

**T.C.**

**FENERBAHÇE ÜNİVERSİTESİ**

**MÜHENDİSLİK VE MİMARLIK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**İŞ İLANINDAKİ POZİSYONA EN UYGUN ÖZGEÇMİŞLERDEKİ YETKİNLİKLERİ WEB TARAMA İLE TOPLAYAN SİSTEM**

**BİTİRME PROJESİ**

**Hazırlayan**

**BEDİRHAN İLERİ**

**Danışman**

**Doç. Dr. Ferdi Sönmez**

**HAZİRAN – 2023**

ÖZET

Bu çalışmanın konusu, eleman arayan şirketlerdeki İK çalışanlarının kullandığı özgeçmiş eleme yazılımları gibi dinamik ve verimli bir sistem tasarlamaktır. Çalışmanın amacı, web tarama yapılarak istenilen sonuca ulaşmak ve bunu en iyi şekilde açıklayıp Türk literatürüne saygıdeğer bir çalışma kazandırmaktır. Çalışmada öncelikli olarak GitHub’daki kullanıcıların, projelerinin konusu ve açıklaması ile projelerinde kullandığı programlama dili verisi web tarama ile toplanmış ve işlenebilir hale getirilmiştir. Bu verilerden, kullanıcının programlama dili yeteneği temel yetkinlik olarak kabul edilir. Daha sonra, Baykar web sitesindeki web yazılım uzmanı iş ilanı web tarama ile toplanmıştır. Sistem çalıştırıldığında, eklenmek istenen GitHub kullanıcı adı girilir. Sonuç olarak, iş ilanındaki aranan niteliklerle kullanıcının en çok eşleşen yetkinlikleri bulunur ve gösterilir.

**ANAHTAR KELİMELER:** Özgeçmiş, web tarama, GitHub, iş ilanı, yetkinlik

ABSTRACT

The subject of this study is to design a dynamic and efficient system, such as the CV screening software used by HR employees in companies seeking employees. The aim of the study is to reach the desired result by scanning the web and to explain it in the best way and to bring a respectable study to the Turkish literature. In the study, primarily the subject and description of the projects and the programming language data used by the users on GitHub in their projects were collected and processed by web browsing. From these data, the user's programming language ability is considered the core competency. Then, web software specialist job posting on Baykar website was collected by web browsing. When the system is started, the GitHub username to be added is entered. As a result, the most matching competencies of the user with the required qualifications in the job posting are found and displayed.

**KEYWORDS**: CV, web scraping, GitHub, job posting, competence

İÇİNDEKİLER

**[İÇİNDEKİLER vii](#_Toc177)**

**[ŞEKİL LİSTESİ viii](#_Toc9786)**

**[KISALTMALAR LİSTESİ ix](#_Toc28749)**

**[1. GİRİŞ 1](#_Toc22715)**

**[2. LİTERATÜR TARAMASI 3](#_Toc10094)**

**[3. MATERYAL VE YÖNTEM 5](#_Toc20367)**

**[4. UYGULAMA 7](#_Toc22002)**

**[5. BULGULAR VE TARTIŞMALAR 9](#_Toc21540)**

**[6. SONUÇ VE ÖNERİLER 10](#_Toc15474)**

**[KAYNAKLAR 11](#_Toc22026)**

**[ÖZGEÇMİŞ 16](#_Toc24623)**

**ŞEKİL LİSTESİ**

[Şekil 1.1. Algoritma Akış Diyagramı 2](#_Toc30539)

[Şekil 3.1. Baykar İş İlanı 5](#_Toc30086)

[Şekil 3.2. GitHub Kullanıcı Profili 6](#_Toc21010)

[Şekil 4.1. İşin Tanımı ve Genel Nitelikler 7](#_Toc13518)

[Şekil 5.1. GitHub Kullanıcısının Eklenmesi 9](#_Toc17763)

[Şekil 5.2. Hatalı Giriş ve Sistemden Çıkış Yapılması Durumu 9](#_Toc9814)

