iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VII **Month of publication:** July 2022

DOI: https://doi.org/10.22214/ijraset.2022.45702

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

High Performance Mining of Covid-19

Dr Girish Kumar D[1], Kavya B P[2], Kavyashree S[3], Asmiya Afsha[4]
[1, 2, 3, 4,]*Department of Computer Science and Engineering, Ballari Institute of Technology & Management, Visvesvaraya Technological University, 583104 Ballari, India.*

Abstract: ***The COVID-19 global pandemic is an unprecedented health crisis. Many researchers around the world have produced an extensive collection of literature since the outbreak. Analysing this information to extract knowledge and provide meaningful insights in a timely manner requires a considerable amount of computational power. Cloud platforms are designed to provide this computational power in an on-demand and elastic manner. Specifically, hybrid clouds, composed of private and public data centers, are particularly well suited to deploy computationally intensive workloads in a cost-efficient, yet scalable manner. In this paper, we developed a system utilising the Aneka Platform as a Service middleware with parallel processing and multi-cloud capability to accelerate the data process pipeline and article categorising process using machine learning on a hybrid cloud. The results are then persisted for further referencing, searching and visualising. The performance evaluation shows that the system can help with reducing processing time and achieving linear scalability. Beyond COVID-19, the application might be used directly in broader scholarly article indexing and analysing.***
Keywords: Covid -19, Cloud Computing, Aneka platform

I. INTRODUCTION

COVID-19 has given place to a global scale health crisis. Since the outbreak, a massive amount of research efforts have been poured into many aspects of this highly infectious disease. To help the research community, in March 2020, the White House and the Allen Institute for AI teamed up with many researchers and released the COVID-19 Open Research Dataset (CORD-19) [1]. As of July 2020, CORD-19 contained over 199,000 research papers, with nearly half of them being open access publications [2]. The rapid increase on the number of articles has posed a challenge for the research community to process these data in a timely manner. Furthermore, the general public is also interested in many aspects of the disease, especially on findings related to day-to-day life. As a result, tools and platforms that support machine learning approaches to extract knowledge from this vast amount of data are required.

This is a challenging endeavor as a considerable amount of computing power is needed by Extract, Transform, Load (ETL) and ML techniques in order to produce results in a reasonable amount of time. Cloud computing is the de facto standard to access computing resources on demand and on a pay-as-you-go manner. Hybrid Cloud Environments (HCEs) combine private data centers with resources offered by public cloud providers; thus enabling applications to save cost by using existing on premise resources and to scale onto public clouds if the private data center's capacity is not sufficient [3].

This approach can be greatly beneficial to organizations seeking to process the CORD-19 dataset. However, building an application using HCE is also a demanding task as it requires detailed knowledge of cloud computing techniques.In this context, we propose a system design and implementation to accelerate ETL processing and text classification based on ML techniques in an HCE. The architecture is designed with the following requirements in mind: scalability, availability, stability, high performance and portability.

To achieve these goals and for ease of development, the proposed application deploys on top of Aneka [4], which is a resource management framework that provides high level Application Programming Interfaces (APIs) and Software Development Kits (SDKs) to transparently enable the deployment of applications on cloud resources. It allows developers to focus on implementing their program logic without spending too much time considering deployment and scalability. When additional resources are required, they can be seamlessly acquired from different CSPs via Aneka dynamic provisioning mechanism.

II. OBJECTIVES

1) To Process the covid related data using the data mining technology.
2) To implement an Machine learning algorithms for text processing.
3) To implement an NLP –Natural Language Processing for efficient text classification.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

III. FUNCTIONAL REQUIREMENTS

1) This section presents the sentiment classification architecture for tweet data analysis. The architecture consists of five steps. The first step involves inputting the text dataset of tweets.
2) The second step is to pre-process tweets' text and reduce noise by reducing unwanted characters and symbols from the text.
3) In the third step, pre-processing step, the text is changed into a value based on its frequency.
4) In the fourth step, feature mapping is performed, and weights assigned by LSTM-RNN mapping features are changed to reduce overlapping between features. Improving weights by using the attention mechanism helps to select relevant information.
5) In the fifth step, a classifier is applied in the proposed approach using a basic softmax classifier.

