

**T. C.  
YOZGAT BOZOK ÜNİVERSİTESİ  
MÜHENDİSLİK MİMARLIK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**TWITTER ÜZERİNDEN ELDE EDİLEN VERİLER KULLANILARAK TWEET  
DUYGU ANALİZİNİN YAPILMASI VE GÖRSELLEŞTİRİLMESİ**

**Engin KARATAŞ  
16008118040**

**BMH481 BİTİRME PROJESİ**

**YOZGAT 2022**

**T. C.  
YOZGAT BOZOK ÜNİVERSİTESİ  
MÜHENDİSLİK MİMARLIK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**TWITTER ÜZERİNDEN ELDE EDİLEN VERİLER KULLANILARAK TWEET  
DUYGU ANALİZİNİN YAPILMASI VE GÖRSELLEŞTİRİLMESİ**

**Engin KARATAŞ  
16008118040**

**Bitirme Projesi Danışmanı:**

**Doç. Dr. MEHMET BAKIR**

**YOZGAT 2022**

**T. C.**  
**YOZGAT BOZOK ÜNİVERSİTESİ**  
**MÜHENDİSLİK MİMARLIK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**TWITTER ÜZERİNDEN ELDE EDİLEN VERİLER KULLANILARAK TWEET**  
**DUYGU ANALİZİNİN YAPILMASI VE GÖRSELLEŞTİRİLMESİ**

Engin KARATAŞ  
I. ÖĞRETİM

Danışman: Doç. Dr. MEHMET BAKIR .....

Bölüm Başkanı: Doç. Dr. MEHMET BAKIR .....

**YOZGAT 2022**

## **ÖZET**

### **TWITTER ÜZERİNDEN ELDE EDİLEN VERİLER KULLANILARAK, TWEET DUYGU ANALİZİNİN YAPILMASI**

Bir konu hakkında Twitter üzerinde insanların genel kanısı bilimsel yaklaşımlarla araştırmak istenildiğinde, ilgili konunun hashtag(#) ilgili Tweet'leri okumak ve araştırma yapan araştırmacının bireysel fikri ile değerlendirmek gerekmektedir. Her bir tweet paylaşımını teker teker okuyup değerlendirmek bilimsel bir analiz niteliğine sahip olmayacaktır. Kullanıcıların duygularını nicel bir bakış ile değerlendirilmesini zorlaştıracaktır. Elastic Search kullanarak Twitter tweet analizi projesi, kullanıcıların yaptığı yorumlar bilgisayar tarafından analiz edilip konu bazında insanların genel kanılarının objektif bir şekilde değerlendirilip sınıflandırma işlemlerini standartlar çerçevesinde değerlendirilmesini amaçlar. Sınıflandırılan bu veriler üzerinde filtrelemeler yaparak **eş zamanlı izlenebilen grafikler** oluşturulmuştur. Bu sayede bu grafikler üzerinde duygu verilerinin yorumlanması sağlanmıştır.

Bilgisayar bilimleri, insan diline en yakın yeni nesil 5. Nesil programlama dillerinin gelişimi ile gelen yeni teknolojiler ile arama, veri kaydetme, görselleştirme işlemleri, dağıtık sistemler üzerinde etkili şekilde çalıştırılabilir duruma gelmiştir. Yeni teknolojiler üzerinde çalışan projede, insan faktörünü en aza indirilmiş, yapılacak çalışmanın analiz edilmesini ve nicel sonuçlar elde edilip değerlendirilmesi kolaylaştırmıştır.

Tez özelinde Twitter üzerindeki hashtag(topic)'lerden iki tanesi "Ukrayna Rusya savaşı ve Dolar Kuru" incelenmiştir ve Ukrayna Rusya savaşı incelenerek sonuç çıkarılmıştır.

**Anahtar Kelimeler:** Twitter, Duygu Analizi, Elastic Search, Kibana, AWS, Python, Harita Görselleştirme

## ÖNSÖZ

Proje üzerinde yapılan çalışmada, Twitter üzerinden incelenecek konu her ne olursa olsun ilgili konunun konum bazında duygu analizi yapılabilmektedir.

Veriler eşzamanlı olarak görüntülenebilmektedir ve bununla ilgili gerçek zamanlı incelenebilir birçok grafik oluşturulmuştur.

Yapmış olduğum proje esnasında, proje ilerleyişini takip eden, teknik konularda desteğini esirgemeyen değerli hocam Doç.Dr. Mehmet BAKIR'a teşekkürlerimi sunarım.

Python ile projenin temeli olan duygu analizi konusunda veri madenciliği alanında yardımlarını esirgemeyen değerli hocam Öğr. Gör. Ömer Dağistanlı'ya teşekkür ederim.

Engin Karataş

Yozgat 2022

ÖZET .....	i
ÖNSÖZ .....	ii
İÇİNDEKİLER .....	iii
ŞEKİLLER DİZİNİ .....	v
TABLolar DİZİNİ .....	vi
SİMGELER VE KISALTMALAR .....	vii
1.GİRİŞ .....	1
2.YÖNTEM VE ARAÇLAR.....	4
2.1.PYTHON PROGRAMLAMA.....	4
2.1.1.PYTHON KÜTÜPHANELERİ.....	4
2.2.AWS (AMAZON WEB SERVICE) NEDİR? .....	6
2.2.1.AWS'NİN SUNDUĞU ÖNEMLİ HİZMETLER: .....	7
2.3.SSH PROTOKOLÜ NEDİR? .....	8
2.4.ELK STACK NEDİR? .....	8
2.4.1.ELASTİC SEARCH NEDİR? .....	9
2.4.2.KIBANA NEDİR? .....	12
2.4.3.LOGSTASH NEDİR?.....	13
2.4.4.BEATS NEDİR? .....	14
2.5.ELK STACK İLE NELER YAPILABİLİR?.....	14
2.6.ELASTİC.CO ELASTİC SEARCH NEDİR? .....	14
2.6.1.ELASTİC.CO ÜZERİNDEN ELASTİC SEARCH KULLANIMININ AVANTAJLARI.....	15
2.6.2. NO SQL VERİ TABANI İLE SQL VERİ TABANININ KARŞILAŞTIRILMASI.....	15
2.6.3.ELASTİC SEARCH ARAÇLARI.....	16
2.7.TWİTTER DEVELOPER ACCOUNT(TWİTTER GELİŞTİRİCİ HESABI) NEDİR? .....	16
2.7.1.TWİTTER GELİŞTİRİCİ HESAP TÜRLERİ .....	17

2.7.2.HOBİ AMAÇLI GELİŞTİRİCİ HESABI .....	18
3.YAPILAN ÇALIŞMALAR.....	19
3.1.TWİTTER DEVELOPER HESABI API BAŞVURU SÜREÇLERİ.....	19
3.1.1.PROJEDE TWİTTER GELİŞTİRİCİ HESABI .....	19
3.2.BULUT SUNUCU HİZMETİNE SSH PROTOKOLÜ İLE ERİŞİM.....	19
3.3.AWS EC2 SANAL MAKİNE KULLANIMI .....	20
3.3.1.AWS SANAL MAKİNE SEÇİMİ.....	20
3.4.API ANAHTARININ GÜVENLİĞİNİN SAĞLANMASI.....	20
3.4.1.GİTHUB NEDİR? .....	21
3.5.PYTHON İLE VERİ ÖN İŞLEME .....	22
3.6.PYTHON İLE SENTİMENT(DUYGU) ANALİZİ .....	23
3.6.1.PYTHON İLE DUYGU ANALİZİ .....	23
3.7.PYTHON İLE VERİ TABANI İŞLEMLERİ.....	24
3.7.1.ELASTIC CLOUD API ANAHTAR ERİŞİMİ .....	24
3.8.ELASTİC SEARCH API İLE VERİ TABANI İŞLEMLERİ .....	25
3.8.1.CRUD NEDİR? .....	26
3.8.2.ELASTICSEARCH API İSTEKLERİNİN ÇALIŞTIRILMASI.....	26
3.9.INDEX EXPORT CSV İŞLEMİ .....	28
3.9.1.EXPORT İŞLEMİ İÇİN RAPOR OLUŞTURULMASI .....	28
3.9.2.RAPOR GÖRÜNTÜLEME VE İNDİRME İŞLEMİ.....	28
3.10.KİBANA ÜZERİNDEN MONİTORİNG, ANALYTİCS VE VİSUALİZATİON	29
3.10.1.PROJE ÖZELİNDE OLUŞTURULAN GÖRSEL ANALİZLER .....	30
3.11.UKRAYNA RUSYA SAVAŞININ İZLENMESİ.....	33
3.12.DÜNYA HARİTASINDA KONUMA GÖRE DUYGU ANALİZİ .....	35
3.12.1.DÜNYA HARİTASI YAZILIMININ GELİŞTİRİLMESİ.....	35
3.13.EKSTRA ÇALIŞMALAR.....	38
4.SONUÇLAR.....	39
5.KAYNAKLAR .....	40
ÖZGEÇMİŞ	

## ŞEKİLLER DİZİNİ

SAYFA

Şekil 1. Projenin amacını gösteren diyagram .....	1
Şekil 2. Projenin işleyişini gösteren görsel.....	2
Şekil 3. ELK Stack Nedir .....	9
Şekil 4. Elastic Search Full-Text Search çalışma prensibi .....	12
Şekil 5. ELK Stack çalışma adımları.....	14
Şekil 6. Navbar(yan menü) > Manage This Deployment sayfası butonu.....	24
Şekil 7. Manage This Deployment sayfasında bulunan Cloud ID metni .....	25
Şekil 8. Kibana gösterge paneli, yeni görselleştirme ekle(create visualization) .....	29
Şekil 9. Bir e-ticaret sisteminin Kibana ile eş zamanlı izlenmesi .....	30
Şekil 10. Lokasyon Bazında Kayıt Sayısı. ....	30
Şekil 11. Toplam Tweet Sayısı.....	31
Şekil 12. Ortalama Duygu Analiz Değerleri. ....	31
Şekil 13. Pozitif ve Negatif Tweet Oranı ve Sayısı. ....	31
Şekil 14. Ülke ve Konuma Göre Duygu Analizi. ....	32
Şekil 15. Yoğunluk Değerleri bazında Lokasyon, Yazar ve Mesaj Tablosu.....	32
Şekil 16. Yoğunluk Değerlerinin Dağılım Grafiği. ....	32
Şekil 17. En Çok Takipçi Sayısına Sahip Lokasyonlar. ....	33
Şekil 18. 17 Haziran 2022 İncelenen Bar Grafiği. ....	33
Şekil 19. 10-11-12 Mayıs 2022 İncelenen Bar Grafiği. ....	34
Şekil 20. 3-4-5 Mayıs 2022 İncelenen Bar Grafiği. ....	35
Şekil 21. Avrupa haritası ve duygu analizi, dünya haritasından bir kesit. ....	35



## **TABLolar DİZİNİ**

## **SAYFA**

Tablo 1 Elastic Search ve SQL üzerindeki terimlerin karşılaştırılması.....	15
--	----

## **SİMGELER VE KISALTMALAR**

#	hashtag
SQL	Structered Query Language
ELK	Elastic, Logstash, Kibana
GUI	Graphical User Interface
JSON	JavaScript Object Notation
API	Application Program İnterface
CRUD	Create, Read, Update, Delete
ID	Identification
ELK	Elastic Search, Logstash, Kibana
HTTP	HyperText Transfer Protocol
CURL	Client URL

## 1. GİRİŞ

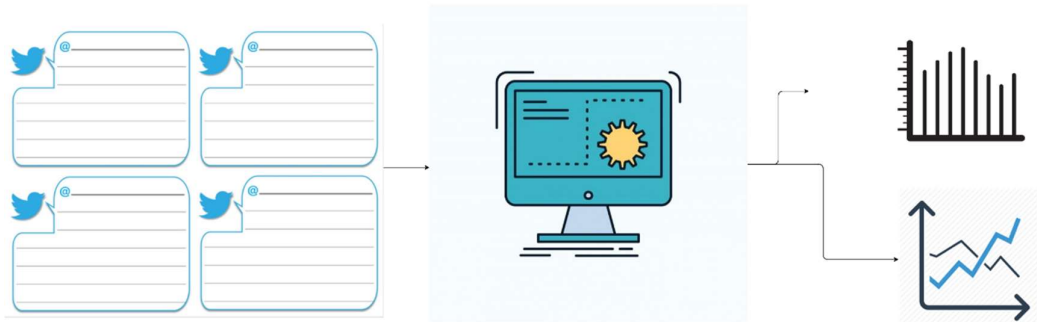
Twitter, günümüzde kullanılan, web 2.0 teknolojisi üzerinde çalışan bir sosyal medya, içerik oluşturma ve yayınlama sistemidir. Twitter'ın dünya genelinde kullanımı oldukça yaygın olduğundan, gerçek veriler için oldukça zengin bir kaynak sunmaktadır. Bu nedenle insan merkezli araştırmalarda, duygu analizlerinde araştırmacılar tarafından başvurulur. Kullanıcılar, konu bazında diğer insanların fikirlerini okuyabilir ve kendi fikirlerini de aynı konu üzerinden yayınlatabilir.