KISALTMALAR LİSTESİ

**CV :** Curriculum Vitae

**DVM :** Destek Vektör Makinesi

**FP :** Frequent Pattern

**HR :** Human Resources

**İK :** İnsan Kaynakları

**K-NN :** K-Nearest Neighbors

**LDA :** Latent Dirichlet Allocation

**LR :** Lojistik Regresyon

**NB :** Naive Bayes

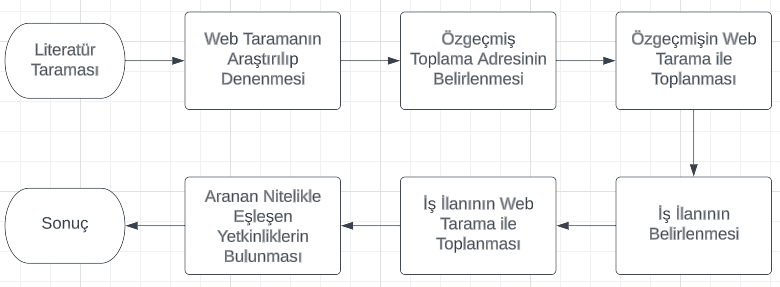
**TF-IDF :** Term Frequency - Inverse Document Frequency

1. GİRİŞ

Günümüzde teknolojinin yaygınlaşması ve sürekli olarak gelişmesiyle verilerin boyutu da eskiye kıyasla bir hayli artış göstermiştir. Büyük veri (big data), çok büyük ve karmaşık verileri ifade eder. Bu verilerle insanların başa çıkması imkansızdır. Dolayısıyla zaman ve maliyet açısından tasarruf sağlayan daha dinamik sistemlere ihtiyaç vardır. Web tarama sayesinde bir insanın iş yükünü önemli ölçüde azaltan bir sistem geliştirilebilir. Web tarama, web sitelerindeki ihtiyaç duyulan verileri toplama işlemidir. Web tarama sayesinde, insanların tek tek web sitelerine girip ihtiyacı olan verileri bulmak için zamanını heba ederek mental yorgunluk yaşamalarının önüne geçilir.

Eleman arayan şirketlerdeki İK çalışanlarının öncelikli görevi, işin tanımını yapmak ve aranan elemanın niteliklerini belirlemektir. Bu adımları tamamladıktan sonra geriye uygun elemanı bulmak kalıyor. Özellikle de dünya çapındaki büyük şirketlerin iş ilanlarına binlerce başvuru yapıldığı düşünüldüğünde, İK çalışanlarının ilana başvuran herkesin özgeçmişine tek tek bakması zaman ve maliyet açısından mümkün değildir. Bu nedenle şirketlerdeki İK çalışanlarının neredeyse tamamı, iş ilanındaki aranan eleman tanımıyla en çok eşleşen özgeçmişleri bulup geri kalanını eleyen yazılımlar kullanıyor. Bu yazılımlar, iş ilanına başvuran özgeçmişleri önemli ölçüde indirgediği için İK çalışanlarının kalan özgeçmişleri değerlendirmesine olanak sağlıyor. Çalışmadaki amaç, çok karmaşık olmayan, şirketlerin kullanabileceği seviyede, zaman ve mali açıdan kazanç sağlayacak bir İK yazılımı geliştirmektir.

Bu çalışmayı hazırlarken internet ortamından web tarama yöntemleri araştırılmış, ilgili akademik çalışmalar incelenmiştir. Bu doğrultuda tasarım açısından fikir sahibi olunmuştur. İlk olarak Python’da, bazı web adreslerindeki verileri web taramayla toplama işlemleri denenmiştir. Aynı kod parçacığıyla bazılarında başarılı olunup bazılarında başarısız olunduktan sonra bazı web sitelerinin web tarama işlemine izin vermediği anlaşılmıştır. Bu sebeple, web tarama işlemine izin verilen ve kişilerin yetkinliklerinin açık kaynak olarak yayınlandığı bir web sitesini bulmak için araştırmalar yapılmıştır. Bunun sonucunda, GitHub platformunun web taramaya müsade ettiği, kişilerin projelerini ve yetkinliklerini barındıran bir web sitesi olduğu tespit edilmiştir. GitHub’tan özgeçmiş toplama denemeleri yapılmış ve başarılı olunmuştur. İş ilanını belirlemek için araştırmalar yapılmış ve Baykar sitesindeki iş ilanlarının projeye uygun olduğu saptanmıştır. Bu sitedeki bir iş ilanı seçilip web taramayla başarıyla çekilmiştir. Daha sonra, iş ilanındaki aranan niteliklerle eşleşen özgeçmişteki yetkinlikleri bulma algoritması düşünülmüştür. Çeşitli denemeler sonucunda algoritma tasarlanmış ve işlem başarıyla gerçekleşmiştir.