IV. NON-FUNCTIONAL REQUIREMENTS

1) *Scalability:* All major components are horizontally and vertically scalable. When additional capability is required, the application can easily scale up by provisioning more powerful virtual machines, or scale out by adding more nodes.
2) *Availability:* Minio and Elasticsearch use shared-nothing architecture, which is designed for highly available systems. Aneka also has a robust mechanism to handle task/node failure automatically.
3) *Stability:* This can be achieved using fail-fast and idempotent processing. If a task fails for any reason, it can be rescheduled either periodically or automatically without affecting the whole system.

V. ALGORITHM

```
# plot the confusion matrix
mat = confusion_matrix(test_data.target, predicted_categories)
sns.heatmap(mat.T,square=True,annot=True,fmt="d", xticklabels=train_data.target_names,yticklabels=train_data.target_names)
plt.xlabel("true labels")
plt.ylabel("predicted label")
plt.show()
print("The accuracy is {}".format(accuracy_score(test_data.target, predicted_categories)))
# Build the model
model = make_pipeline(TfidfVectorizer(), MultinomialNB())
# Train the model using the training data
model.fit(train_data.data, train_data.target)
# Predict the categories of the test data
predicted_categories = model.predict(test_data.data).
```

Naive Bayes:

It has commonly been used to construct textual emotion prediction and so it can be found in models made by various researchers. This trained model predicts the text is generated by a parametric model and utilizes training data to finds out Bayes-optimal estimates of the model parameters. It focuses on two Naive Bayes models: Multinomial Naive Bayes: The purpose of the model is to determine the number of times a term occurs within a document (term frequency). Since a term plays a substantial role in deciding the sentiment of a given document, Multinomial Naïve Bayes would be a good choice within the classification. Term frequency is helpful whilst deciding if a term would be useful within the analysis or not [38]. Bernoulli Naïve Bayes: Features are independent binary variables as they will indicate the presence or absence of a feature (1 and 0). The difference between Multinomial and Bernoulli is that the multinomial approach takes into consideration the term frequencies whereas the Bernoulli approach is interested in concocting whether a term is present or absent in the document under consideration.

VI. DESIGN

This section details the implementation of proposed system. The proposed system contains five models. They are given below.

1) Collecting dataset.
2) Dataset Preprocessing Module.
3) Training Dataset.
4) Dataset testing Module.
5) SVM Analysis.

6) Dataset Text Classification.
7) Analysis of Covid-19 Tweets output model.