Twitter'da geçtiğimiz 2021 yılı içerisinde dünya çapında 322 milyon kullanıcı hesabı olduğu tahmin edilmektedir [1]. Twitter etkileşimli bir platformdur. Kullanıcı, diğer kullanıcıların Tweet'lerini okuyup beğenebilir ve yorum yapabilir. Kullanıcılar, kendi profilleri üzerinden kendi düşüncelerini yazı formatında paylaşabildiği gibi, fotoğraf ve video eklemeleri de yapabilirler.

Konu bazında içerik paylaşılmasına hashtag denir. Konu bazında içerik paylaşımları sekmesinden hashtag ile diğer kullanıcıların yorumlarını incelenebilir.

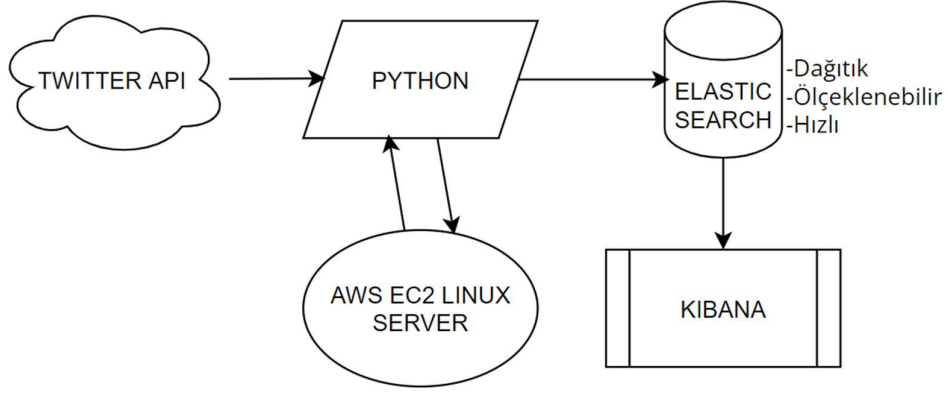
Araştırmacılar, Veri Bilimciler, Büyük Veri mühendisleri veya herhangi sebepten dolayı kullanıcı veya kullanıcılar üzerinde çeşitli analizler yapmak isteyen kurum/kuruluşlar çeşitli sınıflandırmalar ve analizler yaparlar.

Analiz ve sınıflandırmalar, WEB tarayıcı üzerinde gezinme yoluyla okunup yorumlanabilir. Fakat araştırmanın sonucunda bilgisayarlı sistemlere göre, çok daha az başarı ve veri elde edilecektir.



Şekil 1. Projenin amacını gösteren diyagram

Şekil 1 'de otomatizme edilmiş bir sistem üzerinden elde edilen bilgilerin grafikler üzerinde gösterildiği genel olarak gösterilmiştir. Projenin işleyişini gösteren görsel, Şekil 2'de gösterilmiştir.



Şekil 2. Projenin işleyişini gösteren görsel.

Tez için yapılan çalışmada Twitter'dan elde edilen bilgileri, bilgisayar bilimlerinin bir konusu olan veri madenciliği işlemlerinden geçirilmiştir. Veriler işledikten sonra kayıtların yazılabildiği veri tabanına veri olabilecek en anlamlı şekilde getirilip kaydedilmeye çalışılmış ve sınıflandırılan bu anlamlı sonuçlar, en kolay şekilde incelenen bilinmesi ve analiz edilebilmesi için görselleştirilmiştir.

### Tezin Amacı

Duygu analizi seçim zamanlarında politika belirlemekte, yatırım dönemlerinde yükseliş ve azalış durumlarını tahmin etmede, bir ürünün üzerindeki memnuniyeti ölçümlemek için kullanılabilir. Örneğin bir televizyon programının izleyiciler üzerindeki etkisini, beğenisi ölçülebilmesi için, izleyicilerin duygularının analiz edilmesi gerekir. Duyguların değerlendirilip sınıflandırılmasından sonra, verileri anlamlandırmak için grafiksel ara yüz oluşturulması projenin temel amacı olmasıyla birlikte, proje kazanımları aşağıdaki gibidir yapmak

- Çok sayıda insana ait veriyi Twitter aracılığıyla elde etmek
- Verileri veri madenciliği ile anlamlandırmak ve sonuçları sınıflandırmak

- Elde edilen sonuçları anlamlandıracak grafikler oluşturmak
- Anlamlı sonuçların bir kısmını Twitter üzerinde otomatik bir şekilde paylaşmak

Twitter`ın Kullanımı oldukça yaygın olduğundan, içerik bakımından oldukça zengin bir kaynak sunmaktadır. Kullanıcılar, konu bazında diğer kullanıcıların fikirlerini, konu için belirlenmiş hashtag `ler ile oluşturulmuş konu sayfalarından okuyabilir ve kendi fikirlerini de aynı konu üzerinden yayınlayabilir. Kullanıcılar, kendi profilleri üzerinden kendi düşüncelerini yazı formatında paylaşabildiği gibi, fotoğraf ve video eklemeleri de yapabilirler. Twitter etkileşimli bir platform olduğundan, kullanıcı, diğer kullanıcıların Tweet`lerini okuyup beğenebilir ve yorum yapabilir.

Projede twitter üzerinden alınan veriler, Python programlama dili içerisine getirildikten sonra sınıflandırılmış ve Ela\*stic Search veri tabanına yazılmıştır. Yazılan bu veriler, Elastic Search ile uyumlu çalışan görselleştirme, analiz aracı Kibana ile çalışılarak görselleştirilmiştir. Projede python programlama dili, duygu analizi yapılması ve sınıflandırılması için kullanılmıştır.

## **2. YÖNTEM VE ARAÇLAR**

### **2.1. Python Programlama**

Python programlama dili 5. Seviye yeni nesil bir programlama dilidir. Syntax(sözdizim) kuralları basit olduğundan ve hala geliştirilmeye devam edilen bir dil olduğundan dolayı Python oldukça popülerdir. Python programlama dilinin kullanım alanları aşağıdaki gibidir.

#### **Python programlama dilinin kullanım alanları:**

- Metin madenciliği
- Makine öğrenmesi
- Veri analizi
- Oyun geliştirme
- Web geliştirme
- Yapay zekâ
- Robotik uygulamaları

#### **2.1.1. Python Kütüphaneleri**

Kütüphane kullanımı sayesinde geliştirici her ayrıntıyı, teorik ve matematiksel kısımları bilgisayara anlatmak için veri yapıları seviyesinde düşünerek çözüm getirmeye gerek kalmadan geliştirme yapabilir. Python programlama dili birçok kütüphaneye sahiptir.

Python programlama dili içerisinde çeşitli amaçlara yönelik birçok kod geliştirilmiş ve diğer geliştiricilerin hizmetine bir çok kütüphane sunulmuştur. Bu kütüphaneler amaca yönelik algoritmaları ve geliştirme aşamasında ekstra yük olacak kod bloklarını ve mantık süreçlerini azaltmaya yönelik kullanılan yardımcılardır. Python diline ait proje özelinde geliştirilmiş kütüphaneler aşağıda tanımlanmış ve projedeki kullanım alanları açıklanmıştır.

##### **2.1.1.1. Tweepy**

Tweepy adındaki Python kütüphanesi, Twitter API'yi kullanabilmek için geliştirilmiş ve hala geliştirilmeye devam eden açık kaynak bir kütüphanedir. Tweepy, tüketici ve erişim

anahtarlarını kullanarak, Twitter API için tanımlanmış hedefe istekte bulunmak için Twitter API için standartlaştırılmış isteklerini gönderebilmek için geliştirilmiştir. API bilgileri kendi kütüphanesinde düzenlediği gibi, geliştiriciye sadece ulaşılmak istenilen hizmetler için genellikle tek satırda yazılabilen fonksiyonlar ile API isteği yürütülmesini sağlamıştır. Twitter API ile bağlantı sağlanırken consumer key(tüketici anahtarı), consumer secret(tüketici gizli anahtarı), access token(erişim anahtarı), access token secret(erişim gizli anahtarı) adında Twitter API oturumundan, geliştirici hesabı özelinde verilen anahtarlara ihtiyaç duyulur. Python komutları Tweepy kütüphanesi tarafından çalıştırılırken bu 4 anahtar kullanılır.

#### **2.1.1.2. TextBlob**

TextBlob, metin verilerini işlemek için kullanılan bir kütüphanedir. Konuşma bölümü etiketleme, duygu analizi, sınıflandırma, çeviri ve daha fazlası gibi yaygın doğal dil işleme işlemlerini yapmak için basit bir kullanım sağlayan Python kütüphanesidir. Türkçe metinleri TextBlob kütüphanesi aracılığıyla herhangi bir yabancı dile çevirip, TextBlob kütüphanesiyle duygu analizini yapmak mümkün olmaktadır. TextBlob kütüphanesiyle sınıflandırılan duygular, word Map(kelime bulutu), Heat Map(ısı haritası) gibi çeşitli yöntemler ile görselleştirilerek yorumlanması ve analiz edilmesi kolaylaştırılabilir.

#### **2.1.1.3. Elastic Search Python Kütüphanesi**

Elastic Search veri tabanı üzerindeki her türlü veri tabanı işlemini gerçekleştirmek için kullanılır. Elastic Search komutlarının geliştiriciye en kolay şekilde kullanması için geliştirilmiş bir kütüphanedir. Bu kolaylığının yanında hantal kalmaktadır. Elastic Search veri tabanı işlemleri bu kütüphane üzerinden yürütülmektedir. Proje özelinde Kısım 1.3'te bahsedilen ELK Stack Elastic.co bulut hizmeti üzerindeki veri tabanına Cloud ID adındaki özel bir anahtar ve deployment(dağıtım)'ın kullanıcı adı ve şifresi girilerek bağlantı sağlanıp yönetilmiştir. Yazılan kodlar, ELK bulut hizmeti üzerindeki veri tabanını doğrudan etkilemekte ve değiştirmektedir.

Elastic Search kütüphanesi, elastic.co bulutuna uzaktan bağlandığı için bu Python client(istemci)'i olarak adlandırılır.

#### **2.1.1.4. Geopy**

Geopy, Python geliştiricilerinin coğrafi konumlarını ve diğer veri kaynaklarını kullanarak dünya çapındaki adreslerin, şehirlerin, ülkelerin ve yer işaretlerinin koordinatlarını kolayca bulmasını sağlar.

#### **Geopy Nominatim**

Nominatim, adınızı ve adresinizi verdikten sonra size veri sağlayan ve özel anahtar girilmesine gerek kalmayan API olarak da bilinen ücretsiz bir araçtır. Proje özelinde konumlarının enlem ve boylamlarını almak için bu class kullanılır.

#### **2.1.1.5. Pandas**

Pandas, veri işlemesi ve analizi için Python programlama dilinde yazılmış olan bir yazılım kütüphanesidir. Bu kütüphane temel olarak zaman etiketli serileri ve sayısal tabloları işlemek için bir veri yapısı oluşturur ve bu şekilde çeşitli işlemler bu veri yapısı üzerinde gerçekleştirilebilir olur. [2] Pandas içerisindeki tablo şablonlarından biri olan Data Frame Yapıları veri ön işleme adımlarında kolaylık sağlanması için kullanılmıştır.

#### **2.1.1.6. Re Kütüphanesi**

Re kütüphanesi, regular expression (düzenli ifadeler) işlemleri için oluşturulmuş bir kütüphanedir.

#### **Regular Expression nedir?**

- Yoğun veri yığınlarından gerekli bilgiler ayıklar.
- Genellikle form bilgisi gönderirken kullanıcı tarafından girilen girişi kontrol eder.
- Verileri kullanım amacına uygun bir formata çevirir. Örneğin email gönderirken hatalı bir format kabul edilemez.

### **2.2. AWS (Amazon Web Service) Nedir?**

Amazon Web Service, bulut tabanlı, Web 2,0 üzerinde çalışan, istenilen miktardaki bulut işlem donanımları ve API'ları kiralanabilen bir platformdur. Amazon firmasının yan kuruluşu olan AWS, ödediğin kadar kullan mantığıyla web sunucuları, web siteleri için web servisleri sağlamaktadır. EC2 servisi, “cluster of computers” ile çalışır. Çalıştırılacak sistemler fiziksel olarak bir makineye ait değildir. Aynı küme içerisinde



tanımlı donanımlar yazılım ile kontrol edilir. Sanal bir makine ile gerçek bir bilgisayarı ayırt edecek en büyük fark donanımsal özelliklerdir. EC2 servisi üzerinden satın alınan bir makinede sanallaştırma teknolojileri kullanılır. Tercih edilen işletim sistemi özellikleri ve donanım kapasiteleri, EC2 gösterge paneli üzerinden seçilir.

AWS hizmetlerine erişebilmek için öncelikle [aws.amazon.com](https://aws.amazon.com) üzerinde kaydolmak gerekmektedir. Kaydolurken kredi kartı bilgileri girilmeli ve 1 USD(\$) ödeme yapılması gerekmektedir. Kaydolma işlemi tamamlandıktan sonra kullandığın kadar öde ilkesiyle AWS'nin sunduğu servisler kullanılabilir durumda olacaktır. AWS'nin sunduğu hizmetlerden bazıları aşağıdaki gibidir.

