Online]. Available: https://www.perspectiveapi.com. [Accessed: May. 21, 2021].

[11] K. Dilek, et al. "Named entity recognition experiments on Turkish texts" in *International Conference on Flexible Query Answering Systems. 2009, Roskilde, Denmark, October 26-28, 2009*. Springer, Berlin, Heidelberg, 2009. pp. 524-535. doi:10.1007/978-3-642-04957-6_45

[12] [Online]. Available: https://en.wikipedia.org/wiki/Scunthorpe_problem. [Accessed: Haz. 17, 2021].

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arxiv.org* , 2017. [Online]. Available: https://arxiv.org/pdf/1706.03762.pdf. [Accessed: May. 23, 2021].

[14] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune bert for text classification?", *Chinese Computational Linguistics*, pp. 194-206, 2019. doi: 10.1007/978-3-030-32381-3_16

[15] C. Y. Peng, K. L. Lee, and G. Ingersoll, "An introduction to logisticr egression analysis and reporting" *The Journal of Educational Research*, vol. 96, pp. 14 – 3, 2002.
doi: 10.1080/00220670209598786

[16] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, 2004. doi: 10.1007/BF00994018

[17] G. Dogan, N. Alotaibi, E. Sahin, S. S. Ertas, I. Cay, R. Keskin, M. J. H. Heijnen, and K. Ricanek, "Using artificial intelligence to predict fall-risk during adaptive locomotion in humans" *in 2020 International Conference on Artificial Intelligence Modern Assistive Technology (ICAIMAT), Riyadh, Saudi Arabia, Nov. 24-26, 2020.* IEEE 2020. pp. 1-7. doi: 10.1109/ICAIMAT51101.2020.9308007

[18] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2004. doi: 10.1023/A:1010933404324

[19] T. Chen and C. Guestrin, "Xgboost: A Scalable Tree Boosting System" in *KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Ağustos 13-17, 2016, San Francisco, CA, USA*. 2016. pp. 785-794. doi: 10.1145/2939672.2939785

[20] P. Soucy and G. Mineau, "A simple knn algorithm for text categorization" in *Proceedings 2001 IEEE International Conference on Data Mining, 29 Nov-2 Dec. 2001, San Jose, CA, USA*, IEEE, 2002. pp. 647–648. doi: 10.1109/ICDM.2001.989592

[21] G. M. Demirci, R. Keskin, and G. Doğan, "Sentiment analysis in turkish with deep learning" in *2019 IEEE International Conference on Big Data (Big Data), 9-12 Dec. 2019, Los Angeles, CA, USA.* IEEE, 2019. pp. 2215–2221. doi: 10.1109/BigData47090.2019.9006066

[22] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster,

cheaper and lighter", Ekim 2019. doi: arxiv-1910.01108.

[23] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators", *Arxiv* 2020. doi: arxiv-2003.10555.

[24] [Online]. Available: https://huggingface.co/docs/transformers/model_doc/electra. [Accessed: May. 27, 2021].

[25] [Online]. Available: https://huggingface.co/docs/transformers/model_doc/distilbert. [Accessed: May. 27, 2021].