YAPAY ZEKA YÖNTEMLERİ İLE ATATÜRK ÜNİVERSİTESİ AÇIKÖĞRETİM FAKÜLTESİ SINAV SORULARININ ZORLUK DERECESİNİN TESPİTİ

Adem CANPOLAT

Yüksek Lisans Tezi
Bilgisayar Mühendisliği Ana Bilim Dalı
Dr. Öğr. Üyesi Barış ÖZYER
2021

T.C.
ATATÜRK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI

YAPAY ZEKA YÖNTEMLERİ İLE ATATÜRK ÜNİVERSİTESİ AÇIKÖĞRETİM FAKÜLTESİ SINAV SORULARININ ZORLUK DERECESİNİN TESPİTİ

(Determining The Difficulty Level Of Atatürk University Open Education Faculty Exam Questions By Artificial Intelligence Methods)

YÜKSEK LİSANS TEZİ

Adem CANPOLAT

Danışman: Dr. Öğr. Üyesi Barış ÖZYER

Erzurum

Eylül, 2021

KABUL VE ONAY TUTANAĞI

Adem CANPOLAT tarafından hazırlanan "YAPAY ZEKA YÖNTEMLERİ İLE ATATÜRK ÜNİVERSİTESİ AÇIKÖĞRETİM FAKÜLTESİ SINAV SORULARININ ZORLUK DERECESİNİN TESPİTİ " başlıklı çalışması 11/10/2021 tarihinde yapılan tez savunma sınavı sonucunda başarılı bulunarak jürimiz tarafından Bilgisayar Mühendisliği Ana Bilim Dalı, Bilgisayar Mühendisliği Bilim Dalında Yüksek Lisans Tezi tezi olarak kabul edilmiştir.

Jüri Başkanı:	Prof. Dr. Köksal ERENTÜRK *Atatürk Üniversitesi*	Aslı Islak İmzalıdır
Danışman:	Dr. Öğr. Üyesi Barış ÖZYER *Atatürk Üniversitesi*	Aslı Islak İmzalıdır
Jüri Üyesi:	Dr. Öğr. Üyesi ÖZGE Öztimur KARADAĞ *Alanya Alaaddin Keykubat Üniversitesi*	Aslı Islak İmzalıdır

Enstitü Yönetim Kurulunun …./…/…. tarih ve …… sayılı kararı.

Bu tezin Atatürk Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliği'nin ilgili maddelerinde belirtilen şartları yerine getirdiğini onaylarım.

Prof. Dr. Saltuk Buğrahan CEYHUN
Enstitü Müdürü

FEN BİLİMLERİ ENSTİTÜSÜ
Graduate School of Natural and Applied Sciences

TEZ BENZERLİK ORANI BEYAN FORMU[1]

Öğrencinin Adı ve Soyadı	Adem CANPOLAT
Öğrencinin Numarası	**18040201013**
Ana Bilim Dalı	Bilgisayar Mühendisliği
Bilim Dalı	Bilgisayar Mühendisliği
Öğrencinin Kayıtlı Olduğu Program Türü	Yüksek Lisans

Yukarıda bilgileri verilen tezin intihal tespit yazılımıyla (Turnitin) yapılan tarama sonucunda elde edilen benzerlik oranları aşağıdaki gibidir. Sunulan bilgilerin doğru olduğunu, aksi halde doğacak hukuki sorumlulukları kabul ettiğimizi beyan ederiz.

Tez Bölümleri	Tezin Benzerlik Oranı (%)	Maksimum Oran (%)
Giriş	*5*	**30**
Kuramsal Temeller	*13*	**30**
Materyal ve Yöntem	*8*	**35**
Bulgular	*2*	**20**
Tartışma	*2*	**20**
Tezin Geneli	*6*	**25**

***Not:** Yedi kelimeye kadar benzerlikler ile Başlık, Kaynakça, İçindekiler, Teşekkür, Dizin ve Ekler kısımları tarama dışı bırakılabilir. Yukarıdaki azami benzerlik oranları yanında tek bir kaynaktan olan benzerlik oranlarının **%5'den** büyük olmaması gerekir.*

Tez Yazarı (Öğrenci)	**Tez Danışmanı**
Adem CANPOLAT	**Dr. Öğr. Üyesi Barış ÖZYER**
22.9.2021	**22.9.2021**
İmza: Aslı Islak İmzalıdır	İmza: Aslı Islak İmzalıdır

[1] Bu form bilgisayar ortamında doldurulmalı, çıktısı imzalanıp Tez Savunması Jüri Öneri Formu'yla birlikte Ana Bilim Dalı Başkanlığı aracılığıyla ÜBYS üzerinden Enstitüye iletilmelidir.

TEŞEKKÜR

Bu tezin hazırlanması süresince verdiği büyük destek ve katkı için danışmanım ve değerli hocam Sayın Dr. Öğr. Üyesi Barış ÖZYER' e ve fikirleri ile her zaman yanımızda olan Dr. Öğr. Üyesi Gülşah TÜMÜKLÜ ÖZYER' e teşekkürlerimi borç bilirim.

Tez süresi boyunca manevi olarak her zaman beni destekleyen Açıköğretim Fakültesi'nde görev alan hocalarım ve arkadaşlarıma, eğitimim için hayatım boyunca yanımda olan sevgili aileme sonsuz teşekkürler.

Adem CANPOLAT

ÖZET

YÜKSEK LİSANS TEZİ

YAPAY ZEKA YÖNTEMLERİ İLE ATATÜRK ÜNİVERSİTESİ AÇIKÖĞRETİM FAKÜLTESİ SINAV SORULARININ ZORLUK DERECESİNİN TESPİTİ

Adem CANPOLAT

Danışman: Dr. Öğr. Üyesi Barış ÖZYER

Amaç: Atatürk Üniversitesi bünyesinde yer alan Açıköğretim Fakültesinde toplam 47 program yer almaktadır. Bu programlarda ortak dersler ve her programa özgü dersler olmak üzere toplam 420 ders bulunmaktadır. Bu programlara kayıtlı ortalama 200 bin öğrenciye her eğitim-öğretim yılında iki ara sınav, iki yarıyıl sonu sınavı, iki bütünleme sınavı ve bir üç ders sınavı olmak üzere toplam 7 sınav yapılmaktadır. Yapılan her sınav sonrasında öğrencilerin verdikleri cevaplar göz önüne alınarak soruların zorluk derecesi tespit edilmektedir. Bu tez kapsamında daha önce zorluk seviyeleri tespit edilen sorulardan yola çıkarak, henüz sınavlarda sorulmamış soruların zorluk derecesinin tespit edilmesi ve yapılacak sınavların zorluk derecesinin önceden belirlenebilmesi amaçlanmıştır.

Yöntem: Bu çalışmada 2 (iki) farklı veri seti oluşturuldu. Birinci veri setimizde tüm sorular kullanıldı. İkinci veri setimizde ayırt edicilik değeri yüksek sorulara yer verildi. Tüm veri setleri üzerinde durak kelimeler atıldıktan sonra iki farklı yöntem ile kelimeler eklerinden ayrıldı. Ön işlemi tamamlananan sorular üzerinde ikisi NLP tabanlı ve birisi genel metin sınıflandırma yöntemlerinden olmak üzere toplam 3(üç) farklı yöntem denedik. Veri setleri üzerinde sıklık dağılımı ile vektörize edilen sorular Levenshtein, Euclidean ve Manhattan algoritmaları ile ağırlıklandırıldı, ardından KNN sınıflandırıcısı ile sorular sınıflarına ayrıldı. Bir diğer vektörize yöntemi olarak TF-IDF algoritması ile soru metinleri vektör uzay modeline dönüştürüldü. Oluşan vektör uzay modelleri üzerinde Euclidean ve Manhattan algoritmaları ile uzaklık ölçümü yapıldı ve KNN ile sınıflandırıldı. Veri setlerimiz üzerinde bir başka sınıflandırma yöntemi olarak Naive Bayes sınıflandırıcı kullanıldı. NLP tabanlı yöntem olarak ise FastText ve BERT algoritmaları kullanılarak sınıflandırma yapıldı.

Bulgular: Soru benzerliklerinden yola çıkarak, oluşturulan yeni soruların zorluk derecesi tespit edilebilmektedir. Fakat ders bazlı soru sayısının düşük olması sınıflandırma başarısını olumsuz olarak etkilemektedir.

Sonuç: Yapılan çalışmalar neticesinde geleneksel soru ağırlıklandırma yöntemleri ile FastText algoritması benzer sonuçlar vermektedir. BERT algoritması bazı derslerde çok iyi sonuçlar yakalarken bazı derslerde başarısız sonuçlar ortaya çıkmaktadır. Birinci ve ikinci veri setlerimizde Naive Bayes sınıflandırıcısı dışında yapılan çalışmalarda bazı derslerde %100'lük başarı oranı yakalanmıştır. Ortalama başarı oranı ise birinci veri seti için %45, ikinci veri seti için %60 olarak gösterilebilir. Bazı derslerde ders ünitelerinin nadiren güncellenmesi ve hazırlanabilecek soru kalıplarının standart olması sebebiyle yüksek başarı oranı yakalandığı düşünülmektedir.

Anahtar Kelimeler: Soru Zorluğu, Metin Madenciliği, Makine Öğrenmesi, Metin Sınıflandırma, Metin Benzerliği

Eylül 2021, 71 sayfa

ABSTRACT

MS THESIS

DETERMINING THE DIFFICULTY LEVEL OF ATATÜRK UNIVERSITY OPEN EDUCATION FACULTY EXAM QUESTIONS BY ARTIFICIAL INTELLIGENCE METHODS

Adem CANPOLAT

Supervisor: Asst. Prof. Dr. Barış ÖZYER

Purpose: There are 47 programs in the Open Education Faculty of Atatürk University. In these programs, there are 420 courses in total, including common courses and courses specific to each program. A total of 7 exams, two midterm exams, two final exams, two make-up exams and one three course exam, are given to an average of 200 thousand students enrolled in these programs each academic year. After each exam, the difficulty level of the questions is determined by considering the answers given by the students. Within the scope of this thesis, it is aimed to determine the difficulty level of the questions that have not yet been asked in the exams and to determine the difficulty level of the exams based on the questions whose difficulty levels have been determined before.

Method: In this study, 2 (two) different data sets were created. All questions were used in our first data set. In our second data set, questions with high distinctiveness were included. After the stop words were removed on all data sets, the words were separated from their suffixes with two different methods. After the pre-processing was completed, we tried a total of 3 (three) different methods, two of which are NLP-based and one of the general text classification methods. The questions vectorized with frequency distribution on the datasets were weighted with the Levenshtein, Euclidean and Manhattan algorithms, then the questions were divided into classes with the KNN classifier. As another vectorization method, the question texts were transformed into a vector space model with the TF-IDF algorithm. Distance measurements were made with Euclidean and Manhattan algorithms on the resulting vector space models and classified with KNN. Naive Bayes classifier was used as another classification method on our datasets. Classification was made using FastText and BERT algorithms as a method based on NLP.

Findings: Based on the similarities of the questions, the difficulty level of the new questions can be determined. However, the low number of course-based questions negatively affects the success of the classification.

Results: As a result of the studies, traditional question weighting methods and FastText algorithm give similar results. While the BERT algorithm achieves very good results in some courses, there are zero percent unsuccessful results in some courses. In our first and second datasets, 100% success rate was achieved in some courses in studies other than the Naive Bayes classifier. The average success rate can be shown as 45% for the first data set and 60% for the second data set. It is thought that a high success rate has been achieved in some courses due to the fact that the course units are rarely updated and the question patterns that can be prepared are standard.

Keywords: Question Difficulty, Text Mining, Text, Machine Learning, Text Clustering, Text Smilarity

September 2021, 71 pages

İÇİNDEKİLER

TABLOLAR DİZİNİ

ŞEKİLLER DİZİNİ

SİMGELER ve KISALTMALAR DİZİNİ

Kısaltmalar

İGYS	İçerik Geliştirme ve Yönetim Sistemi
SBS	Soru Bankası Sistemi

GİRİŞ

Motivasyon

Bilgi teknolojilerinin hayatımıza girmesiyle beraber bilgisayar, tablet, telefon gibi cihazlar ve bu cihazların birbirleri ile iletişimini sağlayan en önemli ağlardan biri olan internet tüm insanların günlük yaşantısında temel bir ihtiyaç haline gelmiştir. Bu sayede bilgiye erişmek ve bilgiye katkıda bulunmak çok kolay bir hal almıştır. İnternetin yaygınlaşması ile birlikte binlerce gazete, dergi, televizyon ve radyo gibi kitle iletişim kuruluşları; üniversite, lise, ilkokul gibi eğitim ve araştırma kurumları; merkezi yönetimlerden taşra teşkilatlarına kadar devletin çeşitli organları; sayısız haber, ilan, duyuru, sınav, makale, kitap, sesli veya görüntülü yayın ürünlerini sunmak üzere web siteleri aracılığıyla bu iletişim ağına dâhil olmuşlardır. Bu denli geniş bir bilgi havuzunda istendik bilgiye ulaşmanın oldukça güç olduğu ise aşikârdır. Paylaşılan yazıların ilgili kişilere ulaşabilmesi için her bir yazının içeriğinin anlaşılması ve sınıflandırılabilmesi gerekmektedir. Bu gibi büyük çaplı ve karmaşık bir işi insan gücü ile gerçekleştirmek ekonomik olmadığı gibi oldukça zor bir iştir. Bu amaçla pek çok alanda insan gücünün yerini tutacak makine öğrenmesine ihtiyaç duyulmaktadır. Paylaşılan bir yazının konusu ve içerdiği duygu, gönderilen bir kısa mesajın reklam içerip içermediği, e-postaların istenmeyen posta olup olmadığı, sınavlarda sorulan bir sorunun daha önce hazırlanıp hazırlanmadığının tespiti gibi durumlar içinde yaşadığımız bilgi çağının sorduğu sorular haline gelmiştir. Bu sorunların çözümü için ise araştırmacılar tarafından birçok metin sınıflandırma yöntemi geliştirilmiştir. Geleneksel metin sınıflandırma yöntemleri uzun yıllar bu ihtiyacı karşılamış olmasına rağmen artık günümüzde oldukça gelişmiş sınıflandırma algoritmaları kullanılmaya başlanmıştır. Bunların başında Google tarafından yapılan aramalarda kullanıcıya en doğru sonucu göstermek üzere BERT algoritması kullanılırken, Facebook ekibi tarafından geliştirilen FastText algoritması da tercih edilen algoritmalar arasındadır. Bu gibi algoritmalar kullanıcı deneyimlerini birer veri olarak alıp modellerini eğiterek kullanıcının daha sonra yapacağı işlemlerde ihtiyaca yönelik sonuçları vermeyi hedeflemektedir.