### 2.2.1. AWS'nin Sunduğu Önemli Hizmetler:

- **Amazon EC2 (Elastic Compute Cloud):** EC2, bulutta bulunan işletim sistemlerinin kontrol edilebildiği ve oluşturulan bu işletim sistemine istenildiği zaman erişile bilinen bir sanal makine hizmetidir.
- **AWS Amazon S3 (Simple Storage Service):** Bu serviste; dosya, klasör, resim, belge gibi nesneler saklanabilir. Bu dosyalar, host üzerinde bulundurulmak yönetilebilirliği ve performansı düşürür. Özellikle resimler, web sitesi ilk açıldığında performansın büyük bir oranını etkilemektedir. Bu nedenle birçok büyük e-ticaret firması S3 gibi sistemler kullanır. Resimler, site yüklenirken S3 servisinden maliyeti düşük şekilde getirilir.
- **EKS (Kubernetes için Elastik Konteyner Hizmeti):** Bu araç ile Kubernetes kurulum yapılmadan Amazon bulut ortamında kullanılabilir.
- **Elasticache:** Elasticache, bulut bilişim ile verileri önbelleğe alma yeteneğini getirir. Çok kullanılan veriler önbelleğe alınır. Bu sayede Elastic Search üzerindeki veri yük azaltılır
- **DynamoDB:** Yüksek performanslı bir No SQL veri tabanıdır.
- **Storage Gateway:** Sunucu üzerinde kurulan bir sanal makinedir. Firma üzerindeki veriler AWS üzerinde yedeklenir ve verilerin korunumu artar
- **Glacier:** Bu arşivleme hizmeti, dosyaları uzun süreler boyunca ucuz bir şekilde saklayabilir.

- **IAM (Kimlik ve Eriřim Yönetimi):** Kullanıcıların yönetilmesi ve birden çok gruplar oluşturulmasına olanak tanır.
- **Inspector:** Bir sanal makineye yüklenen ve tüm güvenlik açıklarını raporlayan bir güvenlik aracıdır.
- **Shield(Kalkan):** Web uygulamaları üzerinde giriş yaparken, DDOS saldırılarını engellemek için kullanılır.

### 2.3. SSH Protokolü Nedir?

Güvenli Kabuk (SSH), ağ hizmetlerinin güvenli olmayan bir ağ üzerinde güvenli şekilde çalıştırılması için kullanılan bir kriptolanmış ağ protokolüdür [3]. Uzun yıllar boyunca kullanılan çeşitli bağlantı kurma protokolleri örneğin rsh, rexec, rlogin vardır. Fakat bu bağlantı yöntemleri özellikle şifreleri açık bir şekilde gönderdiğinden oldukça güvensizdir. SSH protokolü, güvenlik açığı bulunan bu protokollere göre daha güvenlidir. İki ağ arasında SSH protokolü ile iki yerel ağ arasında bağlantı sağlanabilir.

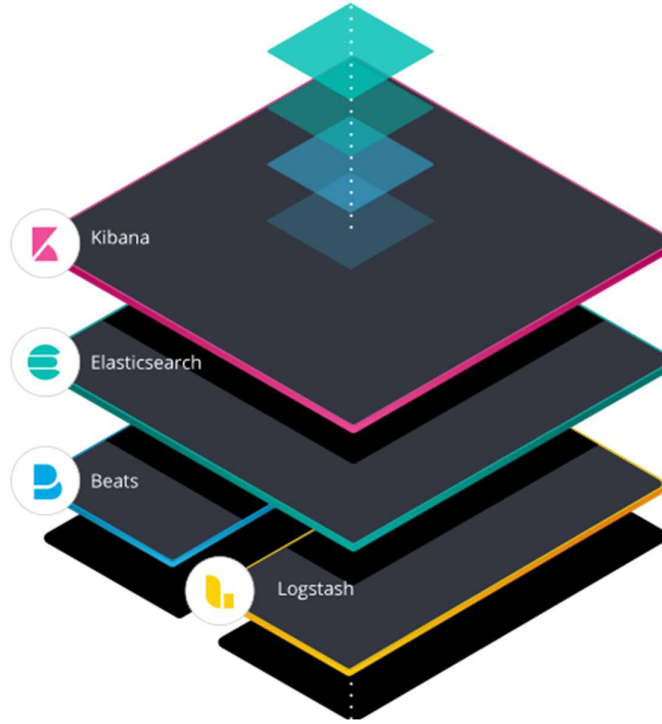
Uzaktaki bir bilgisayara bağlanmak için çeşitli şekillerde SSH protokolü çalıştırılabilir. SSH kullanım alanı genellikle Linux tabanlı işletim sistemlerine bağlanma şeklindedir. Bu bağlantı direkt işletim sistemine gönderilen bir komut satırı ile bağlantı sağlanması yöntemi şeklindedir.

SSH protokolü her ne kadar güvenli bir protokol olsa da Edward Snowden tarafından, Amerikan Ulusal Güvenlik Ajansı tarafından SSH oturum bilgilerinin çözülebildiğini, dinlenebildiğini ortaya çıkarmıştır [4].

AWS sanal makine üzerine bağlanırken SSH protokolü kullanılmıştır.

### 2.4. ELK Stack Nedir?

Şekil 2'de betimlenen Elastic Search, Kibana ve Logstash kelimelerinin baş harflerinden oluşan ELK Stack, beraber birçok işlemi bir araya getiren üç aracın kısaltımı için kullanılan bir kelimedir. Bu başlık altında ELK Stack içerisindeki ElasticSearch, Kibana ve Logstash teknolojileri anlatılacaktır.



Şekil 3. ELK Stack Nedir

#### 2.4.1. Elastic Search Nedir?

Elastic Search, Java dili ile yazılmış, Apache Lucene teknolojisi üzerinde çalışan, dağıtık sistemlerde çalışabilen, ilişkisel olmayan(unstructured) bir veri tabanı, analiz aracı ve Full-Text arama motorudur. Apache Lucene bir bilgisayar üzerinde arama yapabilir. Fakat dağıtık sistemlerde anlık gerçekleşecek aramalar söz konusu olduğunda Apache Lucene yetersiz kalmaktadır. Dağıtık sistemler üzerinde çalışabilmesi, Büyük Veri'ye sahip sistemlerde yüksek performans ile çalışması, esnek yapıya sahip olması gibi özellikleri Elastic Search'i güçlü kılan özelliklerindendir.

Veriler, Elastic Search üzerine veriler henüz kaydedilme aşamasında index'ler üzerinde ayrılır. Bu sayede aranan ifadeler kolaylıkla bulunabilir.

Elastic Search, açık kaynak bir sistem olmasından dolayı, Linux, MacOS, Windows üzerinde veya bu işletim sistemleri üzerinde oluşturulmuş konteynerler(sistemlerin dağıtılması ve hızlıca yönetilebilmesi için kullanılan bir sistem) üzerinde indirilip çalıştırılabilir.

Elastic Search sürüm kullanımı geriye uyumlu(backwards compatible)'dur. Bu nedenle geçmiş sürümlerde çalışan kodlar sonraki sürümlerde çalışmazken ileriki sürümlerde yazılan kodlar geçmiş sürümler ile çalışabilir.

#### 2.4.1.1.Elastic Search Yararları

- **Hızlıca arama yapılabilir:** Elastic Search'ın dağıtılmış altyapısı, büyük miktarda veriyi paralel olarak işler ve sorgular için en iyi eşleşmeleri hızla bulunmasını sağlar.
- **Elastic Search üzerine birçok yardımcı araç kurulabilmektedir:** Elastic Search görselleştirme için Kibana ile uyumlu bir şekilde çalışır. Geliştirilen uygulama özelinde ihtiyaç duyulan birçok araç kurulabilir.
- **Veri tabanı olarak kullanılabilir:** Elastic Search verilerin kaydedilip erişilebildiği bir veri tabanı olarak kullanılabilir ve Metin verilerinde oldukça hızlı sonuçlar alınabilir.
- **Tamamen açık kaynaklıdır:** Proje üzerindeki popülerliğin artmasıyla birlikte topluluğun projeyi daha çok desteklemesi sağlanmıştır. Açık kaynaklı yazılımlar şeffaf ve güvenilirdir. Projenin özel olma, korunma maliyetleri en aza indirilir.
- NoSQL veri tabanı gibi çalışır ve örneğin **MongoDB** gibi diğer NoSQL veri tabanlarından bilgileri kendi üzerine aktarabilir.
- **RestFul mimarisini üzerinde JSON objesi ile çalışır:** Elastic Search, basit bir REST tabanlı API, basit bir HTTP arabirimi ve şemasız JSON belgeleri sağlayarak hizmeti hızlı bir şekilde kullanmaya başlamanıza ve farklı kullanım durumları için uygulamalar tasarlamanıza olanak tanır.
- **Gerçek Zamanlı İşlemler Yapılabilir:** Gerçek zamanlıya yakın operasyonlar ile Veri okuma veya yazma gibi Elastic Search işlemleri genellikle bir saniyeden daha kısa sürede tamamlanır. Bu sayede uygulama üzerinde izleme ve anormallik tespiti gibi gerçek zamanlıya yakın kullanım amaçları için Elastic Search 'ten faydalanılır.
- **Kolay Uygulama Geliştirme Rahatlığı:** Elastic Search, Java, Python, PHP, JavaScript, Node.js, Ruby ve daha fazla dil için destek sağlar.

#### 2.4.1.2. Elastic Search Veri tabanı Terimleri ve İlişkisel Veri Tabanları Karşılaştırılması

- **Type(Tür):** İlişkisel bir veri tabanındaki bir tablodur ve verileri mantıksal bölümlere ayırır. Elastic Search, aynı dizinde birden çok Tür içerebilir.
- **Document(Belge):** Elastic Search ‘teki Tür(Type) yapısındaki bir satırı temsil eder. İlişkisel veri tabanlarında ise bir satırı temsil eder.
- **Mapping(Haritalama):** İndekslenen verinin veri tipinin tanımlanması işlemidir.
- **Fields(Alanlar):** Diğer veri tabanı türlerindeki sütunlara Elastic Search'te alanlar denir. Her belge birden fazla alan içerebilir.
- **Cluster(Küme):** Bir veya daha fazla düğümün(node) toplamıdır. Cluster(küme) nedeniyle, tüm verilerle dizin oluşturma ve arama işlevleri oluşturulur.
- **Indice:** Herhangi arama motorunda bir veri tabanı işlemi aranırken, Elastic Search, aramaları Indice adlı bir dizin dosyasında gerçekleştirilir. Bir Elastic Search kümesi(cluster), birden çok index içerebilir.

#### 2.4.1.3. Elastic Search Full-Text Search Özelliği

Elastic Search veri tabanı üzerinde yapılan aramalar çok hızlı bir şekilde gerçekleşir. Hızlı aramaları sağlayan Elastic Search özelliği, Full-Text Search özelliğidir. Yeni bir kayıt eklendiğinde metinleri index'lere ayırarak kaydeder. Yeni bir sorgu yazıldığında, bünyesine kaydettiği index'ler üzerinden arama yapar.

## Full-text Search

"keeper" kelimesi sorgusu çalıştırılması

1	The old night keeper keeps the keep in the town
2	In the big old house in the big old gown.
3	The house in the town had the big old keep
4	Where the old night keeper never did sleep.
5	The night keeper keeps the keep in the night
6	And keeps in the dark and sleeps in the light.

6 ADET  
SATIR(DOCUMENT)  
EKLEME

dictionary yapısı

Term	Documents
and	<6>
big	<2> <3>
dark	<6>
did	<4>
gown	<2>
had	<3>
house	<2> <3>
in	<1> <2> <3> <5> <6>
keep	<1> <3> <5>
keeper	<1> <4> <5>
keeps	<1> <5> <6>
light	<6>
never	<4>
night	<1> <4> <5>
old	<1> <2> <3> <4>
sleep	<4>
sleeps	<6>
the	<1> <2> <3> <4> <5> <6>
town	<1> <3>
where	<4>

Şekil 4. Elastic Search Full-Text Search çalışma prensibi

### 2.4.2. Kibana Nedir?

Kibana, Elastic Search ile uyumlu bir görselleştirme platformudur. Elastic Search cluster(küme) içerisinde depolanan verileri aramak, görüntülemek ve analiz etmek için web tabanlı ara yüz sağlar. (4) Eğer Elastic Search üzerindeki veriler zaman belirtiyorsa, bir alan içerisinde zamanı belirten timestamp(zamanı belirten 13 haneli bir tamsayı) ile belirtilmelidir. Kibana üzerindeki görselleştirme(visualize) bölümü, görselleştirme eklentilerinden biri ile verileri görselleştirme olanağı sağlar. Bilgiler tablolar, çizelgeler, haritalar, histogramlar şeklinde gösterilebilir.

#### 2.4.2.1. Kibana Avantajları

- **Etkileşimli Grafikler** : Kibana, Elastic Search veri tabanı ile eş zamanlı, etkileşimli olarak grafikler oluşturulmasını sağlayan raporlar sağlar. Zaman üzerinde dinamik arama yapılabilir. Belirli veri aralıklarına yakınlaştırıp uzaklaştırabilir.

- **Çizim desteği:** Kibana, coğrafi bilgileri sorunsuz bir şekilde veriler üzerine yerleştirile bilinen ve sonuçları bir harita üzerinde görselleştirebilen için güçlü bir coğrafi çizim yeteneğine sahiptir.
- **Önceden Oluşturulmuş Protokoller ve Filtreler:** Kibana'nın filtrelerini kullanarak, histogramlar, sorgular veya e-ticaret sisteminde en çok satılanlar gibi çeşitli analizler birkaç tıklamayla gerçekleştirebilir.
- **Kolay erişim paneli:** Panoları ve raporları kolayca oluşturabilir ve paylaşılabilir. Verileri görüntülemek için bir Chrome, Firefox gibi bir WEB tarayıcı yeterlidir.