Şekil 1.1. Algoritma Akış Diyagramı

1. LİTERATÜR TARAMASI

Bir üniversitenin bilişim sistemindeki mesajlaşmalardaki anahtar kelimeleri bulmak için yapılan bir çalışmada tf-idf ve ki-kare yöntemleri kullanılarak 3961 adet mesaj üzerinde metin madenciliği uygulanmış ve en çok kullanılan kelimeler listelenmiştir (Yıldız, 2016).

Bir turizm platformu olan TripAdvisor’dan, Antalya bölgesindeki otellerle ilgili 212345 adet yorumun toplanıp metin madenciliği işlemine tabi tutulduğu bir çalışmada yorumların puan ortalaması, puan ortalamasının bölgelere göre dağılımı, verilen puanla yorumun ne kadar çeliştiği, yorumların konusu vs. çeşitli bilgiler elde edilmiştir. Bu bilgileri analiz ederken LR, DVM, NB ve LDA yöntemleri kullanılmıştır. Buna benzer bir başka çalışmada, bir turizm platformu olan booking.com’dan Bakü’deki 5 yıldızlı otellere dair 3275 yorum RapidMiner yazılımıyla toplanıp bu yorumlar üzerinde LDA, K-Ortalamalar ve TF-IDF gibi çeşitli yöntemlerle analizler uygulanmıştır. Bu analizlere göre pozitif ve negatif yorumlardaki konu dağılımları tespit edilmiştir (Büyükeke ve Sökmen, 2020; Öztürk ve Tengilimoğlu, 2019).

Bir çalışmaya göre, birçok yazarın birçok konu hakkında yorum yaptığı bir platform olan Ekşi Sözlük’ten 574 çikolatalı krema yorumu WebHarvy ile toplanmıştır. Python ve MAXQDA kullanılarak yorumlar üzerinde metin madenciliği ve çeşitli analizler yapılmıştır. Bu analizlerin sonucuna göre insanların farklı çikolatalı krema markalarını tercih etmelerinin en büyük sebepleri bulunmuştur (Berk, Değirmen, ve Yılmaz, 2022).

Bir çalışmaya göre, Twitter’dan 14145 tane mobilya konulu yorum RapidMiner yazılımıyla toplanmıştır. Bu yorumlar, duygu analizi yapılarak olumlu ve olumsuz olmak üzere sınıflandırılmıştır. Karar ağacı ve FP-Growth birliktelik algoritması kullanılarak duyguları olumlu ve olumsuz etkileyen anahtar kelimeler bulunmuştur (Karayılmazlar vd., 2019).

Bir çalışmaya göre, metin madenciliği yapılarak konusu Bitcoin olan 292814 adet tweet Web Collector yazılımıyla toplanmıştır. Toplanan tweetler, QDA Miner uygulaması ile duygu analizine tabi tutulmuştur. Yorumların atıldığı tarihe eşdeğer olarak [www.coindesk.com](http://www.coindesk.com) sitesinden Bitcoin fiyatları toplanmıştır. Birim kök, Johansen ve Granger Nedensellik analiz yöntemleri uygulanarak Twitter yorumları ile Bitcoin fiyatları arasındaki ilişki açıklanmaya çalışılmıştır. %94 güven aralığındaki sonuca göre, yapılan yorumların Bitcoin fiyatlarını etkilemediği ancak Bitcoin fiyatlarının yapılan yorumları etkilediği kanısına varılmıştır (Polat ve Akbıyık, 2019).