Üniversiteler bünyesinde kurulmuş Açıköğretim fakülteleri her eğitim-öğretim yılında onlarca farklı bölümden yüzbinlerce öğrenciyi bünyesine katmaktadır. Bu öğrenciler öğrenim hayatları boyunca birçok sınava katılıp değerlendirilmektedirler. Sınavların güvenilirlik ve geçerlilikleri ise sınavların uygulanması neticesinde her bir soru için elde edilen korelasyonlar ve madde güçlük indeksleri yardımı ile belirlenmektedir. Bir sınavın güvenilirliği, özdeş gruplara farklı tarihlerde uygulanmış olsa dahi tutarlı ve kararlı olması ile belirlenir. Sınavın geçerliliği ise amaçlanan ölçmenin ne derece doğru ölçüldüğüyle ilgilidir. Bu amaçla soruların

zorluk düzeylerinin homojen bir dağılımla sınava yansıtılması büyük önem arz etmektedir.(Kili et al., n.d.)

Atatürk Üniversitesi Açıköğretim Fakültesi ülkemizde yer alan sayılı açıköğretim kurumlarından birisidir. Hâlihazırda yaklaşık 311 bin öğrencisi ve 200 bin mezununa açıköğretim sistemiyle yükseköğrenim vermenin yanı sıra sağladığı imkânlar sayesinde ülkemizdeki eğitim kalitesinin ve çeşitliliğinin arttırılmasında önemli bir rol üstlenmektedir. Bu amaçlar doğrultusunda hızlı bir şekilde büyümekte olan fakülte, her geçen gün program sayısını arttırmakta ve artan ders sayıları ile birlikte bu dersler için hazırlanan sınav sorularının zorluk derecesinin homojen olarak dağılması konusunda zorluk yaşamaktadır.

Eğitim dönemi boyunca öğrencilere pek çok değerlendirme sınavı yapılmaktadır. Her sınavda bir ders için çoktan seçmeli toplam 20 soru sorulmaktadır. Bu sebeple her bir sınav için binlerce soruya ihtiyaç duyulmaktadır. Sınav soruların hazırlık süreci pek çok aşamadan oluşmaktadır. Sınav sorularının çeşitlendirilebilmesi ve güncelliğinin sürdürülebilmesi için alanında uzman akademisyenler belirlenir. Belirlenen akademisyenler İçerik Geliştirme ve Yönetim Sistemi (İGYS) aracılığı ile yetkili oldukları dersin ünitelerine ulaşıp bu üniteler üzerinden istenen kriterlere göre soru hazırlamaktadırlar. Hazırlanan bu sorular İGYS üzerinden sisteme girilerek fakülteye gönderilir. Gönderilen soruların doğruluğunun ve güncelliğinin denetlenmesi için fakülte tarafından alanında uzman yeni akademisyenler belirlenir. Belirlenen akademisyenler tarafından tüm sorular denetlenir ve sorularda görülen hatalar soru hazırlayan uzmanların görüşü alınarak düzenlenir. Böylelikle soruların kalite standartı yükseltilir. Uzmanlar tarafından denetimi tamamlanan soruların dil denetim uzmanları tarafından Türkçe dilinde dil düzenlemeleri yapılır ve sınav sorusu formatına uygunluğu kontrol edilir. Tüm bu işlemlerden sonra sorular sınavda sorulabilir hale getirilir ve Soru Bankası Sistemi (SBS)'ne aktarılır.

Sınav hazırlıkları kapsamında sınavda sorulacak soruların tespiti için kriterler belirlenir. SBS' de bulunan sorular içerisinden her bir ders için sistem tarafından bu kriterlere uygun 20 (yirmi) soru seçilir.

Her sınav sonrası; sınavların güvenilirlik ve geçerlilikleri tespiti için, Ölçme-Değerlendirme konusunda alanında uzman akademisyenler tarafından belirlenen formüller ile öğrencilerin sorulara verdiği cevaplar göz önünde bulundurularak; sınav sorularının madde güçlüğü, ayırt ediciliği ve korelasyonu tespit edilmektedir. Yapılan bu analizler sonucu sorulan bazı soruların hatalı olduğu, yanlış bilgi ölçtüğü veya ünite içerisinde yeterince bahsedilmediği için öğrenciler tarafından cevaplandırılamadığı görülmekte ve bu sebeple analiz sonucunda bu soruların çok zor olduğu tespit edilmektedir. Bunun yansıra öğrenciler tarafından yapılan

itirazlar sonucu da bazı soruların hatalı olduğu tespit edilmektedir. Sınav öncesi yapılan pek çok değerlendirme aşamasına rağmen bu sorular denetleyenlerin gözünden kaçmakta ve öğrenci karşısına zor soru olarak çıkmaktadır. Makine öğrenmesi tekniklerini kullanarak ölçme-değerlendirme açısından yetersiz soruların sınavlarda sorulmasının önüne geçmek için bu konuda yapılmış çalışmalardan faydalandık. Yapılan bu çalışmada Atatürk Üniversitesi Açıköğretim Fakültesi 2018-2019 eğitim yılı sınav verileri kullanılmış, geleneksel metin sınıflandırma yöntemleri ile birlikte BERT ve FastText algoritmaları ile soruların zorluk dereceleri analiz edilerek, bu algoritmalar yardımıyla uygulanacak olan yeni sınavların geçerlilik ve güvenilirliğini artırmaya yönelik bir model geliştirilmeye çalışılmıştır.

Tezin Amacı ve Katkısı

Atatürk Üniversitesi Açıköğretim Fakültesi bünyesinde bulunan program sayısı ve ders sayısı sürekli artmaktadır. Bu kapsamda oluşturulan soru bankası her gün gelişmekte ve alanında uzman kişilerin çok sayıda soruyu kontrol etmesi ve soruların zorluk derecelerini tespit etmesi zorlaşmaktadır. Bu zor görev için herhangi bir çözüm bulunmamakta ve sınav sorularının zorluk derecesi ayarlanamamaktadır. Bu tez kapsamında sınavlarda sorulacak soruların sınav öncesinde analiz edilmesi ve zorluk dereceleri belirlenerek sınav kalitesinin yükseltilebilmesi amaçlanmaktadır. Bu amaç doğrultusunda tezin katkıları aşağıda sunulmuştur.

- Atatürk Üniversitesi Açıköğretim Fakültesi tarafından 2018-2019 Eğitim-Öğretim yılı içerisinde yapılan vize, final ve bütünleme sınavı soruları ve bu sorulara ait analiz değerleri alınarak veri seti oluşturuldu.
- Metin benzerlik algoritmaları yardımıyla soru metinleri ve doğru cevap şıkları arasındaki benzerlikler bulundu.
- K-NN ve Naive Bayes sınıflandırıcıları yardımı ile sorular zorluk derecelerine göre sınıflandırıldı.
- FastText ve BERT algoritması kullanılarak sorular üzerinde analizler yapıldı.
- Sorular etiketlenirken öğrenciler tarafından verilen yanıtlar göz önüne alındı ve her bir sorunun zorluk derecesi etiketlendi.

Yapılan tez çalışması sonucunda, soruların zorluk derecesinin önceden belirlenebilecek olması, sınavlarda ki dağılımın dengeli olmasını sağlayacaktır. Bunun sonucunda ders çalışan öğrencilerin başarı yüzdesi artacak ve eğitim kalitesi yükselecektir. Ayrıca bu iş için ekstra görevlendirme yapılmayacak ve bu görevi yerine getirmek için gerekli olan süreye ve maddi kaynağa daha fazla ihtiyaç duyulmayacaktır.

Çalışmanın kapsamı; literatür araştırması, açıköğretim sınav sistemi ile ilgili terimler , tez çalışmasında kullanılan sorular ile ilgili bilgiler, farklı yöntemlerle oluşturulan sınıflandırma sonuçlarının kıyaslanması ve elde edilen sonuçları içermektedir. Çalışmanın son bölümünde ise sonuçlar özetlenmiş ve gelecek çalışmalarla ilgili bilgiler verilmiştir.

KURAMSAL TEMELLER

Çoktan seçmeli soruların zorluğunun tespiti her zaman zorlu bir süreç olmuştur. Bunun için birden fazla manuel yöntem uygulanmaktadır. Alanında uzman kişilere sorular gösterilerek zorluk seviyeleri bu kişilerden alınabilir. Bu seçenek fazlasıyla öznel bir yaklaşımdır. Diğer bir seçenek olarak her sınavdan önce ön test yapılması düşünülebilir. Bu seçenekte küçük bir grup öğrenciye sorular sorulur ve her bir sorunun öğrenci yanıtlarına göre soru zorluk derecesi belirlenir. Bu seçenekte soruların sınavlardan önce açığa çıkma riski yer almaktadır. Uygulanan yöntemler hem zaman alıcı hem de güvenilir değildir. Bu sebeple bu problem için bilgisayar bilimlerinde çözüm aranmış ve bu zorlu süreç için geleneksel metin benzerlik algoritmaları önerilmiştir.

Hayatımızın pek çok alanında bilgisayarlar bizlere kolaylık sağlamaktadır. Yeni gelişen teknolojiler ile birlikte saatten bilekliğe, gözlükten şapkaya kadar hayatımız her alanında bilgisayarlar yer almaktadır. “Bilgisayarların konuştuğumuz dili anlaması, işlemesi, yorum yapması ve hatta cümle üretebilmesi” doğal dil işlemedir. Bugün hayatımız pek çok yerinde doğal dil işleme uygulamaları ile karşı karşıyayız. Web sitelerinde ki chat botlar, mesaj yazarken gelen otomatik cümle önerileri, telefonlarımızla evlerimize giren sanal asistanlar hatta Google Translate uygulaması bile birer doğal dil işleme örnekleridir. Metin sınıflandırması da doğal dil işlemenin bir alt dalıdır. İşlenen metinler çeşitli algoritmalar kullanılarak sınıflarına ayrılır. E-posta kutularına gelen e-maillerin spam kontrolü, internet ortamında yayımlanan bir haberin konusu, paylaşılan bir tweetin işlediği duygu, gelen bir kısa mesajın içeriği henüz kişiler bu metinleri okumadan tespit edilebilir ve sınıflandırılabilir. Veri miktarı artıkça bu verilerin bilgisayar yardımı olmaksızın ayrıştırılması ve yönetimi zorlaşmakta, faydalı olan içeriklere ulaşım azalmaktadır. Bu da verimliliğin düşmesine sebep olmaktadır. Metin sınıflandırması sayesinde verimlilik artırılmaktadır. Bir dersin sınavında sorulacak soruların zorluk derecesinin bilgiyi ölçmesi sebebiyle dersin başarım verimliliğini doğrudan etkilediği bilinmektedir. Metin sınıflandırması sayesinde sorular arasında ki uzaklıklar tespit edilecek ve sorular zorluk derecelerine göre sınıflandırılabilecek, zor olduğu tespit edilen sorular incelenerek bilgiyi ölçen soruların sınavlarda sorulması sağlanacaktır.

Literatür araştırmaları kapsamında metin sınıflandırma ve metin analizi çalışmaları incelenmiş, bu çalışmaların amaçları ve yerine getirdiği işlevler detaylı bir şekilde açıklanmıştır.

Yapılan bir çalışmada “Stackoverflow” ve “Yahoo! Answers” gibi topluluk soru cevaplama servislerindeki sorular ele alınmıştır. Soruların başlangıç zorluklarının tespitleri için sorulara verilen cevaplar içerisindeki en iyi olarak işaretlenen yanıtı hangi kullanıcının verdiğine bakılmıştır. Yanıtı veren kullanıcının site üzerindeki uzmanlık puanına göre sorular kolay, normal veya zor olarak etiketlenmiştir. Daha sonra her bir soru için etiket kelimeler tespit edilmiş ve bu etiket kelimeler sorunun zorluğunu tespit etmede kullanılmıştır.(Liu et al., 2013)

Başka bir çalışmada Anadili İngilizce olmayan öğrencilerin dinleme yeterliliğini değerlendirmek için oluşturulan sözlü metinlerin zorluğu tespit edilmeye çalışılmıştır. Anadili İngilizce olmayan 15 öğrenciye dağıtılan Likert ölçekli bir anket ile hazırlanan metinlerin zorlukları hakkında bilgi toplanmıştır. Farklı zorluk seviyelerini işaret eden üç öğrencinin değerlendirmelerinin ortalamaları alınmış ve regresyon modeli eğitilmiş ve 0.76’lık korelasyon hesaplanmıştır.(Yoon et al., 2016)

Çoktan seçmeli, boşluk doldurmalı ve kısa cevaplı soruların zorluk derecelerinin tespit edilmek istendiği başka bir çalışmada sorular ve soruya ait doğru cevap seçeneği kullanılmış ve yanlış cevaplar dikkate alınmamıştır. Metinler üzerinden durak kelimler çıkarılmış ardından Porter kök bulma algoritması ile kelime kökleri bulunmuştur. Vector Space modeli ile sorular vektörel olarak temsil edilmiştir. Ardından yapay sinir ağları aracılığı ile sınıflandırma işlemi yapılmıştır. Soruların %73’ü eğitim %27’si test için kullanılmıştır. (Fei et al., 2003)

Online soru-cevap forumlarına girilen soruların daha önceden eklenmiş sorular ile benzerlerinin araştırıldığı başka bir çalışmada Glove ile sorular vektörize edildikten sonra Evrişimli Sinir Ağları ile eğitilip test edilmiştir. %79 oranında başarı oranı yakalanmıştır. (Prabowo & Budi Herwanto, 2019)

Geleneksel Metin Benzerlik Algoritmaları ile Yapılan Çalışmalar

Levenshtein, Manhattan, Euclidean vs. gibi geleneksel bir takım metin uzaklık algoritmaları sayesinde metinler arasında ki benzerlik oranları tespit edilebilmektedir. Bu konuda pek çok çalışma gerçekleştirilmiştir.