### 2.4.3. Logstash Nedir?

Logstash, çeşitli kaynaklardan veri toplanmasına, anında kayıt alınmasına ve istenilen hedefe göndermeye olanak tanıyan hafif, açık kaynaklı bir sunucu tarafı bir pipeline(veri işleme hattı)'dır. Elastic Search için bir veri hattı olarak kullanılır. Logstash, toplanan günlük verileri toplar. Topladığı verileri JSON formatına çevirir ve Elastic Search Cluster yapısında depolar.

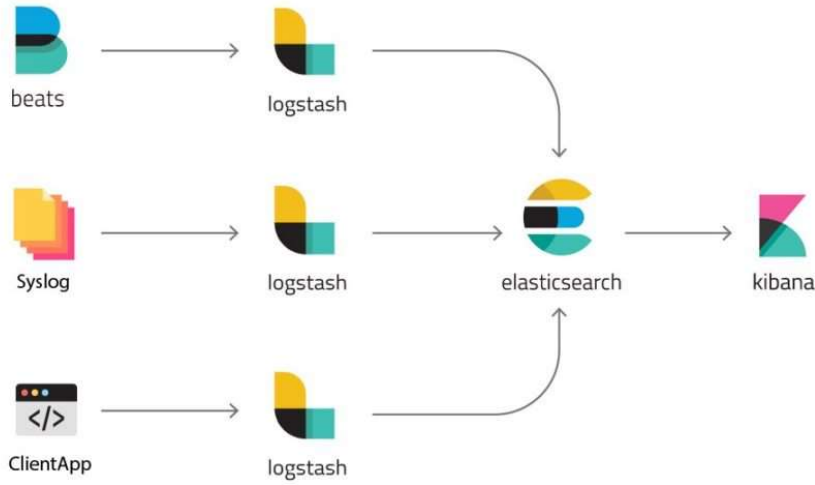
- **Yapılandırılmamış Verileri Kolayca Yükler:** Sistem günlükleri(log), WEB sitesi günlükleri ve uygulama sunucusu günlükleri dahil olmak üzere çeşitli veri kaynaklarından yapılandırılmamış verileri kolayca ortaya çıkarılmasına olanak tanır.
- **Önceden Oluşturulmuş Filtreler Sağlar:** Logstash önceden oluşturulmuş filtreler sağlar, Elastic Search'te dizini üzerinde özel sorgular, modeller oluşturmak zorunda kalmadan sorgulama imkânı sunar.
- **Birçok Hazır Eklentiye Sahip Olma Özelliği:** Github'da halihazırda mevcut olan 200'den fazla eklentiyle, veri hattını(modeller) özelleştirmek için gereken eklentiler diğer geliştiriciler tarafından oluşturulmuştur. İhtiyaçlarınıza uygun bir eklenti yoksa, proje özelinde oluşturulabilir.

#### 2.4.4. Beats Nedir?

ELK Stack içinde tanımlı bir diğer teknoloji Beats'dır. Kelime anlamı atım olan Beats, tek amaçlı veri gönderimleri için ücretsiz ve açık bir platformdur. Yüzlerce veya binlerce makine ve sistemden Logstash veya Elasticsearch'e veri gönderir.

#### 2.5. ELK Stack ile Neler Yapılabilir?

IOT, Web Servisleri, Veri tabanı Kayıtları, Cihaz durumları gibi sistemler ve uygulamalar üzerinde oluşan bilgi toplayıp, toplanılan bilgileri ve sistem altyapısı ve bilgilerini görselleştirme, analiz etmek için kullanılabilir. Bu sayede sistem üzerindeki yöneticinin toplanan bilgileri en hızlı şekilde analiz etmesine olanak tanır.



Şekil 5. ELK Stack çalışma adımları

#### 2.6. Elastic.co Elastic Search Nedir?

En iyi şekilde performans elde edilebilmesi için donanımlar, Elastic Search kullanım yoğunluğuna ve birim zamanda kullanılan API isteklerinin sıklığına göre iyileştirilmelidir. Elastic Search ve Elastic Search üzerinde çalışan yardımcı araçları kurmak, konfigüre etmek ve yedekleme işlemleri oldukça maliyetli bir süreçtir. Bu nedenle ELK ürünü ortaya çıkmıştır. Farklı bölgeler üzerindeki instance(örneklem)'ler üzerinde yönetimi ve performansı kolaylaştıracak konfigürasyonlar, kullanım



kolaylığını arttıracak araçlar Elastic.co ürün sitesi üzerinde geliştiriciler için sunulmuştur.

#### **2.6.1. Elastic.co Üzerinden Elastic Search Kullanımının Avantajları**

- Farklı ülke ve bölgelerde farklı bulut servisleri üzerinden instance'leri otomatik oluşturur. Bu sayede sistem yoğunluğuna göre farklı bölgelerdeki servislerin yoğunluğuna göre gelen istekleri dağıtır.
- Kibana, Enterprise Search, APM, Fleet gibi Elastic Stack ile çalışan araçları içerisinde barındırır. Sürümlerden kaynaklanan sorunlar meydana gelmeden, optimize sürümler ile çalışır.
- Elastic Search veri tabanı, araçları arasında hızlı bir şekilde incelenebilmesi
- Servis örneklerinin yoğunluğunu ve sağlığını hızlı bir şekilde ölçümlenebilmesi
- Elastic Cloud üzerinde çalışan RestfulAPI ile, farklı sistemler üzerinden çalışan yazılım dillerinden Elastic Search operasyonlarının hızlı bir şekilde yapılabilmesi
- Farklı deployment(dağıtım)'lar üzerinden hızlıca geçiş yapılabilmesi
- Deneme sürümü ile kullanılabilmesi
- Günlük log(kayıt) alınabilme
- Veri analizi ve görselleştirme yapılabilmesi için araçları barındırması
- Eş zamanlı olarak verileri izleyebilmesi
- Elastic Search Veri tabanı üzerinde sorgular, Manage(Yönetim) sekmesinden ara yüz ile hızlıca gönderilebilir.

#### **2.6.2. No Sql Veri tabanı ile Sql veri tabanının karşılaştırılması.**

SQL ilişkisel veri tabanı üzerinde tanımlı tablolar, veri tabanları Elastic Search veri tabanları arasında farklı ifadelerle karşılık gelmektedir. Tez üzerinde ilişkisel ve ilişkisel olmayan veri tabanlarının karşılaştırılması yapılmayacaktır. Fakat çoğu kitle tarafından bilinen çalışma alanı olan SQL'in Elastic Search ile farklılıkları aşağıdaki Tablo 1'de gösterilmesi uygun görülmüştür.

<b>Elastic Search(No SQL)</b>	<b>SQL</b>
DATABASE	INDEX
TABLE	TYPE
ROW	DOCUMENT
COLUMN	FIELDS
SCHEMA	MAPPING

Tablo 1 – Elastic Search ve SQL üzerindeki terimlerin karşılaştırılması

### 2.6.3. Elastic Search Araçları

Elastic Search veri tabanına girilen veriler üzerinde yönetim, görselleştirme sağlanması için bünyesinde birçok araç bulundurulur. Bu araçlardan proje özelinde kullanılanlar aşağıdaki gibidir.

#### 2.6.3.1. Developer Console(Geliştirici konsolu)

Developer konsol, ilgili deployment üzerinde ilgili index, type vb. üzerinde yönetim sağlamak için kullanılan bir I/O(Girdi/Çıktı) mantığıyla çalışan script(yazılım) çalıştırılabilir bir arayüzdür. Elastic.co üzerinden **Deployment>Manage>Dev Tool>Console** yolu takip edilerek developer consol'a erişilebilir.

### 2.7.Twitter Developer Account(Twitter Geliştirici Hesabı) Nedir?

Twitter üzerinden verilerin dağıtımı Twitter Developer Platform aracılığıyla gerçekleşir. Tüm verilere ölçülebilir, izlenebilir bir ortam sağlar. API isteği gerçekleştirilen hesapları yetkinliklerine göre inceler ve yapılan isteklerin geliştirici hesabının yetkilerini, Tweet yazma ve elde etme sınırlarını bir yazılım vasıtasıyla kontrol eder.

Twitter üzerinden başvuru süreçleri başlatılması gerekmektedir. Niyeti belirten birçok paragraf, e-posta ile yazışma sonrasında ancak hesap twitter tarafından verilebilmektedir. Bu nedenle proje amacının henüz başvuru yapılmadan belirlenmesi gerekmektedir.

Twitter API, birçok Twitter özelliklerine erişimi ve bunlarla etkileşim kurulması veya oluşturulmasını sağlar:

- Tweetler

- Kullanıcılar
- Spaces(twitter canlı yayın ortamı)
- Direkt Mesajlar
- Listeler
- Trendler
- Media
- Yerler

### **2.7.1. Twitter Geliştirici Hesap Türleri**

Twitter geliştirici hesabı üzerinden verilere erişim sağlanması için bir geliştirici hesabına ihtiyaç duyulur. Geliştirici hesabına, Twitter hesabı üzerinden başvurulur. Twitter geliştirici hesabına başvuru yaparken, geliştirici hesabının türü ne olursa olsun aşağıdaki sorulara cevap verilmesi Twitter geliştirici hesabı alınabilmesi için gereklidir.

- Twitter verilerini ve/veya API'lerinin nasıl kullanılması planlanmaktadır?
- Twitter verilerini analiz edilecek mi?
- Uygulama Tweet, Retweet, Beğen, Takip Et veya Direkt Mesaj işlevlerini kullanacak mı?
- Twitter dışında Twitter içeriği hakkında Tweetler veya toplu veriler görüntülenmek planlanmakta mıdır?
- Tarafınızdan oluşturulan ürün, hizmet veya analiz, Twitter içeriğini veya türetilmiş bilgileri bir devlet kurumunun kullanımına sunacak mı?

#### **2.7.1.1. Bilimsel Makale İçin Geliştirici Hesabı**

Bilimsel araştırmalar, akademik projeler için Twitter tarafından verilen bir hesap türüdür. Bu hesabın alınabilmesi için akademik bir makalede, başvuru geliştirici hesabındaki isim sahibinin, ilgili makalede adının geçmesi veya daha önceden bu tür çalışmalar yapıldığını ispat edecek nitelikte bir akademik makale başvuru sürecinde belirtilmelidir.

### **2.7.2. Hobi Amaçlı Geliştirici Hesabı**

Hobi amaçlı geliştirici hesabı, akademik amaçlı başvurulmuş geliştirici hesaplarına göre daha kolaydır. Başvuru süreçlerinde akademik amaçlı geliştirici hesaplarında göre daha az bilgi istenir.

### **3. YAPILAN ÇALIŞMALAR**

#### **3.1. Twitter Developer Hesabı API Başvuru Süreçleri**

Twitter üzerinden kullanıcıların bilgilerinin elde edilebilmesi, Twitter üzerinde bot hesabının paylaşım yetkisinin verilmesi ve diğer bütün Twitter üzerinden yapılan işlemlerin yapılabilmesi için Twitter geliştirici hesabına ihtiyaç vardır. Bu bağlamda, bu başlık altında Twitter API başvuru süreçleri anlatılmıştır.

##### **3.1.1. Projede Twitter Geliştirici Hesabı**

Twitter'in verilerinin ölçülebilir şekilde kullanılabilmesi için geliştiriciler için oluşturduğu, API erişim süreçlerinin yapıldığı platform üzerinde hobi Twitter API erişimi isteği gerçekleştirildi. Twitter API'ye erişim sağlanması için gerçekleştirilen bu başvuru sürecinde, projenin detayları hakkında, hangi amaçla, nerede ve hangi verilerle çalışılacağı bilgileri form olarak gönderilmiş ve sonrasında e-posta yoluyla devam eden iletişim sonucunda hobi hesabı, Twitter tarafından proje için tarafımıza verilmiştir. Hobi hesabı kapsamında aylık olarak 2 milyon adet Tweet fetch(getirme) edilebilir durumdadır [5].

#### **3.2. Bulut Sunucu Hizmetine SSH Protokolü ile Erişim**

SSH bağlantısında bir köprü niteliği oluşturacak bağlantıyı sağlayan ve giriş bilgilerini tutup bağlantının gerçekleştirilmesini kolaylaştıran programlar vardır. Bu bağlamda AWS Sanal makine bağlantısı gerçekleştirmesi, SSH protokolünü destekleyen PUTTY programı ile sağlanmıştır.

AWS üzerinde makineye bağlantı gerçekleştirilebilmesi için makine örneği oluşturulurken, AWS tarafından verilen özel bir anahtara ihtiyaç duyulur. Bu anahtar AWS ara yüzü üzerinden sadece makine örneği oluşturulurken bir sefere mahsus AWS tarafından verilir ve kaydedilmesi veya indirilmesi gerekmektedir. Bu anahtarı belirli bir algoritmaya göre çözümleyip özel bağlantı anahtarı oluşturulmasını sağlayan başka bir programa daha ihtiyaç vardır. Bu bağlamda Putty programının anahtar oluşturucu programı PuttyGen ile AWS üzerinden alınan anahtarı, RSA algoritmasına göre oluşturulur ve Putty programında özel bağlantı anahtarı olarak kullanılır. Algoritma sonucunda elde edilen özel anahtar Putty programına eklenir. Oturum açma için gerekli

bilgiler(ip adresi, port numarası vb.) de girildikten sonra ilgili session(oturumu) kaydedildikten sonra farklı zamanlarda hızlıca bağlantı sağlanabilir duruma gelir ve bağlantı sağlanır.