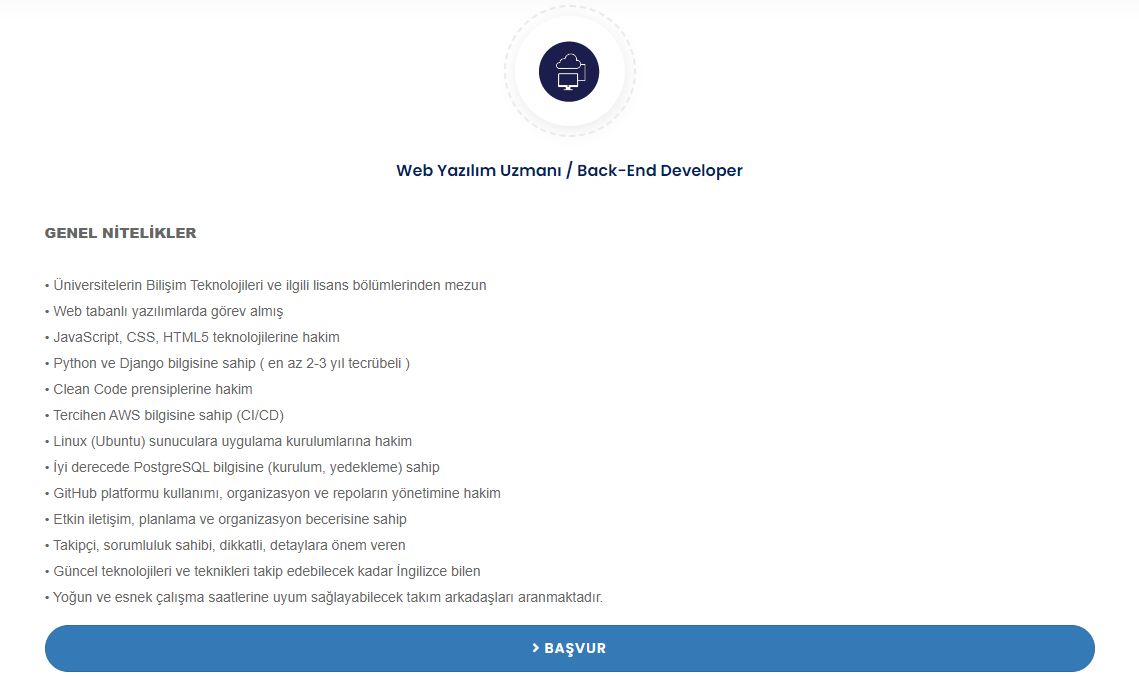
Bir çalışmaya göre, Anadolu Ajansı’ndan 2610 adet Türkçe makale toplanıp Naive Bayes ve K-NN algoritması ile bit, frekans ve TF-IDF ağırlıklandırma yöntemiyle bu makaleler sınıflandırılmıştır (Pilavcılar, 2007).

Yarısı reklam, yarısı normal olan 800 adet Türkçe epostanın kullanıldığı bir çalışmada, metin madenciliği yapılarak reklam içerikli epostaların bulunması amaçlanmıştır. DVM, K-NN ve NB algoritmaları ile binary, frekans ve TF-IDF ağırlıklandırma yöntemleri kullanılarak hangi algoritma ile yöntemin en yüksek başarıma ulaştığı gözlemlenmeye çalışılmıştır. Buna göre, K-NN algoritmasında k’nin 14 olduğu durumda binary vektör ile %96,5 başarı oranıyla en iyi sonuç gözlemlenmiştir (Çalış, Gazdağı, ve Yıldız, 2013).