Yapılan bir çalışmada, internetin hızlı büyümesi Arapça diline çevrilen mevcut çevrimiçi belgelerin sayısını artırdığı için bu belgeleri otomatik olarak düzenleyebilen ve sınıflandırabilen otomatik metin ve belge sınıflandırma sistemlerinin geliştirilmesine yol açmıştır. Bu sebeple N-Gram Frekans teknikleri ile metinler sınıflandırılmıştır. Metin benzerliğinin tespiti için Dice uzaklık algoritması ve Manhattan uzaklık algoritmaları kullanılmış ve belgeler sınıflarına göre etiketlenmiştir. Dice uzaklık algoritmasının, Manhattan

uzaklık algoritmasına göre daha iyi sınıflandırma sonuçları verdiği görülmüştür.(Khreisat, 2006)

Bir diğer çalışmada geliştirilen pek çok teknolojik aletin, acil ihtiyaçlarımızı anlamak ve karşılamak için yaşamımız hakkında düzenli olarak veri topladığı ve toplanan bu verilerin çok büyük boyutlara ulaştığı için sınıflandırılmasının zor olduğundan bahsedilmiştir. Bu problem karşısında toplanan verilerin Manhattan, Euclidean ve Chessboard algoritmaları ile ağırlıklandırılması yapılmış ve çıkan değerler K-Nearest Neighbour (KNN) algoritması ile sınıflandırılmıştır.(Wajeed & Adilakshmi, 2011)

Yapılan başka bir çalışmada ise sosyal bilimler alanında çoktan seçmeli test sorularının zorluğun ön test yapılmadan bulunması amaçlanmıştır. Bunun için her bir soru; soru metni, doğru cevap şıkkı ve çeldiriciler olarak ayırt edilmiş ve bunların her biri word2vec yardımı ile vektörlere dönüştürülmüş ve birbirleriyle arasında ki kosinüs benzerlikleri hesaplanmıştır. Hesaplanan değerler Z-skora dönüştürülmüş ve standart sapması alınmıştır. Ardından SVM sınıflandırıcısı yardımı ile zorluk derecesi tespit edilmiştir. Ön test yapılarak alınan sonuçlara göre daha iyi sonuçlar elde edilmiştir.(Hsu et al., 2018)

Yapılan başka bir çalışmada akademik yazılarda ki intihal oranları yakalanmaya çalışılmıştır. Eşanlamlı sözcüklerin tespiti için eşanlamlar sözlüğü kullanılmıştır. Levenshtein algoritması ile Smith Waterman algoritması birlikte kullanılarak hibrit bir yaklaşım denenmiştir. Levenshtein algoritması ile daha hızlı sonuç alınmıştır.(Z. Su et al., 2008) Bir başka çalışma da ise yazılan program kodlarında ki intihal oranı tespiti edilmeye çalışılmıştır. Bunun için XML ve Levenshtein algoritmaları kullanılmıştır. XML'in ağaç yapısı ile küçük değişikler yakalanmış, Levenshtein algoritması ile de mantıksal değişiklerin uzaklığı tespit edilmiştir.(Noh et al., 2006)

Bu çalışmalardan yola çıkarak hazırladığımız veri tabanımız üzerinde benzer çalışmalar gerçekleştirdik.

TF-IDF Algoritması ile Yapılan Metin Benzerlik Çalışmaları

Najran Üniversitesinde yer alan farklı programlarda ki birkaç derste sorulan soruların zorluk derecelerinin tespit edilebilmesi için yapılan bir çalışmada sorular TF-IDF ile vektörize edilerek Support Vector Machine, Naive Bayes ve K-NN sınıflandırıcıları ile sınıflandırılmıştır. Soruların %70'i eğitim %30'u test için kullanılmıştır.(Yahya et al., 2013)

Cep telefonlarına gelen kısa mesajlar (SMS) üzerinde yapılan bir çalışmada bilgisayar aracılığıyla gelen mesajın türü belirlenmeye çalışılmıştır. Bunun için gelen kısa mesaj TF-IDF algoritması ile ağırlıklandırılmış ve sınıflandırılmıştır.(Al-Talib & Hassan, 2013)

Başka bir çalışmada ise word2vec ve TF-IDF algoritmaları kullanılmıştır. Yapılan çalışma kapsamında cümleler her iki algoritma ile çeşitli kombinasyonlar denenerek sınıflandırılmıştır.(Lilleberg et al., 2015)

Sosyal medya verileri üzerinde yapılan bir çalışmada ise kısa metinler üzerinde TF-IDF algoritmasının tek başına iyi sonuçlar vermediği anlaşılmıştır. Bu sebeple TF-IDF algoritması ile Cosine, Euclidean ve Bray-Curtis algoritmaları birleştirilerek hibrit çalışma gerçekleştirilmiştir.(De Boom et al., 2016)

Yapılan bir çalışmada elektronik belgelerin içeriğinin tespit edilebilmesi amaçlanmıştır. Durak kelimeler çıkarıldıktan sonra kelime kökleri tespit edilmiş ve TF-IDF ile kelime ağırlıkları belirlenmiştir. Bunlara ek olarak iki yeni değer Confidence ve Support değerleri hesaplanmış ve bu değerler üzerinden kelimeler kategorize edilmiştir.(Y. T. Zhang et al., 2005)

FastText Algoritması ile Yapılan Metin Benzerlik Çalışmaları

Twitter verileri kullanılarak yapılan bir çalışmada, atılan tweet metinleri üzerinden kişinin grip olup olmadığı araştırılmıştır. Bunun için FastText algoritması kullanılmış ve yüksek doğruluk oranı yakalanmıştır.(Alessa, 2018)

Bir diğer çalışmada önceden internette artan belgelerin Tibet dilinde sınıflandırılabilmesi için eğitilmiş FastText vektörleri kullanılarak cümlelerin kelime değeri vektörleri çıkarılmış ve ortalaması alınmıştır. Ortaya çıkan yeni vektörler SVM üzerinde sınıflandırılmıştır.(Ma et al., 2019)

Güney Asya'da sosyal medyada yaygın olarak kullanılan bir dil olan Roman Urduca dili için zararlı yorumların tespit edilmesi üzerine yapılan bir çalışmada FastText, Glove ve Word2Vec algoritmaları kullanılmıştır. Oluşturulan model sonrası %86,35'lik doğruluk oranı yakalanmıştır.(Saeed et al., 2021)

Web sitelerini, alan adları ve ağ trafikleri izleyerek sınıflandırmak istenen bir çalışmada veriler FastText ile vektörel hale getirildikten sonra SVM ile sınıflandırılmıştır.(Faroughi et al., 2021)

Türkçe haber metinlerinin sınıflandırılması için yapılan bir çalışmada çeşitli haber sitelerinden alınan veri kümeleri üzerinde metin sınıflandırma çalışması yapılmıştır. Öncelikli olarak haber metinleri ön işlemeden geçirilmiş ve gövdelenmiştir. Ön işlemeden geçirilen metinler Tfidfvectorizer, Word2Vec ve FastText yöntemleri ile ayrı ayrı vektörize edildikten sonra Destek Vektör Makinesi (Support Vector Machine, SVM), Naive Bayes, Logistic Regression, Random Forest ve Yapay Sinir Ağı (Artificial Neural Network, ANN) yöntemleri

ile sınıflandırılmıştır. Yapılan çalışma sonucuna göre en yüksek başarı oranı %95,75 ile FastText yöntemi ve vektör modeli ile elde edilen metnin SVM ile sınıflandırılmasından elde edilmiştir. (ÇELİK & KOÇ, 2021)

Yapılan başka bir çalışmada dergilere veya konferanslara gönderilen bilimsel makalelerin sınıflandırılması yapılmıştır. Sınıflandırma için makalede yer alan başlık, özet, anahtar kelimeler, başlık + anahtar kelime, başlık + özet, anahtar kelime + özet ve başlık + anahtar kelime + özet şekline 7(yedi) başlık ele alınmıştır. Yapılan modelleme sonucunda en iyi sonuç CNN + FastText modelinde ortaya çıkmıştır.(Nguyen et al., 2021)

BERT Algoritması ile Yapılan Metin Benzerlik Çalışmaları

Yapılan bir çalışmada geleneksel metin sınıflandırma yöntemleri ile BERT karşılaştırılmıştır. IMDB üzerinden indirilen 50000 yorum pozitif ve negatif olarak sınıflandırılmak istenmiştir. Verilerin %50 si eğitim için kullanılmış ve TF-IDF ile vektörize edildikten sonra Voting Classifier, Logistic Regression, Linear SVC, Multinomial NB, Ridge Classifier, Passive Aggresive Classifier sınıflandırıcıları ile sınıflandırılmıştır. BERT ile yapılan sınıflandırma sonucunda 0.9387'lik doğruluk elde edilirken buna en yakın sonucu Voting Classifier 0.9007 ile yakalamıştır. BERT bir dil işleme çözümü olarak geleneksel metin sınıflandırma algoritmaları kadar karmaşık işlemler gerektirmeden daha iyi sonuç vermiştir. (Garrido-Merchán & González-Carvajal, 2020)

Yapılan bir diğer çalışmada atılan tweet üzerinde bahsedilen felaketin gerçek olup olmadığı sınıflandırılmıştır. Veri setinin %75'i eğitim için kullanılmış, sınıflandırma işlemi H2OAutoML ve BERT ile yapılmıştır. BERT ile 0.8361 doğruluk elde edilirken H2OAutoMLile 0.7875'lik doğruluk yakalanmıştır. (Garrido-Merchán & González-Carvajal, 2020)

BERT ile İngilizce dili üzerinde yakalanan yüksek başarıların diğer diller üzerinde de test edilebilmesi için yapılan bir çalışmada Portekiz haber sitesinden alınan 9 farklı sınıftaki haber metinleri sınıflandırılmak istemiştir. Veri setinin %75'i eğitim için kullanılmış, sınıflandırma işlemi AutoML ve BERT ile yapılmıştır. BERT ile 0.9093 doğruluk elde edilirken AutoML ile 0.8480 doğruluk yakalanmıştır. (Garrido-Merchán & González-Carvajal, 2020)

Yine farklı bir dil üzerinden yapılan sınıflandırmada Çince kullanılmıştır. Farklı sembollerin yer alması ve boşluklarla kelimelerin ayrılmaması sebebi ile BERT'in kötü sonuç vereceği varsayılmıştır. Yapılan olumlu ve olumsuz otel yorumları sınıflandırılmak istemiştir. Veri setinin %85'i eğitim için kullanılmış, sınıflandırma işlemi AutoML ve BERT ile yapılmıştır. BERT ile 0.9381 doğruluk elde edilirken AutoML ile 0.7399 doğruluk

yakalanmıştır. Dil özelliklerinden bağımsız olarak BERT'in klasik metin sınıflandırma yöntemlerinden daha iyi performans gösterdiği gözlemlenmiştir. (Garrido-Merchán & González-Carvajal, 2020)

Yayınlanan haberlerin sahte mi gerçek mi olduğunun araştırıldığı bir çalışmada birden fazla veri seti kullanılmıştır. Veri setleri üzerinde durak kelimeler atılıp kelime kökleri tespit edildikten sonra geleneksel metin sınıflandırma yöntemleri (SVM, LR, k-NN, Naive Bayes, AdaBoost, Decision Tree), derin öğrenme yöntemleri (CNN, LSTM, Bi-LSTM,C-LSTM, HAN, Conv-HAN) ve önceden eğitilmiş gelişmiş dil modelleri (BERT, RoBERTa, DistilBERT, ELECTRA, ELMo) uygulanmış ve çıkan sonuçlar karşılaştırılmıştır. BERT ve RoBERTa 0.62 ile en yüksek başarı oranını yakalarken, veri sayısının çok az olduğu veri setlerinde RoBERTa daha iyi sonuçlar vermiştir.(Khan et al., 2021)

Zemberek Kütüphanesi

"Zemberek, açık kaynak kodlu Türkçe Doğal dil işleme kütüphanesi ve OpenOffice, LibreOffice eklentisidir. İlk sürümü BSD lisansı ile dağıtılmıştır. Tamamen Java ile geliştirilen kütüphane, yazım denetimi, hatalı kelimeler için öneri, heceleme, deascifier, hatalı kodlama temizleme gibi işlevlere sahiptir. Zemberek2 kodlu ikinci sürümünde MPL lisansına geçilmiş, genel olarak tüm Türk dilleri için bir DDİ altyapısı oluşturulması için gerekli mimari değişiklikler yapılmıştır. Zemberek kullanılarak yazılmış bir sunucu Pardus için genel yazım denetimi desteği vermektedir. Sunucu TCP-IP soketleri üzerinden ISpell benzeri basit bir protokolle diğer uygulamalarla haberleşmektedir, yeni sürümünde DBUS ara yüzü de sunucuya eklenmiştir. Zemberek kütüphanesinin .net sürümünü oluşturmak üzere NZemberek projesi başlatılmıştır.

Zemberek kütüphanesi ve LibreOffice eklentisi Java dilinde yazıldığı için platform bağımsızdır. 2005 yılında LKD 4. Linux ve Özgür Yazılım şenliğinde yılın en iyi özgür yazılımı ödülünü almıştır. Günümüzde ise (2016) İstanbul Kültür Üniversitesi Bilgisayar Mühendisliği bitirme projesiyle katkıda bulunan Berk Dindaroğlu ve Nazlı Ceren Çoğalgil projenin geliştirilmesinde katkıda bulunmaktadırlar."(Akın & Akın, 2007)

MATERYAL ve YÖNTEM

Materyal başlığı altında, Atatürk Üniversitesi Açıköğretim Fakültesi bünyesinde yer alan programlar ve bu programlara ait ders sayıları açıklandı. Ayrıca tez çalışmasında kullanılan soru sayılarına ve soruların zorluk derecelerinin belirlenerek etiketlenme sürecine yer verildi.