### **3.3. AWS EC2 Sanal Makine Kullanımı**

Sanal makineler genellikle IoT, WebServer, Windows Server, Restful Servis gibi servisler oluşturmak için kurulan bilgisayarlardır. Farklı sistemlerin veya doğrudan bir projenin taleplerini karşılar. Sanal makine seçimi, kullanılacak sistemin yoğunluğuna göre seçilmelidir. Projede özelinde bulut sunucu üzerinde kurulmuş bir sanal makine vardır. Sanal makine kurulum ve kullanımı ile geliştiricinin bilgisayarının bozulma, elektrik, internet kesintileri gibi riskler en aza indirilir. Sanal makineler, AWS üzerinde çalışan paylaşımlı Cluster yapısı içerisinde olduğundan oluşacak tehditlerden minimum şekilde etkilenecektir.

#### **3.3.1. AWS Sanal Makine Seçimi**

AWS EC2 sanal makine üzerinde seçim yaparken çalıştırılmak istenilen sistemin ne kadar performansa ihtiyaç duyduğu, hafıza ve belleğini ne ölçüde kullanacağı maliyet açısından oldukça önemli ve dikkate alınıp hesaplanması gereken bir konudur. Milyarlarca Tweet verisi saklanmak istenseydi terabaytlar seviyesinde bir donanıma ihtiyaç duyulabilirdi. Proje özelinde elde edilecek maksimum Tweet sayısı 2 milyon olduğundan, elde edilecek veri miktarı ve kapladıkları alan da düşünüldüğünde 10 GB veri depolama alanı proje özelinde gerekli ve yeterlidir. Ayrıca proje, data-intensity(veri-yoğun) bir uygulama olmadığından, tek çekirdek bir işlemci kullanımı yeterlidir. Projede Amazon Linux 2 AMI (HVM) - Kernel 5.10 64-bit ARM işlemciye sahip bir sanal makinesi kurulmuştur ve 10 GB depolama alanı, SSD disk yapısı içerisinde saklanmaktadır.

### **3.4.API Anahtarının Güvenliğinin sağlanması**

Açık kaynaklı kodların web 3.0 ile birlikte önem kazanmaya başlamasıyla birlikte, Github, Gitbucket, Gitlab gibi git altyapısını kullanan versiyon kontrol sistemlerinin kullanımı yaygınlaşmıştır. Proje özelinde açık kaynak kullanımı önemsenmiş olunup, tüm kodlar ve çalışmalar, açık kaynak olacak şekilde paylaşılmıştır.

Açık kaynak kod paylaşımının insanlığa ve evrensel kod gelişimine katkı sağladığı gibi, büyük kitleler tarafından kullanılan sistemlerde yazılımın güvenilirliğini artırır. Bununla birlikte geliştirilen yazılımlarda paylaşılmaması gereken özel içerikler bulunabilir. Bunlara örnek olarak, ilgili API bağlantı anahtarlarının paylaşılmaması örnek verilebilir. Bu anahtarlar, son derece önemli ve açık kaynak olarak paylaşılmaması gereken özelliklere sahiptir.

Bu API bilgilerinin saklanması için çeşitli yöntemler vardır. Bunlardan en çok kabul göreni, paylaşılan kodlar içerisinde tüm anahtarları tek bir dosya üzerinde saklamak ve bu dosyayı internete yüklemekten kaçınmak şeklindedir. Bu işlemi gerçekleştirmek için .env ve .environment\_example adında iki adet dosya oluşturulur. Bu dosyalardan asıl API anahtarlarını tutan dosya, ilgili programlama dili tarafından okunup program çalışma aşamasında programa girdi olarak verilir. Ve .env dosyası asla internete yüklenmez. Asıl internete yüklenmesi gereken dosya, .env dosyasının nasıl hazırlanması gerektiğini anlatan .environment\_example dosyasıdır. Bu dosya içerisinde, hangi değişkenler ile programın API anahtarlarını elde ettiği bilgisi programa verilir. Ve kullanıcı bu değişkenlere göre bir .env dosyası oluşturur.

Proje özelinde kullanılan Python ile built-in(dahili) gelen OS kütüphanesi ile .env dosyası içerisindeki değişkenlere erişilebilir. Aynı şekilde .env dosyasını okumak ve eşdeğer işlemleri yapmak için farklı programlama dillerinde ilgili kütüphaneler mevcuttur.

Proje en çok kullanılan versyon kontrol sistemi olan Github'a yüklenmiştir.

#### **3.4.1. Github Nedir?**

- Hub, sosyal ağ anlamı taşır. GitHub ise yazılımların sosyal ağıdır. Sosyal bir ortam sunmasıyla birlikte Git altyapısını kullanarak kodları yönetmemizi sağlar.
- Github, git komutlarını yazmadan, bir ara yüz vasıtasıyla Git işlemleri yapılmasını kolaylaştırır.
- Git altyapısını kullanarak projede yapılan değişiklikleri komut satırı(cmd) ekranından görmek yerine, kullanıcıların rahatça anlayabileceği ekranlarda yani UI(user interface/grafiksel kullanıcı ara yüzü)'da izlemesine olanak tanır.

- Dünya üzerindeki milyarlarca sayfa kodu, milyonlarca projeyi hızlıca arayabilmeyi sağlar.
- Yapılan projelerin yayımlanmasını, lisanslanmasını, yönetilmesini sağlar.
- Ekipçe kod geliştirmeyi sağlar.
- Git öğrenim süresini oldukça düşürür. Teknik bilgi gerektirmez. Geliştirme aşamalarındaki sıkıntıları en aza indirir.

### 3.5.Python ile Veri Ön İşleme

Ön işleme adımları, verileri temizleyen ve makine öğrenimi modellerinin verileri daha doğru kullanmasını sağlayan bir dizi yöntemdir. Gerçek hayatta üzerinde çalışılan veri bilimi projelerinde toplanılan veriler istenilen kadar temiz olmayacaktır. Ön işleme/ön eleme adımlarında tutarsız, eksik ve gürültülü veriden gerçek veri temizlenilmeye çalışılır.

Python programlama dilinde ön işleme adımları aşağıdaki şekilde gerçekleştirilmiştir.

Aşağıdaki kod, Twitter`dan gelen tweet`leri okur ve bölüm 1.1.1.5`te anlatıldığı üzere bir dataframe`a dönüştürür. Tanımlanan search\_words fonksiyonuna apply fonksyonu vasıtasıyla tweet parametre olarak gönderilir. Gönderilen parametreye göre bir regex işlemi uygulanır. Bu işlemin sonucunda kelimeler hariç tüm noktalama işaretleri, fazla boşluklar kaldırılmış olunur. Ve işlemin sonucunda bir dizi oluşturulur. Oluşturulan dizi result isimli değişkene atanır. Bir sonraki satırda Join() fonksyonu çalışır. Bu fonksyon verilen dizinin tüm elemanları arasına join fonksiyonuna parametre olarak gönderilen değeri ekler ve string olarak çıktı üretilir. search\_words fonksiyonu sonucunda return edilen değer, aralarında boşluk bulunan kelimelerden oluşan bir string(metin) ifadesi olmuş olur.

df["only\_words"] ifadesi, dataframe içerisindeki only\_words sütununu ifade eder. Bu sütuna ön işleme yapılan veriler eklenmiş olunur. Son adımda, tweetText değişkenine, ilk sütunun içerisinde bulunan metne atama yapılır.

```
df = pd.DataFrame({"tweetTextData": [dict_data["text"]],})

def search_words(text):
    result = re.findall(r"\b[^\d\W]+\b", text)
```



```
return " ".join(result)

df["only_words"] = df["tweetTextData"].apply(lambda x:
search_words(x))
tweettext = df["only_words"].values[0]
```

### 3.6.Python ile Sentiment(Duygu) Analizi

Projenin temel amaçlarından olan Twitter üzerinden duygu analizi, Python programlama dili kullanılarak yapılmıştır. Twitter API bağlantısı için Tweepy kütüphanesi kullanılmıştır. Anlık olarak Tweet'ler Python programlama dilinde, Tweepy kütüphanesi üzerinde tanımlı bir kütüphane olan StreamListener kullanılarak dinlenmiştir. Tweepy kütüphanesi vasıtasıyla Tweet'ler, birer birer getirilir ve JSON formatında dönüştürülür.

Elastic Search veri tabanına Twitter'dan getirilen verilerin yazılması sağlanmıştır. Python ile Elasticsearch-Py veri tabanı arasındaki bağlantıları, ekleme, getirme gibi işlemler için Python programlama dili için yazılmış Elasticsearch-Py kütüphanesi kullanılmıştır. Olumlu ve olumsuz sınıflama TextBlob kütüphanesi ile gerçekleştirilmiştir. 1 ile -1 arasında olumlu ve olumsuz şeklinde sınıflandırıp, Elastic Search veri tabanına kaydedilmiştir. Kibana ile görselleştirme işleminin gerçekleştirilmesi için, Elastic.co üzerinden deneme hesabı üzerinde Elastic Search Kibana ile görselleştirme yapılacaktır.

#### 3.6.1. Python ile duygu analizi

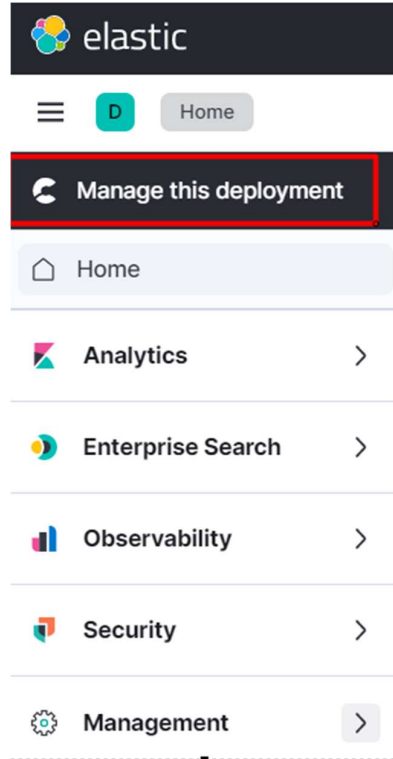
Projenin mantıksal gerçekleşmesi gereken her bir işlem Python dili üzerinden gerçekleşmektedir. Python dili ayrıca çeşitli API'lere bilgi taşınması, getirilmesi ve aksiyon alınması hususunda stabil bağlantı koşullarını sağlayan teknolojilere sahiptir. Python programlama dili ile Twitter API'a bağlantı gerçekleştirilir. Bu bağlantı, Twitter API üzerinden geliştiricilere sağlanan oturum açma anahtarı(Authentication Key) ile sağlanır. Python programlama diliyle Twitter arasında bağlantı kurulmasını sağlayan Twitter API Tweet'ler üzerinde izin verilen get(elde etme) işlem limitleri, proje özelinde elde edilen hobi hesabının limitleri kadardır.

### 3.7. Python ile veri tabanı işlemleri

Python ile Elastic Search arasında iletişim kurulduktan sonra CRUD işlemlerinin gerçekleştirilmesi için kısım 1.1 de anlatıldığı üzere Python Elastic Search istemcisinin elastic.co'ya bağlanması gerekmektedir. Bunun için API bağlantı giriş bilgilerine(kullanıcı adı, şifre, cloud ID) ihtiyaç vardır. API bilgileri, elastic.co bağlantı kullanıcı adı ve şifresi, deployment oluşturulduğunda bir defaya mahsus kullanıcıya gösterilir. Bu bilgileri korumak ve saklamak kullanıcının sorumluluğundadır. Cloud ID bilgisi, ilgili deployment üzerinde tanımlı ve her zaman erişilebilir 256 bit ile kriptolanmış bir anahtardır. Bu key(anahtar)'e erişim aşağıdaki başlık altında anlatılmıştır.

#### 3.7.1. Elastic Cloud Api Anahtar Erişimi

Elastic search cloud id alınabilmesi için elastic.co üzerinden şekil 5'te görüldüğü üzere navbar üzerinde bulunan “Manage this deployment” yazısına tıklayarak ilgili sayfaya girilmelidir.



Şekil 6. Navbar(yan menü) > Manage This Deployment sayfası butonu.

Bu sayfa üzerinde ilgili deployment'e ait instance bilgileri, var olan application endpointleri, cluster ID'leri ve yazılım ile erişilip, işlemleri yazılımla yapabilmek için şekil 6'da görüldüğü üzere gerekli Cloud ID(deployment\_adi:erişim anahtarı) bulunmaktadır.



**Şekil 7.** Manage This Deployment sayfasında bulunan Cloud ID metni

### **3.8. Elastic Search API ile Veri Tabanı İşlemleri**

Elastic.co üzerinde Elastic Search veri tabanına, Python programlama dili ile bağlantı kurulmuştur. Veri tabanı işlemleri, Python programlama dili ile, Elasticsearch-Py kütüphanesini kullanarak gerçekleştirilir.