Bir çalışmaya göre, Weka ve text2arff yazılımı kullanılarak Türkiye’deki 3 büyük siyasi partinin genel başkanlarının konuşmaları toplanmıştır. Karar Ağaçları, NB, DVM, K-NN algoritmaları ve 6 farklı özellik çıkarımı ile 3 farklı ağırlıklandırma yöntemi kullanılarak metin madenciliği gerçekleştirilmiş ve metinlerin sınıflandırma başarısı ölçülmüştür. Çıkan sonuca göre, NB ve 2 gramlardan oluşan özellik vektörleri en yüksek başarı oranına ulaşırken Karar Ağaçları en düşük sınıflandırma başarısında kalmıştır (Seçkin, 2011).

1. MATERYAL VE YÖNTEM

Bu çalışmada, Python’daki requests kütüphanesi kullanılarak Baykar web sitesinde bulunan Şekil 3.1’deki iş ilanı çekilmiştir. BeautifulSoup kütüphanesi ve lxml yardımıyla html formatındaki veriler ayrıştırılmıştır. Bu sayede veri üzerinde işlem yapılması kolay hale getirilmiştir.



Şekil 3.1. Baykar İş İlanı

İş ilanındaki verileri çekmek için bu verilerin html kaynağı dikkatle incelenerek işe yarayan verilerin html başlıkları ve kodları find işlemiyle bulunmuştur. Örneğin, işin tanımı bir başlık altında bulunduğu için find ile bulunur. Ancak işin genel nitelikleri, birden fazla html başlığı altında olduğu için find\_all ile bulunur ve for döngüsüyle her bir elemanına erişim sağlanır.

Bu projede özgemişleri toplama işlemi GitHub adresinden gerçekleşmiştir. GitHub, yazılım geliştirme projelerini barındıran ve yöneten web tabanlı bir platformdur. GitHub platformundaki kullanıcıların, repositories alanında ilk sayfada bulunan projelerindeki Şekil 3.2’de gösterilen kutucuk içindeki veriler toplanmıştır. Bu veriler, projenin konusu ve açıklaması ile projede kullanılan programlama dilini ifade eder. Bu projelerde kullanılan programlama dilleri kullanıcıların yetkinlikleri olarak kabul edilecek ve iş ilanındaki pozisyona uygunlukları değerlendirilecektir.

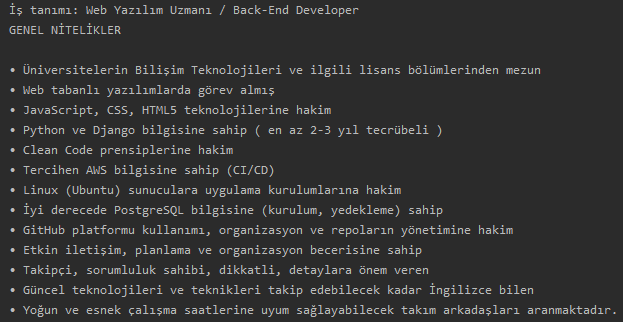


Şekil 3.2. GitHub Kullanıcı Profili

1. UYGULAMA

Bu çalışmada, arayüz ve arka plan olmak üzere 2 bölüm vardır. Arayüz, sistemi kullanan kişilerin gördüğü paneldir. Arka plan ise fonksiyonun olduğu ve işlemlerin gerçekleştiği alandır.

Arka plan dosyasında, ilk olarak Baykar sitesindeki web yazılım uzmanı iş ilanı çekilir ve ayrıştırılarak veri üzerinde işlem yapılması mümkün hale getirilir. Description değişkeninde, işin tanımını çekmek için gerekli olan html başlığı ve sınıf kodu verilerek text formatına dönüştürülür. Qualifications değişkeninde işin niteliklerinin olduğu html başlıkları çekilir. Sayaç ve karakter dizisi tanımlanarak for döngüsüyle qualifications değişkeninin içine girilir. Böylece sayaç yardımıyla web sayfasındaki yalnızca iş ilanı çekilerek karakter dizisine atılır. Karakter dizisindeki boşluklar split metoduyla bölünerek listenin içine kelime kelime atılır.