Yöntem başlığı altında ise Açıköğretim Fakültesine ait sınav sorularının zorluk derecesinin anlamlı bir şekilde elde edilebilmesi için sorular düzenlendi. Düzenlenen sorular üzerinden özellik çıkarımı(Feature Extraction) yapıldı. Metin sınıflandırma konusunda eskiden beri kullanılan Euclidean, Manhattan, Levenshtein algoritmaları uzaklık ölçümü yapılarak KNN ve Naive Bayes ile sorular sınıflandırıldı. Bir diğer yöntem olan TF-IDF algoritması ile de soru metinleri vektörlere dönüştürüldü ve bu vektörler üzerinden KNN ile sınıflandırma yapıldı. Bunların yanında yeni bir yöntem olarak FastText algoritması denendi.

Materyal

Veri Kümesi

Açıköğretim Fakültesi'nde yer alan ve Tablo 1'de gösterilen programlara ait toplam 420 ders bulunmaktadır. Bu derslere ait sorular belirli aralıklarla alanında uzman akademisyenler tarafından hazırlanır ve fakülteye teslim edilir. Böylelikle her yıl soru sayısının artışı sağlanmaktadır.

Tablo 1. Programlar ve Bu Programlara Ait Ders Sayıları

Program Adı	Ders Sayısı
Acil Durum ve Afet Yönetimi Önlisans Programı	28
Acil Yardım ve Afet Yönetimi Lisans Tamamlama Programı	24
Adalet Önlisans Programı	29
Bankacılık ve Sigortacılık Önlisans Programı	31
Bilgi Yönetimi Önlisans Programı	26
Bilgisayar Programcılığı Önlisans Programı	27
Büro Yönetimi ve Yönetici Asistanlığı Önlisans Programı	28
Çağrı Merkezi Hizmetleri Önlisans Programı	28
Çocuk Gelişimi Önlisans Programı	28
Dış Ticaret Önlisans Programı	31
Emlak ve Emlak Yönetimi Önlisans Programı	28
Fotoğrafçılık ve Kameramanlık Önlisans Programı	28
Halkla İlişkiler ve Tanıtım Lisans Programı	50

Tablo 1. (Devamı)

Halkla İlişkiler ve Tanıtım Önlisans Programı	28
İlahiyat Önlisans Programı	30
İş Sağlığı ve Güvenliği Lisans Tamamlama Programı	24
İş Sağlığı ve Güvenliği Önlisans Programı	28
İşletme Lisans Programı	52
İşletme Yönetimi Önlisans Programı	30
Kamu Yönetimi Lisans Programı	51
Laborant ve Veteriner Sağlık Önlisans Programı	28
Lojistik Önlisans Programı	28
Menkul Kıymetler ve Sermaye Piyasası Programı	23
Özel Güvenlik ve Koruma Önlisans Programı	26
Radyo ve Televizyon Programcılığı Önlisans Programı	28
Reklamcılık Lisans Programı	49
Reklamcılık Önlisans Programı	28
Sağlık Kurumları İşletmeciliği Önlisans Programı	28
Sağlık Yönetimi Lisans Programı	52
Sağlık Yönetimi Lisans Tamamlama Programı	24
Sivil Hava Ulaştırma İşletmeciliği Önlisans Programı	28
Sosyal Hizmet Lisans Programı	48
Sosyal Hizmet Lisans Tamamlama Programı	23
Sosyal Hizmetler Önlisans Programı	32
Sosyoloji Lisans Programı	51
Tıbbi Dökümantasyon ve Sekreterlik Önlisans Programı	28
Turizm ve Otel İşletmeciliği Önlisans Programı	28
Turizm ve Seyahat Hizmetleri Önlisans Programı	28
Yeni Medya ve Gazetecilik Önlisans Programı	28
Yerel Yönetimler Önlisans Programı	28

Açıköğretim Fakültesi'nde iki farklı soru tipi mevcuttur. Birinci tip sorular yukarıda da bahsedilen soru hazırlama uzmanlarının ders içiriklerinden yararlanarak hazırladığı sorulardır. Diğer soru tipi ise her dersin her ünitesinin sonunda yer alan değerlendirme sorularıdır. Bu sorular zaten ünitelerde yer aldığı için bu sorulara öğrenciler tarafından kolaylıkla ulaşılabilmektedir.

Sınav soruları hazırlanırken zorluk derecesi, soruların uzunluğu, ünite dağılımı, soru tipi, bir sorunun daha önceki sınavlarda kullanılma durumu gibi kriterler her sınav öncesinde ilgili komisyon üyeleri tarafından belirlenmektedir. Bir sorunun ünitesi, tipi, daha önce ki sınavlarda kullanılma durumu ya da uzunluğu kişiye göre değişmemektedir. Fakat sorunun zorluk derecesi soruyu hazırlayan kişinin öznel görüşü olduğu ve öğrenci bazlı değerlendirilemediği için kullanışlı değildir.

Her sınav sonrasında Ölçme-Değerlendirme uzman görüşü desteğiyle sınav soruları analiz edilmektedir. Bir sorunun madde güçlüğü, ayırt ediciliği ve nokta çift serili korelasyonu

hesaplanmaktadır. Soruyu doğru cevaplayan öğrenci sayısı **X,** testi alan öğrenci sayısı **N** olsun. Madde güçlüğü **p**;

$$p = \frac{X}{N} \text{ ile ifade edilir.} \quad (1)$$

Madde güçlüğü bir sorunun zorluk derecesini ifade eder. Oluşan madde güçlüklerine göre sorular etiketlenmektedir. Tablo 2'de gösterildiği gibi madde güçlüğü 0.3'ten küçük olan zorular ZOR, 0.3'e eşit ve 0.7'den küçük olan sorular ORTA, 0.7'e eşit ve büyük olan sorular ise KOLAY kabul edilmektedir. (D'Sa & Visbal-Dionaldo, 2017)

Tablo 2. Madde Güçlüğü ve Etiket İlişkisi

P	<0.3	>=0.3 ve <0.7	>=0.7
Etiket	Zor	Kolay	Orta

Ölçme-Değerlendirme uzmanlarına göre sorulan bazı sorular etkili ayırt ediciliğe sahip olup sınavda sorulması uygunken, bazı sorular ise etkili ayırt ediciliğe sahip olmayıp düzeltilmesi gereken problemli sorulardır ve sorulması uygun değildir. Bir sorunun etkili ayırt ediciliğe sahip olduğunun belirlenebilmesi için Denklem 2' de belirtilen Madde ayırıcılık indeksi formülü ve Denklem 3' te belirtilen Nokta Çift Serili Korelasyon formülü kullanılmaktadır. **Madde ayırıcılık indeksi, *D***

Bir derste alınan sınav puanlarına bakılarak dersin sınavına giren öğrenciler aldıkları puanlara göre büyükten küçüğe veya küçükten büyüğe doğru sıralanır. Sıralamaya göre en üstte yer alan %27'lik dilime giren öğrenciler ile en altta yer alan %27'lik dilime giren öğrenciler **üst grup** ve **alt grup** adında iki grup oluşturur. Soruyu üst grupta doğru cevaplayan oranı **Üst**, soruyu alt grupta doğru cevaplayan oranı **Alt** olsun. Buna göre;

$$D = Üst - Alt \quad (2)$$

Denklem 2 Madde Ayırıcılık İndeksi

Nokta çift serili korelasyon

Soruyu doğru cevaplayanların test ortalaması O, bütün grubun test ortalaması B, testin standart sapması S olsun. Buna göre Nokta çift serili korelasyon;

$$Nokta\ çift\ serili\ korelasyon = \frac{O-B}{S}\sqrt{\frac{p}{1-p}} \quad (3)$$

Denklem 3 Nokta Çift Serili Korelasyon Formülü

Madde ayırıcılık indeksi (D) 0.40'tan büyük, aynı zamanda Nokta çift serili korelasyon değeri de $\left(\frac{2}{\sqrt{N-1}}\right)$'e eşit veya büyük olan sorular etkili ayırt ediciliğe sahiptir ve sınavlarda sorulmalıdır. **N** sınava giren öğrenci sayısını ifade etmektedir. (Y. Zhang et al., 2010)

Ölçme-değerlendirme formüllerinden yola çıkarak etkili ayırt ediciliğe sahip sorular belirlenir. Daha önce ki sınavlarda çıkmış sorular veya değerlendirme soruları özellikle çalışan öğrenciler tarafından rahatlıkla cevaplandırılacağı için ayırt edici katsayıları pozitif çıkmaktadır. Tüm sorular birinci veri setimizi oluştururken, etkili ayırt ediciliğe sahip sorular ise ikinci veri setimizi oluşturmaktadır. Veri setimizde ki derslere ait özet bilgi Tablo 3'de gösterilmiştir. Tüm derslere ait veriler ise Ekler bölümünde yer alan Tablo 11 'de yer almaktadır.

Sınav sorularında doğru cevap şıkkı soruyu tanımlamada önemli bilgiler içerdiği için tüm veri setlerimizde soru metni ve doğru cevap şıkkı birleştirilerek tek bir metin olarak düşünülmüştür. Örnek olarak; "Disiplin cezalarının da diğer cezalar gibi kanunda düzenlenmesi aşağıdakilerden hangisi ile ilgilidir?" soru metninin doğru cevabı; "Kanunilik (kanunsuz suç ve ceza olmaz)" ilkesidir. Bu soruyu temsilen ""Disiplin cezalarının da diğer cezalar gibi kanunda düzenlenmesi aşağıdakilerden hangisi ile ilgilidir? Kanunilik (kanunsuz suç ve ceza olmaz) ilkesi" metni kullanılmıştır.

Tablo 3. Birinci ve İkinci Veri Setimize Ait Örnek Dersler ve Soru Sayıları

	Soru Sayısı	
Dersin Adı	**Birinci Veri Seti**	**İkinci Veri Seti**
Acil Durum ve Afet Psikolojisi	39	13
Acil Durum ve Afet Yönetimine Giriş	59	24
Acil Hasta Bakımı	39	14
Acil Servis Araçları Eğitimi	42	11
Acil Yardım ve Kurtarma Çalışması	39	13
Adalet Meslek Etiği	55	24
Adli Sosyal Hizmet	70	25
Afet ve Acil Durum Mevzuatı	57	28
Afetlerde Haberleşme	56	21
Afetlerde Halk Sağlığı Hizmetleri	61	19
Afetlerde İlk ve Acil Yardım Yönetimi	40	12

Ders bazlı sorular incelendiğinde bazı derslerde etiketlerin dengeli dağılmadığı görülmüştür. Sebep olarak dersin sorularının sürekli olarak çok zor veya orta derecede hazırlandığı tespit edilmiştir. Bu derslere ait ortalama zorluk dereceleri Tablo 5'de gösterilmiştir. Birinci veri setimizde 17.036 soru bulunurken ikinci veri setimizde 8.578 soru bulunmaktadır. Birinci veri setimizde 7.671 zor, 6.014 orta ve 3.351 kolay soru bulunurken, ikinci veri setimizde ise 2.538 zor, 5.147 orta ve 893 kolay soru bulunmaktadır. Tablo 4'te veri setlerinde yer alan etiket dağılımları gösterilmiştir.

Tablo 4. Veri Setlerinde Bulunan Etiketlere Göre Soru Sayıları

	Soru Sayısı	
Etiket	1.Veri Seti	2.Veri Seti
Kolay	3351	893
Orta	6014	5147
Zor	7671	2538

Tablo 5. Ortalama Zorluk Dereceleri En Düşük ve En Yüksek Dersler

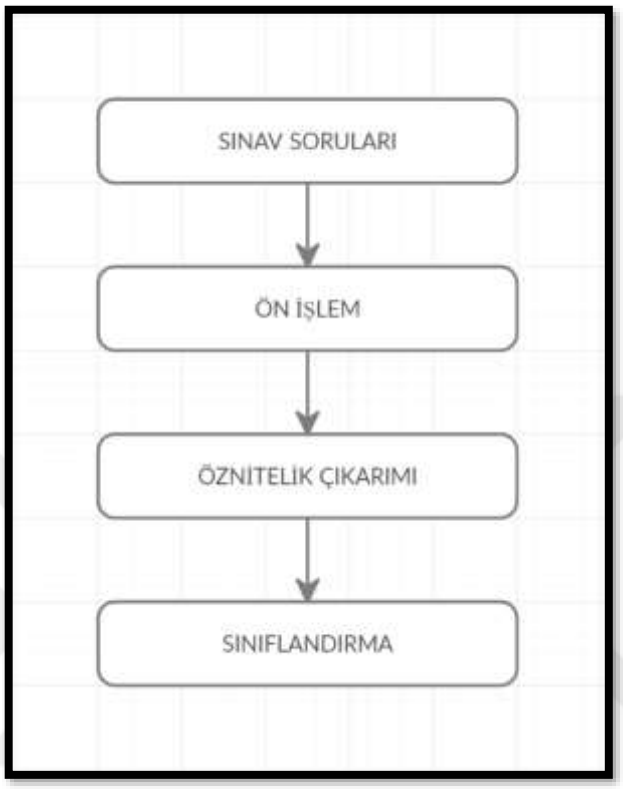
Dersin Adı	**Ortalama Zorluk Derecesi**
İstatistik Analiz	0,17
Veteriner Anatomi	0,21
Veteriner Biyokimya	0,24
Veteriner Mikrobiyoloji ve İmmünoloji	0,24
İlk Dönem İslam Tarihi	0,25
Mikro İktisat	0,26
Suç Sosyolojisi	0,26
Temel Kredi İşlemleri	0,27
Veteriner Histoloji ve Embriyoloji	0,27
Dönem Sonu Muhasebe İşlemleri	0,28
Veri Tabanı Yönetim Sistemleri	0,28
Makro iktisat	0,28
Harita Bilgisi ve Coğrafi Bilgi Sistemleri	0,29
Haberleşme ve Seyrüsefer Sistemleri	0,29
İslam Sanatları Tarihi	0,3
Para ve Bankacılık	0,3
Hadis Tarihi ve Usulü	0,3
İslam Kurumları ve Medeniyeti	0,3
Sosyal Hizmet Kuram ve Yaklaşımları	0,3
İstatistiğe Giriş	0,3
Uluslararası İktisat	0,3
Temel Anatomi	0,3
Borçlar Hukuku	0,31
Veteriner Parazitoloji	0,31
Sosyal Hizmet Tarihi	0,31
Çağdaş Sosyoloji Kuramları	0,31