Elastic Search Restful mimari üzerinde kurgulanmıştır bu nedenle erişimler RestAPI teknolojisi üzerinde çalışır. Bu nedenle diğer Restful teknolojiler ile tam uyumlu ve JSON formatında veriler ile haberleşmeye tam uyumludur. Bu işlemler CURL ile veya elastic.co üzerinden CURL yazımını basitleştirmek için hazırlanmış developer console üzerinden gerçekleştirilir.

#### **CURL Nedir?**

cURL, çeşitli protokoller kullanarak, veri aktarımı sağlamak için kütüphane ve komut satırı aracı sunan bir bilgisayar yazılımı projesidir. cURL projesi, libcurl ve cURL olarak ikiye ayrılır. İlk olarak 1997 yılında yayınlanmıştır. İsminde bulunan "c" İngilizce'deki client kelimesinden gelmektedir. [6]

Python programlama diliyle yazılmış Elasticsearch-Py kütüphanesinde gönderilen istekler, HTTP request(http isteği) ile Elastic.co üzerindeki veri tabanına RestAPI ile gönderir. [7] Veri tabanı teknolojileri üzerinde en temel işlemler CRUD işlemleridir. Bir sonraki başlık altında CRUD işlemleri tanıtılacaktır.

### 3.8.1. CRUD Nedir?

CRUD kelimesi, Create Read, Delete ve Update kelimelerinin baş harflerinden oluşan bir kısaltmadır. Veri tabanında, kullanıcı tarafından yapılacak temel işlemlerin kısaltmasıdır.

CRUD kelimesindeki harfler aşağıda açıklanmıştır:

- Create: Veri tabanı üzerinde kayıt eklemek için kullanılan terimdir
- Read: Veri tabanından veri getirmek için kullanılan bir terimdir
- Delete: Koşullu veya koşulsuz şekilde kayıtları silmek terimidir
- Update: Veri tabanı üzerinde genellikle bir koşul belirterek belirli bir kaydı güncellemek için kullanılan bir terimdir.

### 3.8.2. Elasticsearch API isteklerinin çalıştırılması

Yapılan işlemler Geliştirici Konsolu vasıtasıyla yapılabilir ve yalnızca Elasticsearch API'lerini destekler. Kibana API'leri ile Konsol ile etkileşim kurulamaz ve bunun yerine curl veya başka bir HTTP aracı kullanılması gerekir. [8]

#### 3.8.2.1. Elastic Search ile Ekleme İşlemi

Ekleme yapma işlemi Elastic.co üzerindeki Elastic Search içerisindeki Index, Document ve Document ID belirtilerek yapılır. Eğer Document ID belirtilmez ise, ID otomatik artan şekilde değişir. Index üzerinde bir değişiklik yapıldığında, işlemin onaylanması için ilgili Index ismi yazılır ve refresh ile yenilenerek sonraki gelecek kod satırlarının güncel Index kullanılması sağlanır.

Aşağıdaki kod, elastic.co üzerinde 1000 adında bir document'i dolar index'i üzerine ekler. [9]

```
PUT dolar/_doc/1000
```

```
{
```

```
"account_number": 1000,
```

```
"balance": 65536,
```

```
"firstname": "Engin",  
  
"lastname": "Karatas",  
  
"age": 23,  
  
}
```

### 3.8.2.2.Elastic Search ile Okuma İşlemi

Python Elasticsearch-Py kütüphanesi, veri tabanı üzerinde hızlıca okuma yapılabilecek fonksiyona sahiptir. Arama işlemi gerçekleştirilirken, arama yapılmak istenilen Index ve koşul girilmelidir. İlişkisel veri tabanlarında tüm kayıtları getirmek için Select deyimi kullanılırken, Elastic Search üzerinde tüm verileri getirmek için Match\_All özelliği kullanılır. Match\_All özelliği, Query objesi içerisinde yazılan bir özelliktir. Sorgular, Query objesi içerisinde gönderilir.

- Aşağıdaki kod Elastic Search üzerindeki ukraine\_crisis\_topic isimli index`teki dokümanların hepsini listeler. [9]

```
GET ukraine_crisis_topic/_search  
{  
  "query": {  
    "match_all": {}  
  }  
}
```

- Aşağıdaki kod Elastic Search üzerindeki ukraine\_crisis\_topic isimli index`teki belgelerin JSON formatında görünümü iyileştirilmiş halinde çıktı üretir. Bu işlemler CURL ile yapılır. CURL kullanımı zor olduğundan developer console üzerinden istekler gerçekleştirilmesi önerilir.

### 3.8.2.3.Elastic Search ile Silme İşlemi

Silme işleminde silinecek işlem bir Index ise Index adı parametre olarak gönderilir. Silinecek işlem bir Document ise, Index ismi ile ilgili Document Type ve Document ID belirtilmelidir. Verilen bilgiler sağlandığı durumda, veri tabanındaki ilgili kayıt silinecektir.

Aşağıdaki kod, Developer Console üzerinde index içerisinde girilen sayıdaki document`in silinmesi işlemini gerçekleştirir. [10]

```
DELETE dolar/_doc/1
```

#### 3.8.2.4.Elastic Search ile Güncelleme İşlemi

Güncellenecek kaydın ilk parametreleri genelden özele doğru gitmektedir. Index, Document Type ve Document ID belirtilir. Belirtilen kayıt bulunduktan sonra bir diğer parametreyle güncellenecek obje yerine değiştirilecek yeni obje belirtilir. Güncellenecek kayıt mevcut ise, Elastic Search koşulları sağlayan bu yeni objeye güncellenecek obje ile değiştirir.

#### 3.9.Index Export CSV İşlemi

Elastic Search üzerinde oluşturulmuş index, içerisindeki tüm verileriyle birlikte export edilebilir. Bu çıktı sonucunda .csv adında standartlaştırılmış bir format ile çıktı üretilir. Bu çıktıyı elde etmek için

Yapılan işlemler sonucunda elde edilen .csv formatı Excell programında tüm sütunlar ve değerleri şeklinde sonuçlar görüntülenebilir.

##### 3.9.1. Export İşlemi İçin Rapor Oluşturulması

Index`in export işlemi gerçekleştirilmesi için ilgili indexin rapor olarak oluşturulması gerekmektedir. Elastic.co üzerinde **Analytics > Discover** yolu izlenir. Bu ekranda export edilecek fields(sütunlar) belirtilebilir, eğer belirtilmezse tüm sütunları kapsayan bir csv dosyası oluşturulabilir. Belirli bir filtre özelliğine göre veriler düzenlenebilir.

Rapor oluşturmak için aynı sayfa üzerinde bulunan **Share** butonuna basılıp, **CSV Reports** seçeneği seçilir. Bu ekranda yapılan değişiklikler kaydedilebilir.

##### 3.9.2. Rapor Görüntüleme ve İndirme işlemi

Raporlama özelliği, elastic.co üzerinde Stack Management başlığı altında tanımlanmıştır. **Management > Alerts and Insights > Reporting** yolu takip edilerek ulaşılabilir. Bu sayfa üzerinde oluşturulan tüm raporlar gösterilir. Oluşturulan rapor

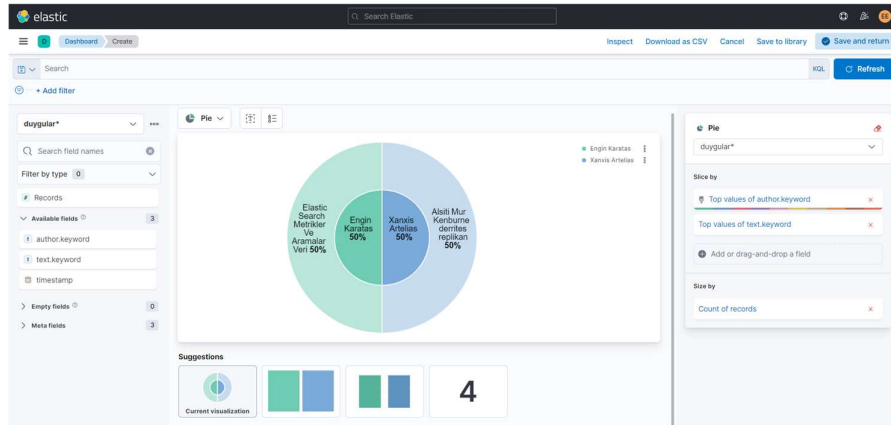
indirilmek istenildiğinde, raporun en sağında bulunan indirme butonuna tıklanıldığında indirme işlemi gerçekleştirilmiş olunur.

### 3.10. Kibana Üzerinden Monitoring, Analytics ve Visualization

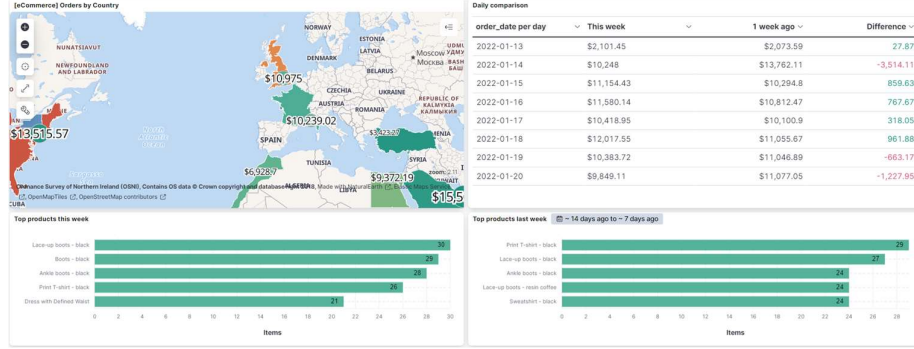
Kibana üzerinden monitoring(izleme), analytics(mantıksal çözümleme) ve görselleştirme işlemlerinin yapılabilmesi için, ilgili indexlerin data view(veri görseli) olarak tanımlanması gerekmektedir. Bunun için ilgili deployment'ta ki index ,data view olarak tanımlanmalıdır. Kibana>Data View yolu içerisinde **Create Data View** butonuna tıklayarak eklenmek istenen data view seçilir ve eklenir. Bu aşamadan sonra **Analytics** içerisindeki Dashboard, Canvas, Maps ilgili index değerlerine göre oluşturulabilir.

Kibana, toplanan anlamlı verileri analiz ettikten sonra bir görselleştirme işlemi gerçekleştirir. Şekil 7'te görüldüğü üzere hacim grafiği, pasta grafiği, çubuk grafik, çizgi veya dağılım grafiği şeklinde kolayca gösterilebilir. Gösterge Paneli'ne geçiş yapıldığında Şekil 8'te görüldüğü üzere bir e-ticaret sisteminin ayrıntılı grafikleri çıkarılabilir. Aylık ülkeye göre satışlar, geçen haftaya göre satış oranlarının değişimi, bu haftanın ve geçen haftanın en çok satılan ürünlerinin grafiği vb. grafikler Şekil 8'te görülmektedir.

Kibana Gösterge Paneli(dashboard), kaydedilmiş birden çok görselleştirmeyi tek bir görünümde birleştirmeye olanak tanır. Gösterge panelindeki öğeler düzenlenebilir ve yeniden boyutlandırılabilir. Doğrudan Kibana GUI'den karmaşık grafikler oluşturmak mümkündür. Gösterge panelinde oluşturulan tablolar kolayca kaydedilebilir.



Şekil 8. Kibana gösterge paneli, yeni görselleştirme ekle(create visualization)

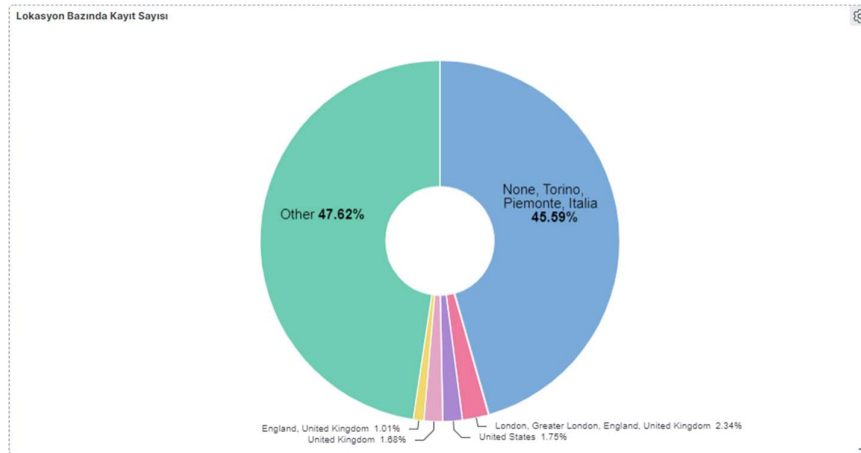


**Şekil 9.** Bir e-ticaret sisteminin Kibana ile eş zamanlı izlenmesi

Kibana üzerinde görselleştirme yapılabilmesi için, Elastic Search veri tabanı üzerinde doğru alanlara(field) uygun verilerin yazılması gerekmektedir. Kibana, Elastic.co üzerine kaydedilen verileri arar. Bunun için Kibana kullanılırken, üzerinde arama yapılacak Index verilmelidir.

### 3.10.1. Proje özelinde Oluşturulan Görsel Analizler

Tweet analizi sonucunda elde edilmiş değerler Kibana ile çeşitli analizlerden geçirilerek birçok analizin içerisinde bulunduğu Dashboard(Gösterge Paneli) üzerinde sıralanmıştır. Analytics > Dashboard Sayfası üzerinde incelenebilen ve oluşturulabilen grafiklerin bir kısmı aşağıda gösterilmiştir.