Şekil 4.1. İşin Tanımı ve Genel Nitelikler

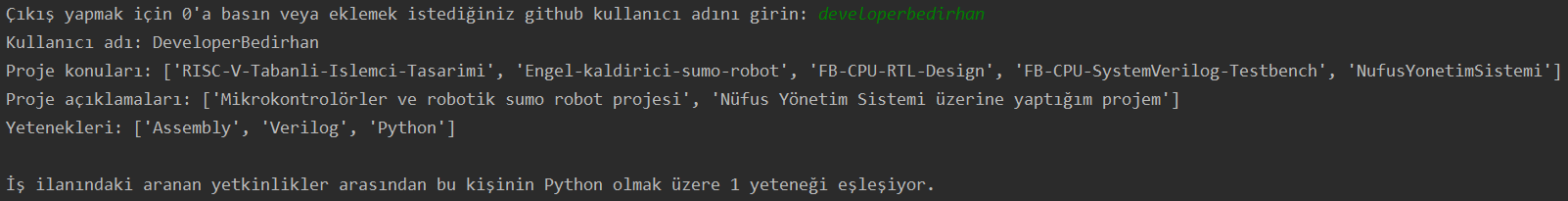
GitHub’taki kişilerin verilerini çekme işlemi scraperFunc fonksiyonunda gerçekleştirilir. Bu fonksiyonda kullanıcının adı username değişkenine atanır. Kullanıcının yaptığı projelerin konuları ve açıklamaları ile projelerinde kullandığı programlama dilleri sırasıyla projectList, explanationList ve programList listelerine eklenir. ProgramList listesinde diğerlerinden farklı olarak projelerde kullanılan programlama dillerinin aynı olması durumu kontrol edildikten sonra listeye eklenir. Son olarak, kullanıcının adı, projelerinin konusu ve açıklaması, projelerinde kullanılan programlama dilleri ekrana bastırılır.

Kişilerin bildikleri programlama dillerinin iş ilanında istenen nitelikler arasında olup olmadığı kontrolü yapılır. Bu kontrolde, iş ilanındaki nitelikleri temsil eden listenin sonundaki virgülle birlikte HTML5 niteliğindeki 5 ibaresi kaldırılarak iş ilanındaki nitelikle kişilerin programlama dillerinin eşleşme kontrolü gerçekleştirilebilir. Ayrıca kaç programın eşleştiği bilgisi counter değişkenine atanırken matching dizisine programlama dilleri eklenir. İş ilanındaki bazı ibareler GitHub’taki programlama dillerinin içine girmediği için keywords listesine eklenen bu anahtar kelimeler proje açıklamalarında geçiyorsa matching dizisine eklenir. Daha sonra iş ilanındaki aranan niteliklerle kişilerin yetkinliklerinin hangileri ve kaç tanesi eşleşiyor bilgisi ekrana bastırılır.

Arayüz panelinde yalnızca while döngüsü vardır. Bu döngüde, hatalı giriş yapıldığında programı sonlandırmayıp hatalı girişi kullanıcının bilgisine sunmak için try ve except kontrolleri kullanılmıştır. Kullanıcı kodu çalıştırdığında karşısına bilgilendirme yazısı çıkar. Kullanıcı, özgeçmişleri iş ilanıyla karşılaştıracak kişilerin GitHub’taki kullanıcı adlarını girerse scraperFunc fonksiyonuna gidilir. Girişi yapılan her bir kişinin GitHub’taki bilgileri ve iş ilanındaki aranan niteliklerle kişilerin yetkinliklerinin hangileri ve kaç tanesi eşleşiyor bilgisi ekrana bastırılır. Kullanıcı çıkış yapmak istediği takdirde 0 girişi yapar ve çıkış yapılır.