Tablo 5. (Devamı)

Havacılık Kuralları	0,31
İslam Mezhepleri Tarihi	0,31
Sosyoloji Tarihi	0,31
Klasik Sosyoloji Kuramları	0,31
Türk İslam Edebiyatı	0,31
Laboratuvar Teknikleri ve Prensipleri	0,31
Gümrük Mevzuatı	0,31
Makina ve Teçhizat	0,31
İslam Düşünce Tarihi	0,32
Havaalanı Donanım ve Faaliyetleri	0,32
Kriminoloji	0,32
Vergi Hukuku	0,32
Risk Yönetimi	0,32
Siyaset Bilimi	0,32
Sermaye ve Para Piyasaları	0,32
Sosyal Antropoloji	0,33
Temel Fotoğrafçılık	0,33
Kelama Giriş	0,33
Maliyet Muhasebesi II	0,33
Kamu Yönetimi	0,33
Medeni Hukuk II	0,33
İslam Hukukuna Giriş	0,33
Şirketler Muhasebesi	0,33
Dış Ticaret Finansmanı	0,33
Mali Tablolar Analizi	0,33
İşletme Finansmanı	0,33
Turizm Ekonomisi	0,34
Genel Muhasebe	0,34
Hukukun Temel Kavramları	0,34
Özel Eğitim I	0,34
Zootekni ve Genetik	0,34
Atatürk İlkeleri ve İnkılap Tarihi I	0,34
Biyoistatistik	0,34
İnsan ve Toplum Felsefesi	0,34
İktisada Giriş	0,34
Dış Ticaret Belgeleri	0,35
Sivil Havacılığa Giriş	0,35
Endüstri Sosyolojisi	0,35
Şehircilik	0,35
Banka Muhasebesi	0,35
Gündelik Yaşam Sosyolojisi	0,35
Temel Sigortacılık ve Risk	0,35
Blok Uygulama	0,71
Blok Uygulama II	0,73

Yöntem

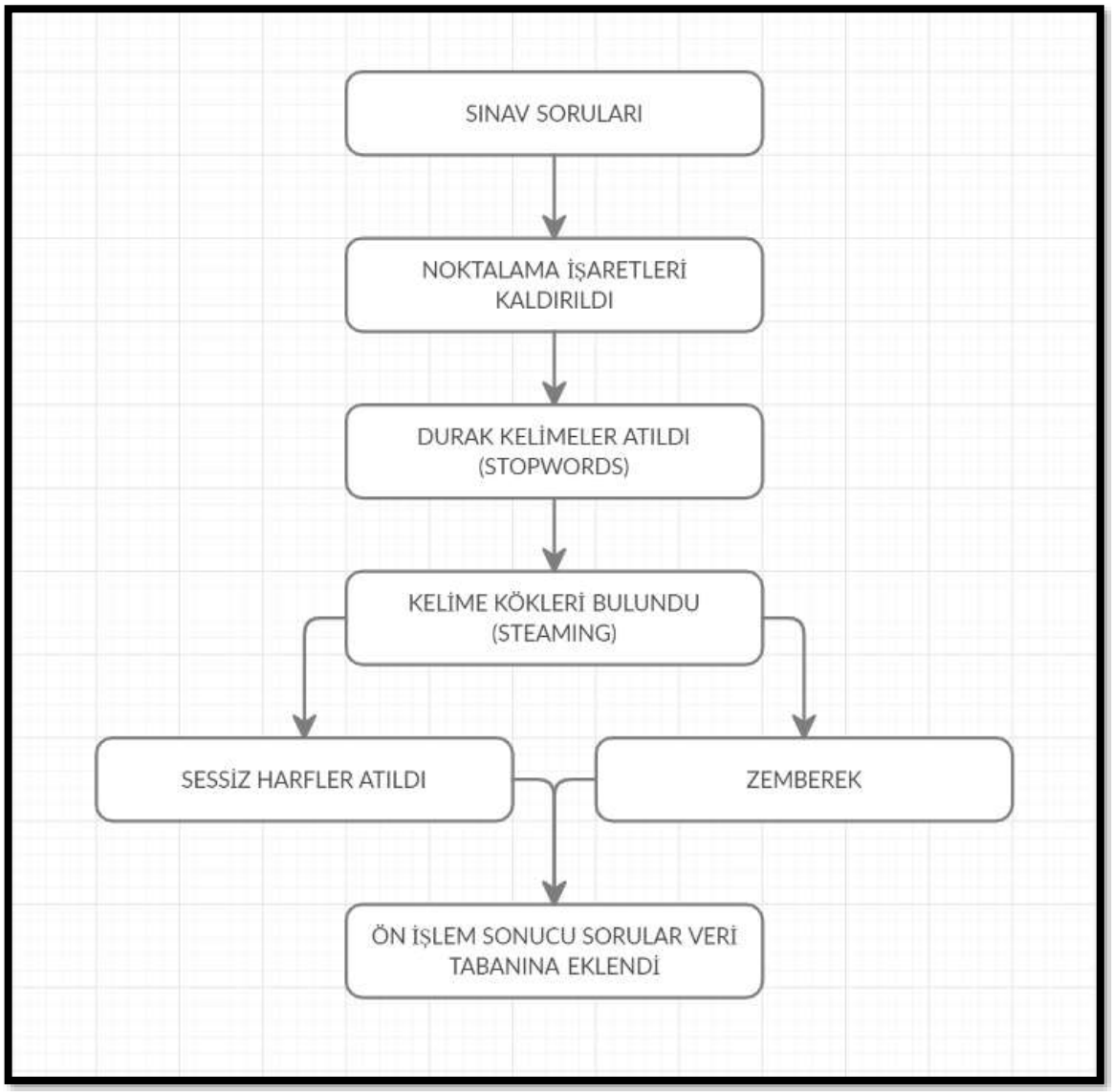
Önerilen yöntem Şekil 1'de görüleceği üzere 3 temel aşamadan meydana gelmektedir. Veri setine eklenen sorular ön işlemden geçirilir. Ardından temizlenen soruların öznitelik vektörleri çıkarılır ve daha sonrasında sınıflandırma işlemi gerçekleştirilir.



Şekil 1. Uygulanan yöntemler

1. Ön İşlem

Sınıflandırma işlemine geçmeden önce her bir veri setimizde bulunan verilerin bir takım ön işlemlerden geçmesi gerekmektedir. Noktalama işaretlerinin kaldırılması, kelime köklerinin bulunması ve durak kelimelerin atılması ön işlem adımlarıdır. Ön işlem sayesinde kelime boyutları düşecek, anlamsız kelimeler ve noktalama işaretleri kaldırıldığı için veri seti boyutu ciddi derecede azalacaktır. Böylece sınıflandırma işlemleri daha hızlı ve daha iyi performans gösterecektir.



Şekil 2. Ön işlem adımları

Noktalama işaretlerinin kaldırılması

Soru metinlerinde geçen nokta, virgül, noktalı virgül, soru işareti gibi işaretler kelime kök bulma işlemine geçilmeden önce sorulardan kaldırıldı.

Durak kelimelerin atılması (Stopwords)

Soru metinlerinde geçen her kelime değerlendirme için uygun değildir. Türkçe ’de tek başına anlamlı olmayan fakat cümleye dahil edildiğinde diğer kelimeleri anlam olarak bütünleyen edat ve bağlaçlar mevcuttur. Bu kelimeler oldukça sık kullanıldığı ve soru üzerinde herhangi bir ayırt ediciliğe sahip olmadıkları için soru metinlerinden çıkarılmaktadır. Diğer yandan her soruda geçen ve soruyla konu bakımından ilgili olmayan birtakım kelimeler mevcuttur. Bu kelimeler soru metinlerinin daha uzun olmasını sağlayacak hem de anlamsal olarak bir bağ taşımayacaklardır. Bu sebeple sınıflandırma performansı düşecektir. Bunların

yansıra soruların kalın, italik, altı çizili vb. şekilsel özelliklerini betimleyen birtakım etiketler vardır. Bu etiketler soruları şekilsel olarak düzenler ve soru zorluğuna etki etmez. Dolayısıyla bu etiketlerde sorulardan çıkarılmaktadır. Ön işlem adımlarında sorulardan çıkarılan bazı durak kelimeler Tablo 6'da gösterilmiştir.

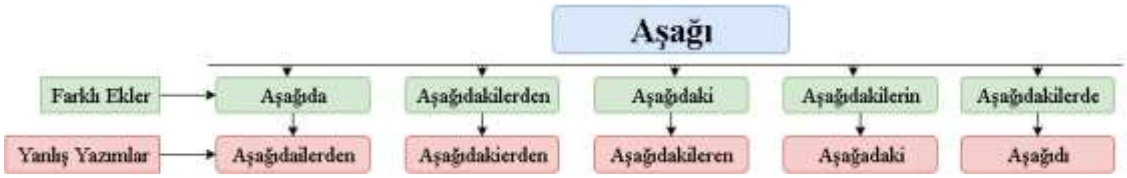
Tablo 6. Durak Kelime Örnekleri

Örnek Durak Kelimeler		
hangisi	özellikleri	III
bir	olmalıdır	IV
ve	neden	konusunda
ile	nasıl	çıkan
olarak	vardır	olur
aşağıdakilerden	verilmiştir	arası
da	belli	gerekir
veya	önce	doğrudan
hangileri	yapmak	hem
yukarıdakilerden	başka	alt
yanlıştır	herhangi	hâlinde
getirilmelidir	çeşitli	bütün
cümlede	yoktur	edilir
biridir	Yalnız IV	tam
kaç	Yalnız III	uzun

Kök bulma (Steaming)

Türkçe'de aldığı eklere göre kelimeler farklılaşmaktadır. Pek çok ek yapısı bulunmaktadır ve bir kökten onlarca farklı kelime türetilebilmektedir. Sınıflandırma öncesi yapacağımız ağırlıklandırma işlemleri kelime bazlı olacağı için alınan her yeni ek aslında aynı değere sahip olması gereken kelimelerin farklı değerler almasını sağlayacaktır. Bu sebeple kelimeler üzerinden yapacağımız sınıflandırmalarda kök bulma işlemi yapılmadan sınıflandırma yapmak kötü performans göstermektedir.

Şekil 3'de örneği görülen kelimeyi ele alalım. "Aşağı" kelimesi sorularda sıklıkla geçmektedir fakat kelime aldığı eklerden dolayı farklılaşmaktadır. Bu haliyle her biri farklı kelime olarak işaretlenecek ve sınıflandırma kötü bir performans sergileyecektir. Bu sebeple kelimenin köküne inme ihtiyacı bulunmaktadır. Aynı zamanda metin sınıflandırmasında büyük verilerle çalışmak zaman açısından problem olmaktadır. Her bir kelimenin temizlenmesi ve kök bulma işleminin gerçekleştirilmesi kelimeleri kısaltacak ve oluşacak özellik vektörlerinin bellekte daha az yer kaplaması sağlanmış olacaktır. Bu sayede yapılan işlemler hızlanacaktır. Türkçe kelimelerde kök bulma işlemi henüz çözülememiş bir problemdir. Bu konuda iki farklı yöntem denedik.



Şekil 3. Aşağı kelimesinin yazım örnekleri

Zemberek kütüphanesi ile kök bulma

Zemberek kütüphanesini kullanarak Visual Studio ortamında C# dili ile Windows Form uygulaması hazırladık. Veri setlerimizde bulunan ve soru kökleri ile doğru cevapların birleştirildiği sorularımızın tamamı hazırlamış olduğumuz uygulamada işlendi. Sorularda ki kelimelerin kökleri bulundu ve hatalı kelimeler düzeltildi. Bu sayede kelimelerin kısaltma işlemi tamamlanmış oldu. Bulunan kelime kökleri birleştirilerek yeni soru metni hazırlandı. Bu sayede farklı sorularda aynı köke sahip ek alarak farklılaşmış kelimeler aynı kelime olarak düzenlenmiş ve sorular arasında ilişki kurabilmek kolaylaştırılmış oldu.

Sesli harflerin atılması (dört harf)

Yapılan bir çalışmada Türkçe kelimelerde sessiz harflerin sesli harflere göre daha önemli olduğu tespit edilmiştir. Bu bilgiden yola çıkarak kelimler de ki sesli harfler atılmış ve ilk iki, üç veya dört harfleri kelime olarak kabul edilmiş ve test edilmiştir.(Çataltepe et al., 2007) Bu makaleden yola çıkarak tüm sorular daha önce hazırlamış olduğumuz Windows Form uygulamamız da işlendi ve kelimelerde yer alan tüm sesli harfler atıldı. Yeni oluşan kelimelerin ilk dört harfi yeni bir kelime olarak kabul edildi ve oluşan bu yeni kelimeler ile soru metinleri yeniden yazıldı.

Yukarıda bahsedilen her iki yöntemde de kelimenin aldığı ekler önemini yitirdiği için kullanışlı oldu. Bazı özel durumlarda yöntemler birbirlerine karşı daha iyi performans gösterdi.

Soru 1	**Türkiye, Azerbaycan, Kırgız, Kazak ve Türkmen Türkçelerinin arasındaki fark aşağıdaki kavramlardan hangisiyle ifade edilir?** A) Ağız B) Lehçe C) Şive D) Diksiyon E) Dil Ailesi Doğru Cevap: Şive Soru Etiketi: Zor Zemberek: türkiye azerbaycan kırgız kazak türkmen türkçe şive Dört Harf: trky zrby krgz kzk trkm trkç şv
Soru 2	**Aşağıdaki atasözlerinden hangisi gerçek anlamıyla kullanılmıştır?** A) Ayağını yorganına göre uzat. B) Çivi çiviyi söker.