**Şekil 10.** Lokasyon Bazında Kayıt Sayısı.

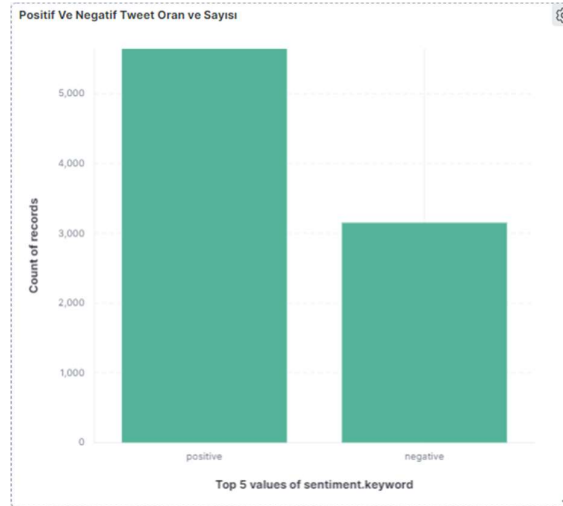




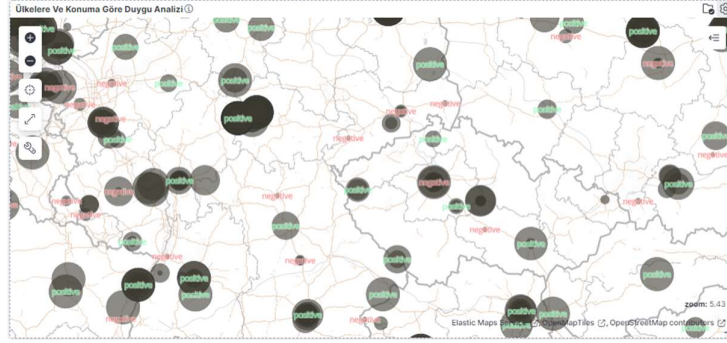
Şekil 11. Toplam Tweet Sayısı.



Şekil 12. Ortalama Duygu Analiz Değerleri.



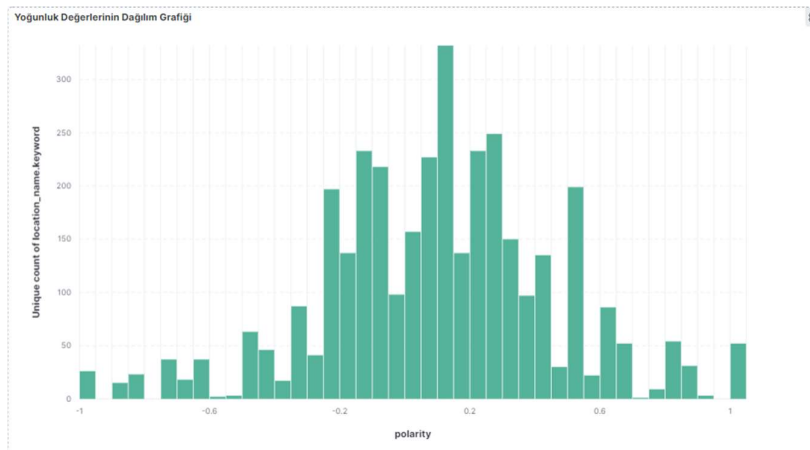
Şekil 13. Pozitif ve Negatif Tweet Oranı ve Sayısı.



Şekil 14. Ülke ve Konuma Göre Duygu Analizi.

Top 5 values of a	Top 5 values of location_name	Top 5 values of message.keyword	Unique count of	Median of pol
007dmax	None, Torino, Piemonte, Italia	RT @williamnhutton: The decision to leave the EU	1	-0.75
Other	London, Greater London, England	RT @williamnhutton: The decision to leave the EU	4	-0.75
Other	United Kingdom	RT @williamnhutton: The decision to leave the EU	3	-0.75
Other	United Kingdom	RT @snb19692: What is the point of Johnson goin	4	-0.714
Other	United Kingdom	RT @DomRQuinn: Ukraine set to join the EU. Brexit	3	-0.333
Other	England, United Kingdom	RT @DomRQuinn: Ukraine set to join the EU. Brexit	3	-0.333
Other	None, Torino, Piemonte, Italia	RT @POTUS: The Major Economies Forum met tod	47	-0.237
Other	United States	RT @POTUS: The Major Economies Forum met tod	8	-0.237
Other	Other	RT @POTUS: The Major Economies Forum met tod	38	-0.237
Other	None, Torino, Piemonte, Italia	RT @MaxBlumenthal: Just in: Ukraine sends urgen	37	-0.2
Other	Other	RT @MaxBlumenthal: Just in: Ukraine sends urgen	39	-0.2

Şekil 15. Yoğunluk Değerleri bazında Lokasyon, Yazar ve Mesaj Tablosu.



Şekil 16. Yoğunluk Değerlerinin Dağılım Grafiği.



Şekil 17. En Çok Takipçi Sayısına Sahip Lokasyonlar.

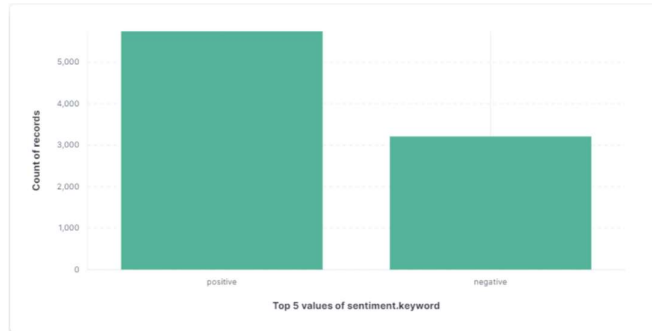
### 3.11. Ukrayna Rusya savaşının izlenmesi

Ukrayna savaşının sonucunda belirli tarihler arasında çeşitli veriler incelenmiştir. Bu incelenen verilerin sonucu, tezin sonuç kısmında gösterilmektedir. İncelenen verilerin sayısı ve hangi tarihler arasındaki maddelerde gösterilmektedir.

17 Haziran tarihinde 8.951 kayıt incelenmiştir.

#### 17 Haziran 2022 Gelişmeler:

- Ukrayna'nın doğusundaki Donetsk kentinde cuma günü meydana gelen bombalı saldırının ardından yanan bir evden duman yükseliyor. [11]
- Ukrayna'nın Avrupa Birliği'ne üyelik hedefi bir adım daha yaklaştı. [11]
- İngiltere Başbakanı Boris Johnson Kiev'i ziyaret etti. [11]
- Rusya Devlet Başkanı Vladimir Putin sert bir eleştiri yayınladı. [11]

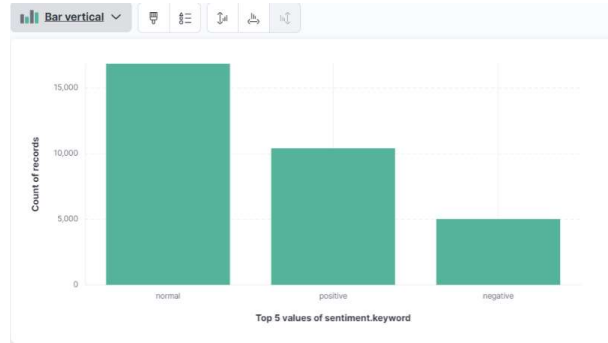


Şekil 18. 17 Haziran 2022 İncelenen Bar Grafiği.

10-11-12 Mayıs 2022 tarihleri arasında 32 bin veri incelenmiştir.

#### 10-11-12 Mayıs 2022 Gelişmeler:

- Ukrayna'nın Karadeniz kenti Odessa'da 10 Mayıs 2022'de Rus füzelerinin 9 Mayıs 2022'de vurulmasıyla yıkılan alışveriş ve eğlence merkezinin önündeki enkaz ve arabaların yanından kurtarma ekipleri yürüyor. [12]
- Mariupol'daki Azovstal çelik fabrikasında mahsur kalan Ukraynalı askerlerin eşleri, çarşamba günü Vatikan'daki haftalık genel görüşme sırasında Papa Francis ile bir araya geldi. [13]
- Siemens yaklaşık 170 yıl sonra Rusya'dan ayrılıyor. [14]
- Finlandiya, NATO'ya katılmak istediğini duyurdu. [14]

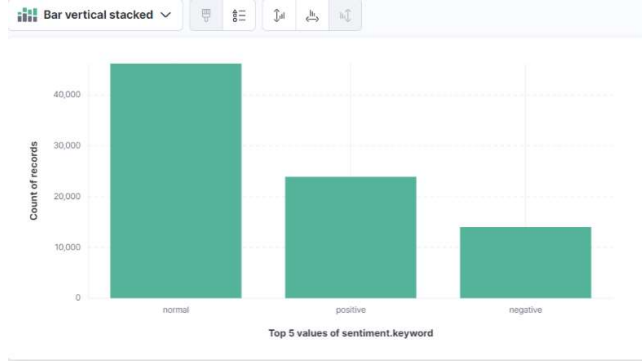


Şekil 19. 10-11-12 Mayıs 2022 İncelenen Bar Grafiği.

3-4-5 Haziran tarihleri arasında 84.114 kayıt incelenmiştir.

#### 3-4-5 Mayıs 2022 Gelişmeler:

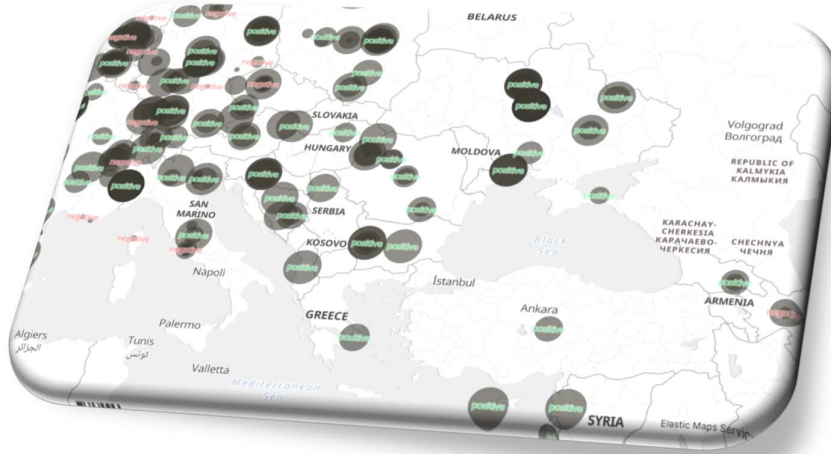
- Ukrayna'nın batısındaki Lviv kentinde salı günü düzenlenen hava saldırısının ardından koyu duman yükseliyor. Rusya, topçularının geçtiğimiz gün ülke genelinde yüzlerce hedefi vurduğunu söyledi. [15]
- Avrupa Birliği, kendisini Rus petrolünden kesmeyi teklif etti. [16]
- Ukrayna kuvvetlerinin stratejik kıyı kenti Mariupol'daki son kalesi olan Azovstal tesisinde çatışmalar devam ediyor. [16]



Şekil 20. 3-4-5 Mayıs 2022 İncelenen Bar Grafiği.

### 3.12. Dünya Haritasında Konuma Göre Duygu Analizi

Yapılan çalışmalar bölümü 2.4 kısmında bahsedilen duygu analizi işlemleri, dünya haritası üzerinde Şekil 9'de ki gibi gösterilmiştir.



Şekil 21. Avrupa haritası ve duygu analizi, dünya haritasından bir kesit.

#### 3.12.1. Dünya haritası yazılımının geliştirilmesi

Dünya haritası özelliği kullanılabilmesi için, elastic.co üzerinde ilgili index üzerinde veri tiplerinin doğru tanımlanması ve Python üzerinde verileri doğru formatta gönderilmesi gerekmektedir.

2.7.1.1 ve 2.7.1.2 kısımlarında Elastic Search ve Python üzerindeki yazılım geliştirme aşamalarından bahsedilecektir.

#### 3.12.1.1. Dünya haritasının Elastic Search üzerindeki geliştirmeleri

Dünya haritası özelliği kullanılabilmesi için, elastic.co index üzerinde veri tiplerinin doğru tanımlanması gerekmektedir. **Elastic Search Developer Console** üzerinde ilgili index'in location isimli property(özellik) tipini geo\_point olarak olarak değiştirilmelidir. Bu sayede Python tarafından eklenecek location isimli property'nin türü geo\_location olarak gösterilir. [9]

Bu bağlamda tipi değiştirilecek olan property'nin index'i yeni oluşturulmalıdır. Çünkü mevcut index üzerinde değiştirme yapmak hata meydana getirmektedir.

Aşağıdaki kod bloğu index\_adi isimli yere index'in adı yazılır.

```
PUT index_adi
{
  "mappings": {
    "properties": {
      "location": {
        "type": "geo_point"
      }
    }
  }
}
```

#### 3.12.1.2. Dünya haritasının Python üzerindeki geliştirmeleri

Python tarafından eklenecek veri içerisinde, geo\_location formatı olarak seçilen property için aşağıdaki şekilde girdi yapılmalıdır. Bu örnekte girilecek olan property, elastic.co üzerinde location olarak tanımlandığından, location adında bir property, index'e ekleme yapılan document içerisine aşağıdaki formatlardan herhangi biri girilmelidir şeklinde girilebilir. [9]

## Elastic Search Lokasyon Tip Değiştirme

```
PUT index_adi
{
  "text": "Geopoint as an object",
  "location": {
    "lat": 41.12,
    "lon": -71.34
  }
}
```

Veya:

```
PUT index_adi
{
  "text": "Geopoint as a string",
  "location": "41.12,-71.34"
}
```

Python üzerinden index üzerine ekleme yapılırken location bilgisi şekildeki gibi girilir.