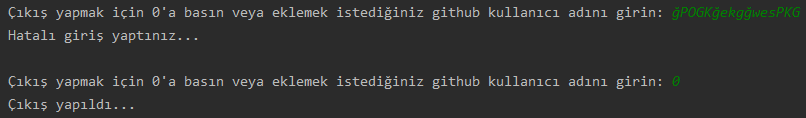
1. BULGULAR VE TARTIŞMALAR

Sistem çalıştırıldığında Şekil 5.1’deki gibi çıktı alınır. Örneğin GitHub’da DeveloperBedirhan kullanıcı adına sahip olan kişi eklenmek istendiğinde kullanıcı adı, büyük veya küçük harf duyarlılığına bakılmaksızın yazılır. Bu kullanıcının adı, GitHub’a eklediği projelerinin konusu ve açıklaması ile bu projelerde kullandığı programlama dili yeteneği çıktı ekranında gösterilir. Sonuç olarak da iş ilanındaki aranan yetkinlikler arasından bu kişinin eşleşen yeteneğinin bilgisi bastırılır.



Şekil 5.1. GitHub Kullanıcısının Eklenmesi

Sistem çalıştırıldığında hatalı bir giriş olması ve istenildiğinde çıkış yapılabilmesi durumları Şekil 5.2’de gösterilmiştir.



Şekil 5.2. Hatalı Giriş ve Sistemden Çıkış Yapılması Durumu

Bu projenin bulgularında, GitHub’taki birkaç kullanıcı üzerinde denemeler yapılmış ve sistemin başarıyla çalıştığı görülmüştür. GitHub’ta, programlama dili kabul edilmeyip iş ilanında istenen Linux ve PostgreSQL gibi durumların da kullanıcıların projelerindeki açıklamalarından ayıklanıp iş ilanındaki aranan nitelikle eşleştirilme işlemi sorunsuz gerçekleşmiştir.

1. SONUÇ VE ÖNERİLER

Bu çalışma, projenin amacına uygun bir şekilde sonuçlanmıştır. Ancak GitHub’daki kullanıcılar ve Baykar’daki iş ilanı web tarama ile toplanıp bir dosyaya kaydedilmemektedir. Bu durumda, sistemin her çalıştırılmasında geçmiş verilerin silinmesinin yanı sıra sistemin hata alıp çalışmaması riski olacaktır. Bu sebeple web tarama ile toplanan verilerin bir dosyaya kaydedilmesi önerilmektedir. Bu sayede web tarama yapılan web sitesinde arıza veya bakım olması, içeriğin değişmesi ya da sitenin hacklenmesi durumunda yeni kullanıcılar eklenemeyecek fakat geçmişte eklenen kullanıcıların verisine erişim sağlanacaktır.

Bu çalışma kapsamında oluşturulan sistem, başarıyla çalışmasına karşın basit çözümler sunmaktadır. Örneğin yalnızca bir iş ilanı değerlendirmeye alınıp kullanıcıların GitHub’daki repositories alanının sadece ilk sayfası çekilmektedir. Kullanıcıların repositories bölümüne yüklediği bütün projeler çekilebilir. Ayrıca özgeçmiş elemek için bu sistemi kullanmak isteyen bir İK personeli, kodun içeriğini değiştirmeden sisteme tanımlı olan iş ilanı dışında başka bir iş ilanı giremeyecektir. Bunun dışında özgeçmişlerini eklemek istediği kişilerin GitHub’daki kullanıcı adlarını manuel olarak tek tek girecektir. Kurumsal bir şirkete yüzlerce kişinin başvurduğu düşünülürse bu sistemi kullanmak epey bir zaman kaybına yol açacaktır. İK personelinin elindeki GitHub kullanıcı adları verisi okunup sistem toplu halde çalıştırılabilir. Bunun sonucunda basit bir İK yazılımı, gelişmiş bir İK yazılımına evrilecektir.

**KAYNAKLAR**

Berk, O. N., Değirmen, S. ve Yılmaz, H. (2022). Tüketici tercihlerini etkileyen faktörlerin metin madenciliği ve içerik analizi ile incelenmesi: Ekşi sözlük krem çikolata örneği. Journal of Tourism and Gastronomy Studies, 10(2), 1066-1087.