	C) Minareyi çalan kılıfını hazırlar. D) Mum dibine ışık vermez. E) Ölüm hak, miras helâl. Doğru Cevap: Ölüm hak, miras helâl. Soru Etiketi: Orta Zemberek: atasöz gerçek anlam kullan ölüm hak miras helal Dört Harf: tszl grçk nlmy klln lm hk mrs hlâl
Soru 3	**Eski Türkçe Dönemi'nin dil, kültür ve edebiyat özelliklerini başarılı bir şekilde yansıtan ve Türk yazı dilinin başlangıcı olarak kabul edilen eser aşağıdakilerden hangisidir?** A) Şaman Yazıtları B) Budizm Yazıtları C) Türkmen Yazıtları D) Uygur Yazıtları E) Orhun(Göktürk) Yazıtları Doğru Cevap: Orhun(Göktürk) Yazıtları Soru Etiketi: Kolay Zemberek: eski türkçe dönem dil kültür edebiyat özel başarı yansı türk yazı dil başlangıç kabul eser orhun göktürk yazıt Dört Harf: sk trkç dnmn dl kltr dbyt zllk bşrl ynst trk yz dlnn bşln kbl sr rhn gktr yztl

2. Öznitelik Çıkarımı

Bu bölüm de ön işlemi tamamlanan soruların öz nitelik vektörlerinin oluşturulması için kullanılan TF*IDF, FastText ve BERT algoritmaları ile bu algortimalar ile beraber kullanılan F-Measure ve N-gram teknikleri anlatılmıştır.

TF*IDF (Term frequency - inverse document frequency) algoritması

TF-IDF ağırlığı, metin sınıflandırmada yaygın olarak kullanılan bir yöntemdir. Üzerinde çalıştığı ana birimler belge ve terimlerdir. Belgeler, içerdiği terimlerin benzerlikleri üzerinde karşılaştırılacaktır. TF 'terim sıklığı' anlamına gelir ve bir belgede belirli bir teriminin kaç kez kullanıldığını sayar. Bu nedenle belge içerisindeki bir terimin ne kadar önemli olduğunu gösterdiği söylenebilir. IDF 'ters belge sıklığı ' anlamına gelir ve belirli bir terimin kaç dokümanda kullanıldığını sayar. Sezgisel olarak, IDF, bir terimin tüm corpus üzerinde ne kadar önemli olduğunun bir ölçüsüdür. Başka bir deyişle, tüm corpus 'ta belirli bir terim ne kadar çok görünürse, o kadar küçük IDF alır yani terim değersizleşir. TF yerel ağırlık, IDF global ağırlık olarak görülebilir. Yerel ve global ağırlığı yeni bir ağırlıklı değere dönüştüren farklı formüller vardır. Bu ağırlıklı değerler, daha sonra belirli bir belge için bir vektör oluşturacaktır. Temel TF-IDF ağırlığı aşağıdaki gibi hesaplanır.

$$w_{i,j} = tf_{i,j} \times \log(\frac{|d|}{n_i})$$

$$t_i teriminin\ d_j\ dökümanında\ kaç\ kere\ geçtiği\ tf_{i,j}\ olsun$$

$$t_i teriminin\ kaç\ dökümanda\ geçtiği\ n_i\ olsun$$

|d| terimi dokümanların sayısını gösterir. Bir başka hesaplama algoritması da aşağıdaki gibidir.

$$w_{i,j} = \log(tf_{i,j} + 1) \times \log(\frac{|d|}{n_i})$$

Bazı önemli kelimler metinde sık geçtiği için yüksek ağırlıklar alıp değersizleşebilir. Bunun önüne geçebilmek için normalizasyon önerilmektedir.

$$max(tf_{dj})\ d_j\ dökümanında\ ki\ t_i\ teriminin\ en\ yüksek\ frekansı\ olsun.$$

Normalizasyon için;

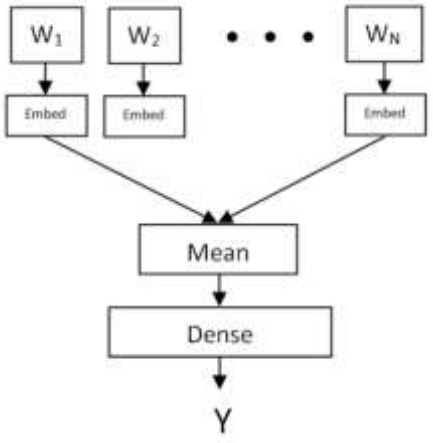
$w_{i,q} = (0.5 + 0.5 \times \frac{tf_{i,j}}{max(tf_{dj})}) \times \log(\frac{|d|}{n_i})$ olur.

FastText

FastText, cümle sınıflandırmasının verimli bir şekilde öğrenilmesi için Facebook Araştırma Ekibi tarafından oluşturulan bir kütüphanedir.(Joulin et al., 2016) FastText Modeli, sözcüklerin vektör temsillerini elde etmek için denetimsiz veya denetimli öğrenme algoritması oluşturmasına izin verir. FastText, kelime gömme işlemi için sinir ağlarını kullanır. FastText algoritmasında ön işlem sayısı ve karmaşıklığı çok azdır. Bag of Word gibi uzun boyutsal vektörlere ve ön işlemlere ihtiyaç duymaz. Derin öğrenme metotları BoW'da yer alan ön işlemleri azaltabilir fakat öğrenme süreci için büyük veri setine ihtiyacı bulunmaktadır. Bu kapsamda FastText kısa öğrenme süresi ile daha az veriyle daha doğru sonuçlar vermektedir. (Y. Su et al., 2018)

Şekil 2'de gösterildiği gibi, eğitilebilir bir gömme katmanı, bir gizli ortalama katman ve bir softmax yoğun katmandan oluşan çok katmanlı bir algılayıcıdır. N-gram ile oluşturulan kelime vektörleri w_1 ,w_2 ,..,w_n ile temsil edilir. Giriş katmanına iletilen vektörler gizli katmandan geçerek ortalaması hesaplanır, sonra etiket tahmini üretmek için lojistik regresyon ile eğitim gerçekleştirilir. Sorularımızda yer alan her bir kelime ve kelime parçaları ilgili nöronları aktif hale getirerek kelimelerin gömülü olduğu gizli katmana iletilir. Sorularda olmayan ve kelime gömme katmanında yer alan kelimeler sıfır(0) ile çarpılarak gizli katman üzerinde alakasız kelime vektörleri hesaplanması engellenir. Gizli katmanın boyutu kelime gömme işleminin boyutlarıyla belirlenir. Gizli katmanda hesaplanan ortalamalar yoğun katmana girdi olarak verilir. Softmax yoğun katmanı temel olarak doğrusal bir sınıflandırıcıdır (lojistik regresyon / çok terimli lojistik regresyon). Eğitim sürecinde üretilen hatalar, geri yayılım yoluyla doğrusal sınıflandırıcı nöronlar üzerindeki ağırlık güncellemelerini hesaplamak

için kullanılır. Doğrusal sınıflandırıcıdaki ağırlık güncellemeye ek olarak, geri yayılımdan iletilen hatalara göre kelime gömmedeki ağırlıklar güncellenir. Eğitim süreci, sınıflandırma için daha optimal bir kelime gömme yapısı ve doğrusal sınıflandırıcı üretir. (Herwanto et al., 2019)

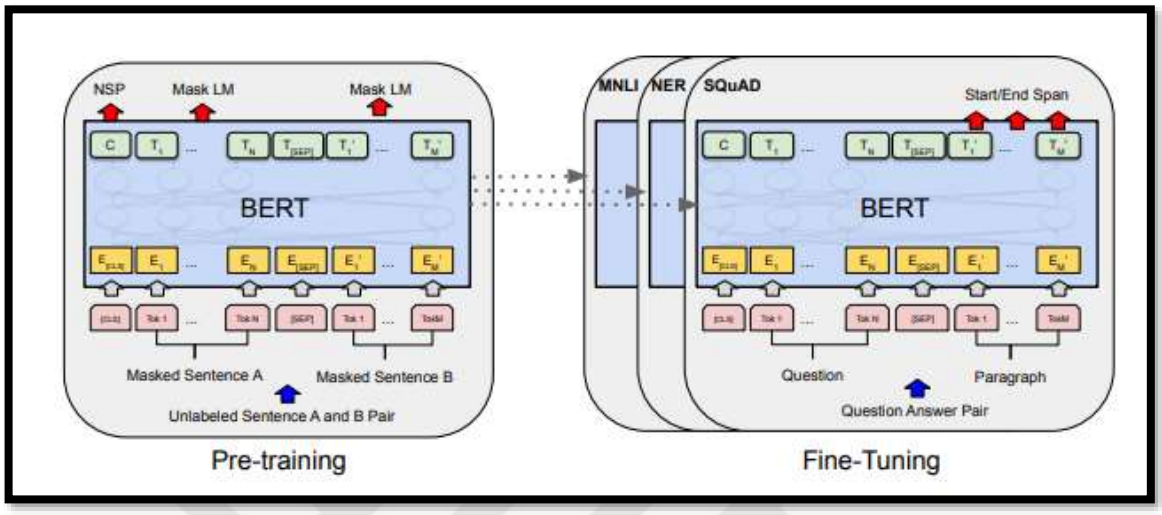


Şekil 4. FastText modeli

BERT (Bidirectional Encoder Representations from Transformers)

2018 yılında, Google tarafından Bidirectional Encoder Representations from Transformers (BERT) modeli duyuruldu. Bir cümleyi hem soldan sağa hem de sağdan sola olarak değerlendirerek cümlenin anlamının ve kelimelerin birbiriyle olan ilişkisinin daha iyi çıkarılması planlanmaktadır. Uzun süreli kelimeler arasında bağımlılıkları belirleme yeteneği ve başarısı yüksek olan çift yönlü kodlayıcı modelidir. 24 Transformer bloğu ve 1024 gizli katmanı ve 349M parametre hesaplayan büyük bir modeldir. 800 milyon İngilizce kelime (BooksCorpus) ve 2.5 Milyar İngilizce kelime (Wikipedia) dahil, toplam 3,3 milyar kelimelik korpus üzerinde eğitilmiştir. Bu modelin eğitimi için 16 TPU'ya ihtiyaç duyulmuştur. BERT'yi önceden eğitmek için geleneksel soldan sağa veya sağdan sola dil modelleri kullanılmamıştır. Bunun yerine, denetimsiz iki görev kullanarak BERT önceden eğitilmiştir. BERT, çift-yönlü olması dışında *Masked Language Modeling (MLM)* ve *Next Sentence Prediction (NSP)* adı verilen iki teknikle eğitilmiştir. Bu teknikler, Şekil 5'in sol kısmında sunulmuştur. C sonraki cümle tahmini (NSP) için kullanılır, cümledeki kelimeler $T_{1,.},T_N$ ile gösterilir ve %15'inde MLM tekniği kullanılmıştır. Bu tekniğin kullanıldığı kelimelerin %80'i [MASK] token'ı ile, %10'u rastgele başka bir kelimeyle değiştirilmiş, geri kalan %10 da değiştirilmeden bırakılmış ve %15 ise rastgele maskelenmiştir. CLS her bir girdinin başında kullanılır ve girdilerde tokenId ile değiştirilen kelimler $E_{1,..,}$ E_N ile ifade edilir. SEP ifadesi ise özel ayırıcı token olarak kullanılmaktadır. İlk teknikte, cümle içerisindeki kelimeler arasındaki ilişki tahmin edilirken, ikinci teknik olan NSP (Next Sentence Prediction)'de ise cümleler arasındaki ilişki tespit edilmiştir. Eğitim esnasında ikili olarak gelen cümle çiftinde, ikinci cümlenin ilk cümlenin

devamı olup olmadığı tahmin edilmiştir. Bu teknikten önce ikinci cümlelerin %50'si rastgele değiştirilmiş, %50'si ise aynı şekilde bırakılmıştır. Eğitim esnasındaki optimizasyon, bu iki tekniğin kullanılırken ortaya çıkan kaybın minimuma indirilmesidir. Şekil 5'te sonra ki cümlenin tahmin edilmesi gösterilmiştir. (Devlin et al., 2019) BERT algoritması genel olarak otomatik soru cevaplama sistemlerinde kullanılmaktadır.



Şekil 5. BERT katmanları

N-Gram

N-gram, belirli bir metin veya konuşma örneğinden n öğeden oluşan bir dizi olarak tanımlanabilir. Öğeler uygulamaya göre, heceler, harfler veya kelimeler olabilir. N-gramlar tipik olarak bir metin veya konuşma külliyatından toplanır. Latince sayısal önekler kullanıldığında, 1 nolu bir n-gram "unigram" olarak adlandırılır; 2 nolu n-gram "bigram", 3 nolu n-gram ise "trigram" olarak adlandırılır. Hesaplamalı biyolojide, bilinen bir boyuttaki bir polimer veya oligomere n-gram yerine k-mer adı verilir ve "monomer", "dimer", "trimer", "tetramer" gibi Yunan sayısal önekleri kullanan özel isimlerle adlandırılır. Metin sınıflandırmasında metni anlamlandırmak için sıkça kullanılan bir yöntemdir (*N-Gram,* n.d.).

Örnek metin üzerinde göstermek gerekirse; "omuz düş iç bük tür beden dil anlam içer kendi emin" metni trigram için [omuz düş iç], [düş iç bük], [iç bük tür], [bük tür beden], [tür beden dil], [beden dil anlam], [dil anlam içer], [anlam içer kendi], [içer kendi emin] şeklinde tanımlanır.

F-Measure

FastText ile sınıflandırma işlemi yaparken doğruluk değerini hesaplamak için F-Measure tekniğini kullandık. F-Measure, Kesinlik (P) ve Duyarlılık(R) değerlerinin harmonik ortalaması ile bulunmaktadır.

$$F = \frac{2RP}{R+P}$$

$$P = \frac{tp}{tp+fp}$$

$$R = \frac{tp}{tp+fn}$$

Burada tp true positives, fn false negatives ve fp false positives sayısını vermektedir.

3. Sınıflandırma

Terim sıklığından faydalanılarak vektörize edilen soruların Euclidean, Manthattan ve Levenshtein algoritmaları ile uzaklıkları uzaklıkları tespit edildi ve K-NN sınıflandırıcısı ile sınıflandırıldı. Ayrıca terim sıklığı ile vektörize edilen sorular Naive Bayes sınıflandırıcısı ile sınıflandırıldı. FastText ve BERT ile model oluşturuldu ve yine FastText ve BERT ile test edilerek sorunun zorluk derecesi tespit edildi.