Bir başka deyişle Elastic Search restful format başlığı adındaki kodlar, Python ile eklenmek istediğinde aşağıdaki şekilde girilmelidir.

```
es.index(
    index="ukraine_crisis_topic",
    doc_type="_doc",
    body={
        "location":{
            "lat":15,
            "lon":27
        }
    },
)
```

Proje özelinde lokasyon property'si içerisinde girilen lat ve lon değerleri, tweet'ler üzerinden alınan lokasyonların Geopy API'si sayesinde elde edilen coğrafi konum verileri ile dinamik olarak verilmektedir.

Enlem ve boylam, bir coğrafi koordinat sisteminin düzlemindeki bir konumu tanımlamak için kullanılan bir çift sayıdır (koordinatlardır). Sayılar ondalık derece biçimindedir ve enlem için -90 ila 90 ve bu değerler, boylam için -180 ila 180 arasındadır. [17]

### **3.13. Ekstra alıřmalar**

Tez alıřması esnasında, tezde beklenen konular dıřında yapılan alıřmayı bytmek iin eřitli denemeler yapılmıřtır. Yapılan ve bahsedilmemiř iřlemler ařağıdaki řekildedir.

- Elastic Search kurulumunu elastic.co rn haricinde manuel olarak linux sistem zerine kurulmuřtur.
- Linux Server zerinde Python kodlarını daha iyi test edilebilmesi iin Linux sistemi zerine Jupyter Notebook kurulmuřtur.
- Farklı Elastic Search srmleri ve migration(srmler arası geiřler) yapılmıřtır.
- Onlarca hatalar alınmıřtır ve zlmřtr.



#### 4. SONUÇLAR

Twitter`ın açık kaynak API`ini kullanarak, Tweet analizi gerçekleştirilmiştir. Girdi olarak Tweet`ler Twitter platformunda, en fazla 280 karakter uzunluğunda metin verileri alınmıştır. Çıktı olarak insanların konu/konular hakkındaki düşünceleri olumlu, olumsuz, normal şeklinde sınıflandırılmış ve görsel analiz yöntemiyle konum, mesaj, tarih, duygu yoğunluğu gibi elde edilen verilere göre analiz işlemleri yapılarak, çeşitli grafikler şeklinde visualization(görselleştirme) yapılmıştır. Elde edilen konum bilgilerinin harita üzerine konumu bulunarak harita üzerinde görselleştirme işlemi için kullanılmıştır. Elde edilen sonuçlarda Ukrayna savaşı ve dolar kuru incelenmiş ve çeşitli çıktılar elde edilmiştir.

##### **Ukrayna Savaşı Analiz Sonuçları**

3-4-5 Mayıs, 10-11-12 Mayıs, 17 Haziran Tarihleri arasında yapılan incelemelerde olumlu, olumsuz, normal olarak elde edilip sınıflandırılan verilerin birbirine oranlarına göre incelendiğinde, her ne kadar gelişmelerin insanlar üzerindeki etkileri değişebilse de ve gelişmelerin iyi veya kötü bir gelişme olarak dağılımı farklılık gösterse de, **yapılan çalışmalar sonucunda elde edilen olumlu/olumsuz/normal sınıflandırmaların korelasyonu, araştırmacının değerlendirmesi sonucuna göre Ukrayna savaşı özelinde doğruluğu saptanamamıştır.**

## 5. KAYNAKLAR

- [1] «Number of Twitter users worldwide from 2019 to 2024,» [Çevrimiçi]. Available: [statista.com/statistics/303681/twitter-users-worldwide/](https://statista.com/statistics/303681/twitter-users-worldwide/). [Erişildi: 1 Ocak 2022].
- [2] «Pandas,» 2022 02 15. [Çevrimiçi]. Available: <https://tr.wikipedia.org/wiki/Pandas>. [Erişildi: 2022 6 17].
- [3] «Güvenli Kabuk (SSH),» [Çevrimiçi]. Available: [https://tr.wikipedia.org/wiki/G%C3%BCvenli\\_kabuk](https://tr.wikipedia.org/wiki/G%C3%BCvenli_kabuk). [Erişildi: 2 Ocak 2022].
- [4] «Güvenli kabuk,» [Çevrimiçi]. Available: [https://en.wikipedia.org/wiki/Secure\\_Shell](https://en.wikipedia.org/wiki/Secure_Shell). [Erişildi: 20 Ocak 2022].
- [5] «Docs,» [Çevrimiçi]. Available: <https://developer.twitter.com/en/docs>.
- [6] «Curl,» 15 03 2022. [Çevrimiçi]. Available: <https://tr.wikipedia.org/wiki/CURL>. [Erişildi: 17 6 2022].
- [7] «Accounts,» [Çevrimiçi]. Available: <https://developer.twitter.com/en/docs/twitter-ads-api/campaign-management/api-reference/accounts>. [Erişildi: 20 Ocak 2022].
- [8] «Run Elasticsearch API requestsedit,» [Çevrimiçi]. Available: <https://www.elastic.co/guide/en/kibana/8.2/console-kibana.html#console-kibana>. [Erişildi: 17 6 2022].
- [9] E. Search, «Elk Documentation,» elastic.co, [Çevrimiçi]. Available: <https://www.elastic.co/guide/en/elasticsearch/reference/current/geo-point.html>. [Erişildi: 13 6 2022].
- [10] «Execute CRUD Operations in Elasticsearch,» [Çevrimiçi]. Available: <https://acloudguru.com/hands-on-labs/execute-crud-operations-in-elasticsearch>. [Erişildi: 13 6 2022].
- [11] «Russia-Ukraine war: What happened today (June 17),» 17 06 2022. [Çevrimiçi]. Available: <https://www.npr.org/2022/06/17/1105671092/russia-ukraine-war-what-happened-today-june-17>. [Erişildi: 18 6 2022].
- [12] «U.S. House passes \$40 billion Ukraine aid package; Ukraine economy predicted to contract 30%,» 10 5 2022. [Çevrimiçi]. Available: <https://www.cnn.com/2022/05/10/russia-ukraine-live-updates.html>.
- [13] «NPR,» 11 5 2022. [Çevrimiçi]. Available: <https://www.npr.org/2022/05/11/1098305702/russia-ukraine-war-what-happened>.

today-may-11.

- [14] «NPR,» 12 5 2022. [Çevrimiçi]. Available:  
<https://www.npr.org/2022/05/12/1098467785/russia-ukraine-war-what-happened-today-may-12>.
- [15] «NPR,» 3 5 2022. [Çevrimiçi]. Available:  
<https://www.npr.org/2022/05/03/1096122593/russia-ukraine-war-what-happened-today-may-3>.
- [16] «NPR,» 4 05 2022. [Çevrimiçi]. Available:  
<https://www.npr.org/2022/05/04/1096606111/russia-ukraine-war-what-happened-today-may-4>.
- [17] «Mapbox Docs,» [Çevrimiçi]. Available:  
] <https://docs.mapbox.com/help/glossary/lat-lon/#:~:text=Latitude%20and%20longitude%20are%20a,180%20to%20180%20for%20longitude..> [Erişildi: 6 13 2022].
- [18] M. Bajer, «Building an IoT Data Hub with Elasticsearch, Logstash and Kibana,» 2017.
- [19] «geeksforgeeks,» [Çevrimiçi]. Available: <https://www.geeksforgeeks.org/how-to-get-geolocation-in-python/>. [Erişildi: 8 6 2022].

## **ÖZGEÇMİŞ**

Engin Karataş, Eylül 1999 doğumlu Kocaeli, Gebze yerlisidir . Yozgat Bozok Üniversitesi Bilgisayar Mühendisliği lisans öğrencisidir. Birçok kez onur öğrencisi olarak seçilmiştir. Yazılım, donanım ve proje etkinliklerine aktif katılım göstermektedir. Kendini yazılım alanında geliştirmiş ve farklı kategoride yazılım projeleri geliştirmiştir. Ağ üzerinden çok oyunculu ve tek oyunculu oynanabilen oyunlar yazmıştır. Yazılım geliştirme prensiplerine hâkim olmayı kendine prensip edinmiştir. Java, Angular, C#, Nodejs, Javascript teknolojileriyle projeler geliştirmiştir. Drone Kontrol istasyonu, e-ticaret yönetimi, firma içi kullanıcıların raporlarının takibi ve yönetimi, ağ elemanları ve çeşitli donanım teknolojileri üzerinde monitoring(izleme) vb. projelerde bireysel veya grup halinde çalışmıştır.

PROJE KODLARI:

**Dosya adı:** project\_map\_location\_visualize\_ukraine.py

**Kod:**

```
import sys

# secret key protection for git commit
import os # package that allows to access env. variables

import json

# import tweepy library
import tweepy

# import textblob library
from textblob import TextBlob

# elk enterprise connection. import elasticsearch library for
python to connect elastic cloud instance
from elasticsearch import Elasticsearch

import pandas as pd
import re as re

# importing geopy library
from geopy.geocoders import Nominatim

sys.path.append("./environment")
from config import *

api_key = os.getenv("api_key")
api_key_secret = os.getenv("api_key_secret")
access_token = os.getenv("access_token")
access_token_secret = os.getenv("access_token_secret")

cloud_id = os.getenv("cloud_id")
user = os.getenv("user")
password = os.getenv("password")

es = Elasticsearch(cloud_id=cloud_id, http_auth=(user,
password))

# calling the nominatim tool
loc = Nominatim(user_agent="GetLoc")
```

```

class TweetStreamListener(tweepy.StreamListener):
    def on_data(self, data):

        dict_data = json.loads(data)

        #preprocessing
        df = pd.DataFrame({"tweetTextData":
[dict_data["text"]],})

        def search_words(text):
            result = re.findall(r"\b[^\d\W]+\b", text)
            return " ".join(result)

        df["only_words"] = df["tweetTextData"].apply(lambda x:
search_words(x))
        tweettext = df["only_words"].values[0]

        #sentiment analysis
        tweet = TextBlob(tweettext)

        print(tweettext)
        if tweet.sentiment.polarity < 0:
            sentiment = "negative"
        elif tweet.sentiment.polarity == 0:
            sentiment = "normal"
        else:
            sentiment = "positive"

        lat=0
        lon=0
        try:
            getLoc = loc.geocode(dict_data["user"]["location"])
            locationName = getLoc.address
            lat = getLoc.latitude
            lon = getLoc.longitude
            print(dict_data["user"]["location"])
            print(tweet.sentiment.polarity)
        except:
            print("Location Found Error")
        if(lat!=0 and lon!=0 and tweet.sentiment.polarity !=
0.0):
            print("sentiment")
            print(sentiment)
            print("polarity")
            print(tweet.sentiment.polarity)

```

```

        es.index(
            index="project_map_location_visualize_ukraine",
            doc_type="_doc",
            body={
                "author": dict_data["user"]["screen_name"],
                "date": dict_data["created_at"],
                "location": dict_data["user"]["location"],
                "followers":
dict_data["user"]["followers_count"],
                "friends":
dict_data["user"]["friends_count"],
                "time_zone": dict_data["user"]["time_zone"],
                "lang": dict_data["user"]["lang"],
                "message": dict_data["text"],
                "polarity": tweet.sentiment.polarity,
                "subjectivity":
tweet.sentiment.subjectivity,
                "sentiment": sentiment,
                "location":{
                    "lat":lat,
                    "lon":lon
                },
                "location_name":locationName
            },
        )
        print("Inserted: " +
str(dict_data["user"]["location"]))

    else:
        print("Insert failed to Elastic Search Cloud.
Reason: No location found")
        return True # continue evry record instance of the
tweepy tweet stream listener

    def on_error(self, status):
        print(status)

if __name__ == "__main__": #is used to execute some code only if
the file was run directly, and not imported
    listener = TweetStreamListener()
    auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
    auth.set_access_token(access_token, access_token_secret)

    stream = tweepy.Stream(auth, listener)
    stream.filter(track=["ukraine"])

```

**Dosya adı:** .env

**Kod:**

**#elastic**

cloud\_id =

testingelastic:dXMtY2VudHJhbDEuZ2NwLmNsb3VkLmVzLmlvOjQ0MyQ0Y2RiN2  
Q4NDFhODg0M5YTg0MDQ5NWE2ZjE0ODIiYiQ2NjVhOTM4NjBiMWE0ZWl4Yjc  
5YmI5NmI1ZDRmYjU0NA==

user = elastic

password = "3lGZd7Nn2XFJKN0DsCOTMWNk"

**#twitter**

api\_key = "AU7Ma83BEiaA7GdF3l9D9tYUf"

api\_key\_secret = "miQyH7ueEiwu7i0989MElgAmibfP99tyPHRIO6O6PI97mkdR"

#twitter api access made with access token. it is stateless authentication.

access\_token = "1374444599461113862-2XWdvEPY89FtsnMLY2E12I9VWtD8c"

access\_token\_secret = "XwIdWhZFXT8eqYAttgxv1mZwvWoZsaxdQ2AX2VIZCziEs"