Büyükeke, A. ve Sökmen, A. (2020, Ekim). Sosyal medya verilerinden rekabet avantajı yaratma: Antalya örneği. International Travel and Tourism Dynamics, Hacı Bayram Veli Üniversitesi, Ankara. 16 Ocak 2023 tarihinde https://www.researchgate.net/profile/Ilkay-Yilmaz-2/publication/344433901\_COGRAFI\_ISARETLI\_TURK\_VE\_ITALYAN\_FINDIKLARININ\_BAZI\_OZELLIKLERININ\_KARSILASTIRMALI\_INCELENMESI/links/5fedce3592851c13fed90436/COGRAFI-ISARETLI-TUeRK-VE-ITALYAN-FINDIKLARININ-BAZI-OeZELLIKLERININ-KARSILASTIRMALI-INCELENMESI.pdf#page=109 adresinden erişildi.

Çalış, K., Gazdağı, O. ve Yıldız, O. (2013). Reklam içerikli epostaların metin madenciliği yöntemleri ile otomatik tespiti. Bilişim Teknolojileri Dergisi, 6(1), 1-7.

Karayılmazlar, S., Bardak, T., Avcı, Ö., Kayahan, K., Karayılmazlar, A. S., Çabuk, Y., ... ve İmren, E. (2019). Veri madenciliği algoritmalarına dayalı olarak sosyal medya üzerinden mobilya seçimindeki yönelimlerin belirlenmesi: Twitter örneği. Turkish Journal of Forestry, 20(4), 447-457.

Öztürk, Y. ve Tengilimoğlu, E. (2019, Haziran). Metin madenciliği yöntemleri ile online yorumların kümelenmesi: Bakü otelleri örneği. Uluslararası Sosyal Bilimler Kongresinde sunulan bildiri, Uluslararası Balkan Üniversitesi, Üsküp. 16 Ocak 2023 tarihinde https://www.researchgate.net/profile/Engin-Tengilimoglu/publication/337159950\_Metin\_Madenciligi\_Yontemleri\_ile\_Online\_Yorumlarin\_Kumelenmesi\_Baku\_Otelleri\_Ornegi/links/5dc911eb92851c8180436812/Metin-Madenciligi-Yoentemleri-ile-Online-Yorumlarin-Kuemelenmesi-Bakue-Otelleri-Oernegi.pdf adresinden erişildi.

Pilavcılar, İ. F. (2007). Metin madenciliği ile metin sınıflandırma. (Yüksek Lisans Tezi). Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

Polat, M. ve Akbıyık, A. (2019). Sosyal medya ve yatırım araçlarının değeri arasındaki ilişkinin incelenmesi: Bitcoin örneği. Akademik İncelemeler Dergisi, 14(1), 443-462.

Seçkin, K. (2011). Metin madenciliğinde kullanılan yöntemlerin karşılaştırılması: Siyasi parti liderlerinin grup genel toplantı konuşmaları ile bir uygulama. (Yüksek Lisans Tezi). Sakarya Üniversitesi, Sosyal Bilimler Enstitüsü, Sakarya.

Web Scraping İçin 5 Popüler Python Kütüphanesi. 30 Mayıs 2023 tarihinde https://teknoloji.org/web-scraping-icin-5-populer-python-kutuphanesi/#:~:text=lxml%20K%C3%BCt%C3%BCphanesi,-Request%20k%C3%BCt%C3%BCphanesinin%20bir&text=Y%C3%BCksek%20performansl%C4%B1%2C%20h%C4%B1zl%C4%B1%2C%20HTML%20ve,ve%20lxml%20kombinasyonu%20%C3%A7ok%20yayg%C4%B1nd%C4%B1r. adresinden erişildi.

Web Tarama Nedir?. 30 Mayıs 2023 tarihinde <https://kronnika.com/tr/web-tarama/#:~:text=Web%20Tarama%20Nedir%3F,Web%20Kaz%C4%B1ma%20olarak%20da%20adland%C4%B1r%C4%B1l%C4%B1r.> adresinden erişildi.

Yıldız, O. (2016). Metin madenciliğinde anahtar kelime seçimi bir üniversite örneği. Yönetim Bilişim Sistemleri Dergisi, 2(3), 29-50.