Euclidean algoritması

Öklid mesafesi geometrik problemler için standart bir metriktir. İki nokta arasındaki mesafeyi ifade etmektedir ve iki veya üç boyutlu uzayda bir cetvelle kolayca ölçülebilir. Öklid algoritması, metin sınıflandırma da dahil olmak üzere kümeleme problemlerinde yaygın olarak kullanılır. Ayrıca k-means algoritmasının varsayılan benzerlik ölçüsüdür. Terim bazlı benzerlik sonucu verir. Verilen iki metin $\boldsymbol{d_a}$ ve $\boldsymbol{d_b}$ olsun. Bu metinleri temsil eden terim vektörleri sırasıyla $\underset{\boldsymbol{t_a}}{\rightarrow}$ ve $\underset{\boldsymbol{t_b}}{\rightarrow}$ olduğunu düşünüldüğünde, iki belgenin Öklid mesafesi şu şekilde tanımlanır.(Huang, 2008)

$$D_E(\underset{t_a}{\rightarrow},\underset{t_b}{\rightarrow}) = (\sum_{t=1}^{m} |w_{t,a} - w_{t,b}|^2)^{1/2}$$

Manhattan algoritması

Grid benzeri bir yol izleyerek, iki veri noktası arasındaki mesafeyi hesaplar. İki öğe arasındaki blok mesafesi, karşılık gelen bileşenlerinin farklılıklarının toplamıdır.(H.Gomaa & A. Fahmy, 2013)

Levenshtein algoritması

Levenshtein Algoritması ile metinler karşılaştırılabilir. 0 ve 1 arasında değerler vermektedir. Benzerliklerin doğru olabilmesi için belirli benzerlik sınırı belirlenmeli ve sadece

bunun üzerindeki değerler doğru kabul edilmelidir. Eşik sebebiyle katı bir sınıflandırma olarak adlandırılabilir, çünkü eşiğe yakın değerler yanlış sınıflandırılabilir.

Metin benzerliğinin yüksek olması vektör değerlerininde yüksek olacağı anlamına gelir.(Hollenstein, 2005)

$$
\begin{aligned}
&LevenshteinDistance(Source, Target)\\
&Int\ Distance = new\ Int[Source.Length + 1, Target.Length + 1]\\
&If(Source.Length == 0)\\
&\quad return\ Target.Length\\
&End\ If\\
&If(Target.Length == 0)\\
&\quad return\ Source.Length\\
&End\ If\\
&For\ Int\ I = 0\ to\ Source.Length\\
&\quad Distance[I, 0] = I\\
&End\ For\\
&For\ Int\ J = 0\ to\ Target.Length\\
&\quad Distance[0, J] = J\\
&End\ For\\
&For\ Int\ I = 1\ to\ Source.Length\\
&\quad For\ Int\ J = 1\ to\ Target.Length\\
&\qquad Int\ Cost = (Target[J - 1] == Source[I - 1]?\ 0 : 1\\
&\qquad Distance[I, J] = Min(Min(Distance[I - 1, J] + 1, Distance[I, J - 1] + 1),\\
&\qquad Distance[I - 1, J - 1] + cost)\\
&\quad End\ For\\
&End\ For\\
&return\ Distance[Source.Length, Target.Length]
\end{aligned}
$$

Denklem 1 Levenshtein Algoritması Sözde Kodu (Bhagwan & Science, 2016)

ARAŞTIRMA BULGULARI ve TARTIŞMA

Çalışmamızda iki farklı veri seti modeli kullanıldı. Her bir veri setimizde yer alan soru ve doğru cevap metinleri iki farklı yöntemle yeniden oluşturuldu. Zemberek ile her bir kelimenin kökleri bulundu ve sadece köklerden oluşan soru ve doğru cevap metni elde edildi. İkinci yöntem olarak ise sesli harfler atıldıktan sonra kalan ilk 4 sessiz harf kök kabul edildi ve sadece köklerden oluşan soru ve doğru cevap metni elde edildi. Oluşan yeni soru metinleri üzerinde soru zorluğu tespiti için sorular sıklık dağılımı ile vektörize edildikten sonra Levenshtein, Manhattan ve Euclidean algoritmaları ile uzaklık tespiti yapıldı, ardından KNN ile sınıflandırıldı. Bunun yansıra TF-IDF algoritması ile ağırlıklandırılan sorular yine Manhattan ve Euclidean algoritmalarıyla uzaklık tespiti yapıldıktan sonra KNN ile sınıflandırıldı. Ayrıca terim sıklığı ile vektörize edilen sorular Naive Bayes ile sınıflandırıldı. Son dönemlerin popüler algoritmalarından FastText ve BERT algoritması ile de soruların sınıfları belirlendi.

Yapılan çalışmalar neticesinden Zemberek ile kökleri bulunan sorular üzerinde KNN sınıflandırıcısı ile yapılan sınıflandırmalarda k=3 için tüm sorular üzerinde yapılan sınıflandırma da Müşteri İlişkileri Yönetimi dersinde TFIDF-Euclidean algoritması ile %100 başarı elde edildi. Bunu %94 ile Levenshtein, Manhattan ve Euclidean algoritmaları takip etti. Naive Bayes ile yapılan sınıflandırmada en yüksek başarı %77 ile yakalandı. FastText algoritmasında ise Veteriner Biyokimya dersinde unigram için %92 ve trigram için %95 başarı oranı yakalandı. Sesli harflerin atılmasıyla oluşan sorular üzerinde ise KNN sınıflandırıcısı ile yapılan sınıflandırmalarda k=3 için tüm sorular üzerinde yapılan sınıflandırma da Müşteri İlişkileri Yönetimi dersinde TFIDF-Euclidean ve Manhattan algoritmalarıyla %100 başarı yakalandı. Naive Bayes ile yapılan sınıflandırmada en yüksek başarı %82 ile yakalandı. FastText algoritmasında ise Veteriner Biyokimya dersinde unigram için %92 ve trigram için %95 başarı oranı yakalandı. BERT algoritmasında birkaç derste %100'lük başarı oranı yakalandı.

İkinci veri setimizde yapılan sınıflandırmalarda doğruluk oranı birinci veri setimize göre daha yüksek oldu. Bu veri tabanında ki soru sayısının daha az olması ve bazı derslerde ki etiket dağılımın eşit olmaması buna sebep olmaktadır. Bu sebeple ayırıcı sorular üzerinde yapılan sınıflandırmalar tüm sorular üzerinde yapılan ve homojen etiket dağılımına sahip olan veri tabanı kadar sağlıklı olmamaktadır.

Zemberek ile bulunan kökler ile sesli harfleri atılarak bulunan köklere uygulanan algoritmalar birbirine yakın sonuçlar verdi. Bunun neticesinde 'Türkçe kelimelerde sesli harflerin öneminin az olması' görüşü desteklendi.

Yapılan çalışmalar neticesinde zorluk derecesi tespitinin daha kolay yapılabilmesinin soru kalitesiyle doğru orantılı olduğu anlaşıldı. Bu sebeple başarının artırılabilmesi için bilgisayar bilimleri aracılığıyla ilgili üniteler üzerinden herhangi bir alan uzmanı olmaksızın yapay zeka kullanılarak sınav sorusu hazırlanabilir veya yeni algoritmalar üzerinde çalışılabilir.

Soru Benzerliklerinin Bulunması ve Sınıflandırma (Text Smilarity and Classification)

Tüm sorular ön işlemden geçirildikten sonra zorluk dereceleri tespit edilebilir hale getirildi. Soru ön işlemde uyguladığımız iki farklı kök bulma yöntemi sebebiyle tüm sorular iki farklı şekilde kaydedildi. Her bir ders için var olan soru sayısının %70'lik kısmı eğitim %30'luk kısmı test verisi olarak hazırlandı. Test kümesinde ki her bir sorunun eğitim kümesinde ki sorularla uzaklığı Levenshtein, Euclidean ve Manhattan algoritmaları ile belirlendi. Sonrasında 3-NN ile sınıflandırıldı. Sınıflandırma için ayrıca Naive Bayes yöntemi uygulandı. Uygulama sonucunda çıkan sonuçların ortalaması Tablo 10'da gösterilmiştir.

FastText algoritması üzerinden doğruluk tespiti için F-Measure tekniği uygulandı. Model eğitimi için fold=25, ngram=1 ve ngram=3, loss='hs', learning rate=0.1, context window=5 parametleri kullanıldı. Oluşturulan model test edildiğinde Duyarlılık ve Kesinlik değerleri eşit çıktı. Bunun sebebi olarak derslerde yer alan soru sayıların yeterli olmaması gösterilebilir. Tablo 7'de FastText ile alınan bazı sonuçlar gösterilmiştir.

Tablo 7. FastText ile alınan en iyi bazı sonuçlar

	Doğruluk-Kesinlik-Duyarlılık (25 Epoch)	
Ders Adı	**Unigram**	**Trigram**
Veteriner Biyokimya	**0.92**	**0.95**
İstatistiğe Giriş	0.89	0.77
İlk Dönem İslam Tarihi	0.87	0.89
Kamu Yönetimi	0.87	0.90
Temel Fotoğrafçılık	0.84	0.78
Laboratuvar Teknikleri ve Prensipleri	0.81	0.81
Yoksulluk ve Sosyal Hizmet	0.78	0.70
İslam Hukukuna Giriş	0.78	0.76
İslam Sanatları Tarihi	0.78	0.87

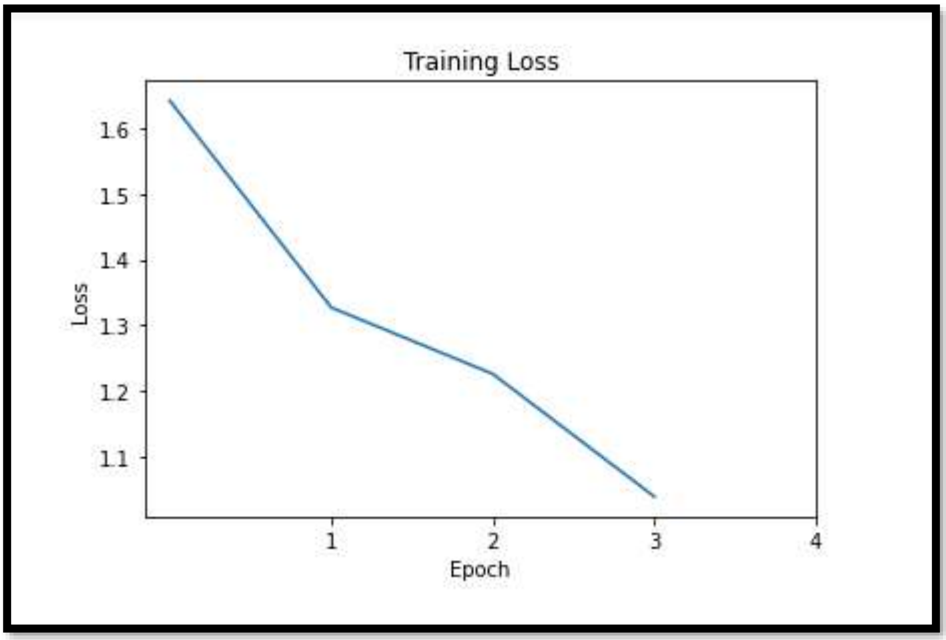
BERT algoritması ile sınıflandırma yapmak için daha önce Türkçe dili için TPU üzerinde oluşturulan ve 35GB boyutuna sahip "bert-base-turkish-128k-uncased" modelini ve

batch_size=32, epoch=4, max_length=90, pad_to_max_length=true, return_attention_mask=true, add_special_tokens=true, return_tensor='pt' parametrelerini kullandık. Tüm derslere ait soruları ders ayrımı yapmadan sınıflandırdığımızda Tablo 8'de yer alan sonuçlara ulaştık.

Tablo 8. Birinci Veri Seti Üzerinde BERT Algoritması İle Bulunan Sonuçlar

Doğruluk	0.464
Kesinlik	0.474
Duyarlılık	0.465
F-Score	0.461

BERT algoritması ile girilen metinde ki her bir kelime tokenIdsi ile yer değiştirilerek ifade edilmektedir. Belirtilen uzunluktan kısa cümleler sıfır ile istenilen uzunluğa tamamlanmaktadır. Örnek olarak orijinal soru metni "ısın çalış katılım kazan özel lider" olan bir soru "tensor([2,9928,52565,4069,2699,24196,3855,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0])" şeklinde işleme alınmaktadır. Şekil 6'da gösterildiği gibi Epoc sayısı arttıkça training loss düşmektedir. Dört epoch sayısı için training loss sıfır olarak kabul edilebilir.



ŞEKİL 6. Epoch sayısı ile Loss ilişkisi

BERT algoritması ile dersleri tek tek sınıflandırdığımızda bazı derslerde %100 başarı yakalandı. En iyi bazı sonuçlar Tablo 9'da gösterildi.

Tablo 9. Birinci Veri Seti Üzerinde Ders Bazlı BERT Algoritması İle Bulunan Sonuçlar

Ders Adı	Doğruluk	Kesinlik	Duyarlılık
Maliyet Muhasebesi II	**1.00**	**1.00**	**1.00**
Meslek Hastalıkları	**1.00**	**1.00**	**1.00**
Radyo ve Televizyonda Temel Kavramlar	**1.00**	**1.00**	**1.00**
Siyasi Tarih	**1.00**	**1.00**	**1.00**
Zootekni ve Genetik	**1.00**	**1.00**	**1.00**
Veri Tabanı Yönetim Sistemleri	0.83	0.88	0.83
Şehircilik	0.83	0.89	0.83
Veteriner Biyokimya	0.75	0.83	0.75
Türk Sosyologları	0.67	0.50	0.67

Tüm algoritmalara ait sonuçlar Ekler bölümünde Tablo 13, Tablo 14 ve Tablo 15 ‘de yer almaktadır. Çıkan sonuçlar incelendiğinde zemberek ile bulunan kökler sonucu oluşan metinler ile ilk dört sessiz harfin alınmasıyla oluşturulan metinlerin sınıflandırılması benzer sonuçlar göstermektedir. Ağırlıklandırma algoritmalarımız arasında da genel olarak benzer sonuçlar çıkmaktadır. Veri sayısının ders bazlı düşünüldüğünde az sayıda olması bunu tetiklemektedir. Ders bazlı maksimum sonuçlar Tablo 11’de gösterilmektedir. Ayırt ediciliği yüksek sorularda genel olarak daha iyi bir performans çıkmaktadır. Ders bazlı düşünüldüğünde ayırt edici soru sayısı çok daha az olduğu için bu performans göz ardı edilmektedir.

Tablo 10. Alınan Tüm Sonuçların Algoritmalara Göre Doğruluk Ortalamaları

Ortalama Sonuçlar	Zemberek- Doğruluk		Dört Harf- Doğruluk	
	1.Veri seti	2.Veri seti	1.Veri seti	2.Veri seti
TF-Levenshtein- KNN (k=3)	0.453910	**0.638578**	0.458778	**0.640631**
TF-Euclidean- KNN (k=3)	0.437916	0.633908	0.441093	0.632578
TF-Manhattan- KNN (k=3)	0.445961	0.627208	0.448263	0.626526
TFIDF-Euclidean- KNN (k=3)	0.447974	0.615408	0.447612	0.616666
TFIDF-Manhattan- KNN (k=3)	0.453890	0.625612	0.451032	0.616031
TF-Naive Bayes	0.384038	0.358172	0.369389	0.344368
FastText-3	0.455512	0.449137	**0.471800**	0.451421
FastText-1	0.445000	0.446751	0.442443	0.449105
BERT	**0.464000**	-	-	-

Tablo 11. Algoritmalar İle Alınan Maksimum Doğruluk Değerleri

Maksimum Sonuçlar	Zemberek- Doğruluk		Dört Harf- Doğruluk	
	1.Veri seti	2.Veri seti	1.Veri seti	2.Veri seti
TF-Levenshtein- KNN(k=3)	0.94	**1.00**	0.86	**1.00**
TF-Euclidean- KNN(k=3)	0.94	**1.00**	0.94	**1.00**
TF-Manhattan- KNN(k=3)	0.94	**1.00**	**1.00**	**1.00**
TFIDF-Euclidean- KNN(k=3)	**1.00**	**1.00**	**1.00**	**1.00**
TFIDF-Manhattan- KNN(k=3)	**1.00**	**1.00**	**1.00**	**1.00**
TF-Naive Bayes	0.77	0.86	0.82	0.80
FastText-3	**1.00**	0.95	**1.00**	0.95
FastText-1	**1.00**	0.95	**1.00**	0.95
BERT	**1.00**	-	-	-

SONUÇ

Bu çalışmada sorular üzerinde metin sınıflandırmada kullanılan K-NN, Naive Bayes, FastText ve BERT algoritmaları kullanılmış ve literatürdeki temel çalışmalarda elde edilen başarı oranı Türkçe için sağlanmıştır. Sonuçlar, soruların zorluk seviyelerinin metin benzerlik ve ağırlıklandırma yöntemleri yardımı ile bulunabileceğini ancak derse bağlı olarak soru hazırlama kalitesini artırarak zorluk seviyesini belirlemenin daha kolay olabileceğini göstermektedir. Deneysel sonuçlar, soruların zorluk düzeylerinin makine öğrenimi yöntemleriyle sınıflandırılabileceği tezimizi doğruladı. Derslerle ilgili yetersiz soru sayısı sınıflandırma başarısını düşürmüştür. Tüm sorular üzerinden yapılan sınıflandırma işleminde BERT ile en yüksek başarı oranı yakalanmıştır. Bazı derslerde ulaşılan yüksek başarı oranı sayesinde sınav soruları hazırlanırken bu derslere ait soruların zorluk seviyeleri önceden belirlenebilecek ve zor etiketli sorular alan uzmanlarına gönderilecektir. Alan uzmanları tarafından soru metni ve ünite içeriği karşılaştırılarak sorulacak sorunun zorluğu ve bilgi ölçmede ki yeterliliği tespit edilecektir. Alan uzmanı raporu sonrası ilgili soru sınavda kullanılabilecektir. Her sınav öncesinde tüm sorulara bu işlemlerin uygulanması çok fazla zaman alacağından sınav takviminin sıkışıklığı göz önüne alındığında bu işlemlerin uygulanması son derece zor olacaktır. Fakat sistem tarafından belirlenen sorulara bu işlemin uygulanması bu zamanın daha tasarruflu kullanılmasını sağlayacağı için uygulanabilirliği daha yüksektir. Yapılacak bu işlem sayesinde soruların zorluk derecesi dağılımı daha homojen olacaktır. Bu sayede bir soruda olması gereken ana bilgiyi ölçme, ünite içerisinde yer alma, anlaşılabilir olma ve test sorusu formatına uygun olma gibi özellikler sağlanarak sınavlarda öğrenci başarısı artırılacaktır. Sonuç olarak eğitim kalitesi yükselecektir.

KAYNAKLAR

Akın, A. A., & Akın, M. D. (2007). Zemberek, An Open Source Nlp Framework for Turkic Languages. *Structure*, *10*, 1–5.

Al-Talib, G., & Hassan, H. (2013). A Study on Analysis of SMS Classification Using TF-IDF Weighting. *International Journal of Computer Networks and Communications Security*, *1*(5), 189–194. http://www.ijcncs.org/published/volume1/issue5/p3_1-5.pdf

Alessa, A. (2018). *Text Classification of Flu-related Tweets Using FastText with Sentiment and Keyword Features*. 366–367. https://doi.org/10.1109/ICHI.2018.00058

Bhagwan, J., & Science, C. (2016). Design and Analysis of aLevenshtein Distance Based Code Clones Detection Algorithm. 259–263. https://doi.org/10.090592/IJCSC.2016.045

Çataltepe, Z., Turan, Y., & Kesgin, F. (2007). *Türkçe'de Daha Kısa Kök Kullanarak Döküman Sınıflandırma*. 7–10.

ÇELİK, Ö., & KOÇ, B. C. (2021). TF-IDF, Word2vec ve Fasttext Vektör Model Yöntemleri ile Türkçe Haber Metinlerinin Sınıflandırılması. *Deu Muhendislik Fakultesi Fen ve Muhendislik*, *23*(67), 121–127. https://doi.org/10.21205/deufmd.2021236710

D'Sa, J. L., & Visbal-Dionaldo, M. L. (2017). Analysis of Multiple Choice Questions: Item Difficulty, Discrimination Index and Distractor Efficiency. *International Journal of Nursing Education*, *9*(3), 109. https://doi.org/10.5958/0974-9357.2017.00079.4

De Boom, C., Van Canneyt, S., Bohez, S., Demeester, T., & Dhoedt, B. (2016). Learning Semantic Similarity for Very Short Texts. *Proceedings - 15th IEEE International Conference on Data Mining Workshop, ICDMW 2015*, 1229–1234. https://doi.org/10.1109/ICDMW.2015.86

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, *1*(Mlm), 4171–4186.

Faroughi, A., Morichetta, A., Vassio, L., Figueiredo, F., Mellia, M., & Javidan, R. (2021). Towards website domain name classification using graph based semi-supervised learning. *Computer Networks*, *188*(January), 107865. https://doi.org/10.1016/j.comnet.2021.107865

Fei, T., Heng, W. J., Toh, K. C., & Qi, T. (2003). Question classification for e-learning by artificial neural network. *ICICS-PCM 2003 - Proceedings of the 2003 Joint Conference of the 4th International Conference on Information, Communications and Signal Processing and 4th Pacific-Rim Conference on Multimedia*, *3*, 1757–1761. https://doi.org/10.1109/ICICS.2003.1292768

Garrido-Merchán, E. C., & González-Carvajal, S. (2020). Comparing BERT against traditional machine learning text classification. *ArXiv*, *Ml*.

H.Gomaa, W., & A. Fahmy, A. (2013). A Survey of Text Similarity Approaches. *International Journal of Computer Applications*, *68*(13), 13–18. https://doi.org/10.5120/11638-7118

Herwanto, G. B., Maulida Ningtyas, A., Nugraha, K. E., & Nyoman Prayana Trisna, I. (2019).

Hate Speech and Abusive Language Classification using fastText. *2019 2nd International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2019*, 69–72. https://doi.org/10.1109/ISRITI48646.2019.9034560

Hollenstein, S. (2005). XQuery Similarity Joins. *Citeseer*, *58*(5), I–I. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.83.1096&rep=rep1&type=pdf

Hsu, F. Y., Lee, H. M., Chang, T. H., & Sung, Y. T. (2018). Automated estimation of item difficulty for multiple-choice tests: An application of word embedding techniques. *Information Processing and Management*, *54*(6), 969–984. https://doi.org/10.1016/j.ipm.2018.06.007

Huang, A. (2008). Similarity measures for text document clustering. *New Zealand Computer Science Research Student Conference, NZCSRSC 2008 - Proceedings, April*, 49–56.

Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., & Mikolov, T. (2016). *FastText.zip: Compressing text classification models*. 1–13. http://arxiv.org/abs/1612.03651

Khan, J. Y., Khondaker, M. T. I., Afroz, S., Uddin, G., & Iqbal, A. (2021). A benchmark study of machine learning models for online fake news detection. *Machine Learning with Applications*, *4*(October 2020), 100032. https://doi.org/10.1016/j.mlwa.2021.100032

Khreisat, L. (2006). Arabic text classification using N-gram frequency statistics a comparative study. *Conference on Data Mining| DMIN'06*, 78–82. https://ai2-s2-pdfs.s3.amazonaws.com/4cbd/81db83c6f70a741fd1082ae3413133f8bd95.pdf%0Ahttp://ww1.ucmss.com/books/LFS/CSREA2006/DMI5552.pdf

Kili, R. K. E., Akg, E., & Dem, F. (n.d.). *Taspinar_E3352E0a17Bfa3Ea7Eb77B7F4D2Ada56.Pdf*.

Lilleberg, J., Zhu, Y., & Zhang, Y. (2015). Support vector machines and Word2vec for text classification with semantic features. *Proceedings of 2015 IEEE 14th International Conference on Cognitive Informatics and Cognitive Computing, ICCI*CC 2015*, 136–140. https://doi.org/10.1109/ICCI-CC.2015.7259377

Liu, J., Wang, Q., Lin, C. Y., & Hon, H. W. (2013). Question difficulty estimation in community question answering services. *EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, *October*, 85–90.

Ma, W., Yu, H., & Ma, J. (2019). *Study of Tibetan Text Classification based on fastText*. *90*(Iccia), 374–380.

N-Gram. (n.d.). https://en.wikipedia.org/wiki/N-gram

Nguyen, D., Huynh, S., Huynh, P., Dinh, C. V, & Nguyen, B. T. (2021). S2CFT: A New Approach for Paper Submission Recommendation. In T. Bureš, R. Dondi, J. Gamper, G. Guerrini, T. Jurdziński, C. Pahl, F. Sikora, & P. W. H. Wong (Eds.), *SOFSEM 2021: Theory and Practice of Computer Science* (pp. 563–573). Springer International Publishing.

Noh, S., Kim, S., & Jung, C. (2006). A Lightweight Program Similarity Detection Model using XML and Levenshtein Distance. *FECS*.

Prabowo, D. A., & Budi Herwanto, G. (2019). Duplicate question detection in question answer website using convolutional neural network. *Proceedings - 2019 5th International*

Conference on Science and Technology, ICST 2019, 1–6. https://doi.org/10.1109/ICST47872.2019.9166343

Saeed, H. H., Ashraf, M. H., Kamiran, F., Karim, A., & Calders, T. (2021). Roman Urdu toxic comment classification. *Language Resources and Evaluation*. https://doi.org/10.1007/s10579-021-09530-y

Su, Y., Lin, R., & Kuo, C.-C. J. (2018). *On Tree-structured Multi-stage Principal Component Analysis (TMPCA) for Text Classification*. *July*. http://arxiv.org/abs/1807.08228

Su, Z., Ahn, B. R., Eom, K. Y., Kang, M. K., Kim, J. P., & Kim, M. K. (2008). Plagiarism detection using the Levenshtein distance and Smith-Waterman algorithm. *3rd International Conference on Innovative Computing Information and Control, ICICIC'08*, 0–3. https://doi.org/10.1109/ICICIC.2008.422

Wajeed, M. A., & Adilakshmi, T. (2011). Different similarity measures for text classification using KNN. *2011 2nd International Conference on Computer and Communication Technology, ICCCT-2011*, 41–45. https://doi.org/10.1109/ICCCT.2011.6075188

Yahya, A. A., Osman, A., Taleb, A., & Alattab, A. A. (2013). Analyzing the Cognitive Level of Classroom Questions Using Machine Learning Techniques. *Procedia - Social and Behavioral Sciences*, *97*, 587–595. https://doi.org/10.1016/j.sbspro.2013.10.277

Yoon, S.-Y., Cho, Y., & Napolitano, D. (2016). *Spoken Text Difficulty Estimation Using Linguistic Features*. 267–276. https://doi.org/10.18653/v1/w16-0531

Zhang, Y., Jin, R., & Zhou, Z. H. (2010). Understanding bag-of-words model: A statistical framework. *International Journal of Machine Learning and Cybernetics*, *1*(1–4), 43–52. https://doi.org/10.1007/s13042-010-0001-0

Zhang, Y. T., Gong, L., & Wang, Y. C. (2005). Improved TF-IDF approach for text classification. *Journal of Zhejiang University: Science*, *6 A*(1), 49–55. https://doi.org/10.1631/jzus.2005.A0049

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı:	Adem CANPOLAT
Doğum tarihi:	
Doğum Yeri:	
Uyruğu:	
Adres:	
Tel:	
E-mail:	
Eğitim	
Lise:	Erzurum Merkez Anadolu Lisesi
Lisans:	Dokuz Eylül Üniversitesi, Mühendislik Fakültesi
Yabancı Dil Bilgisi	
İngilizce:	İyi