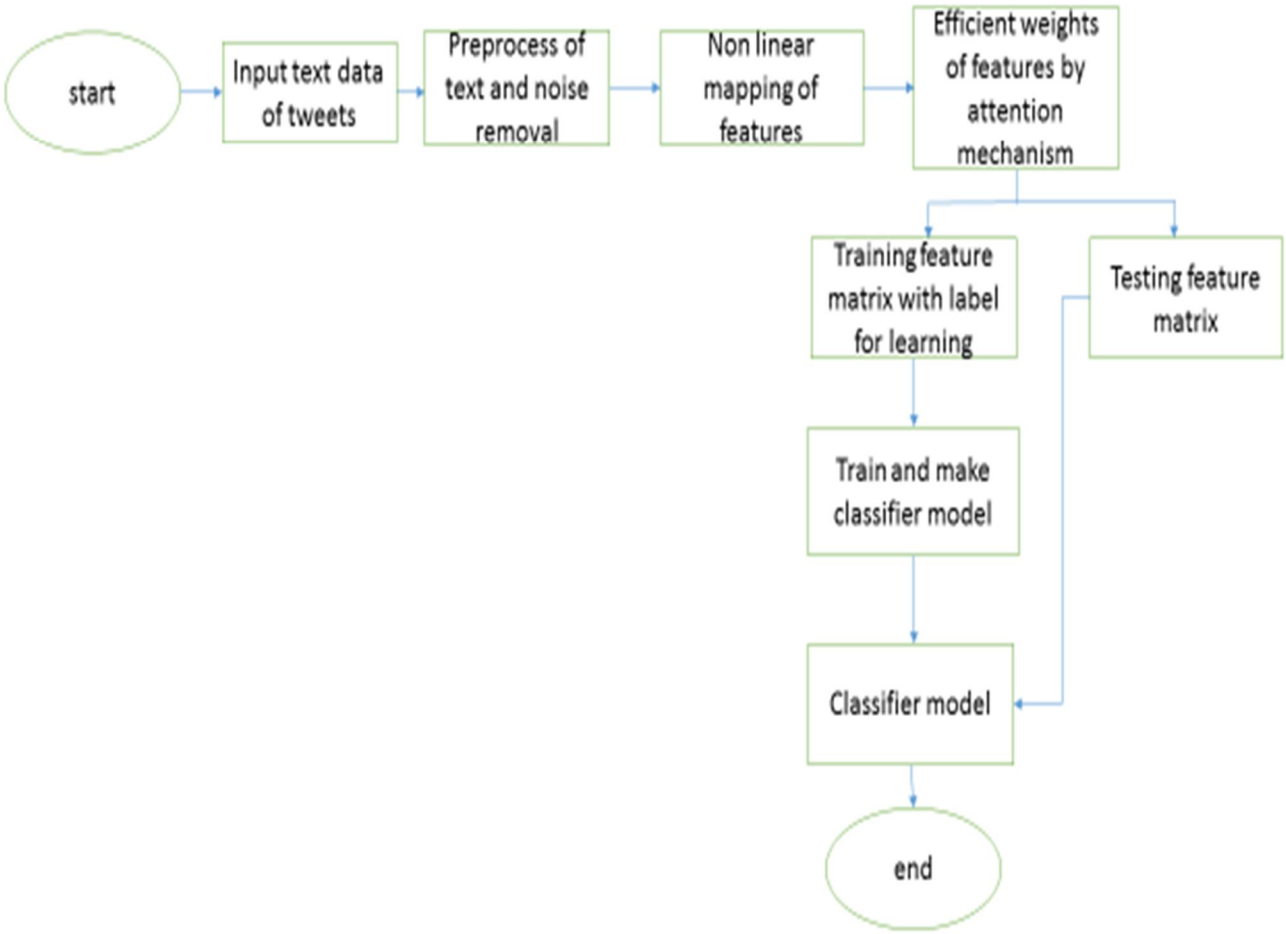
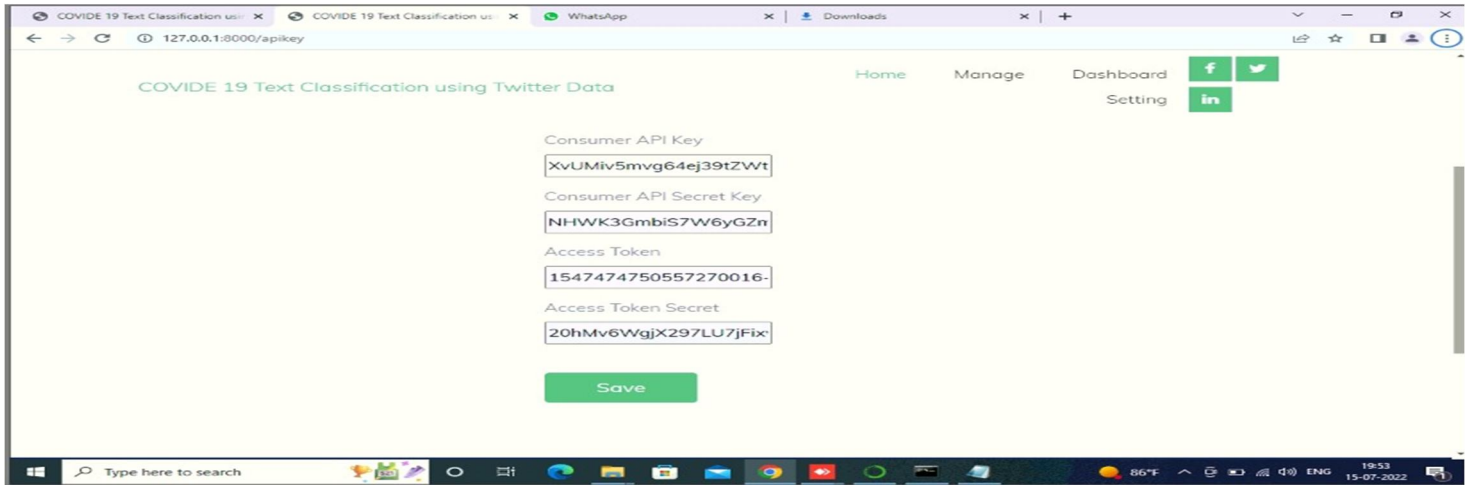
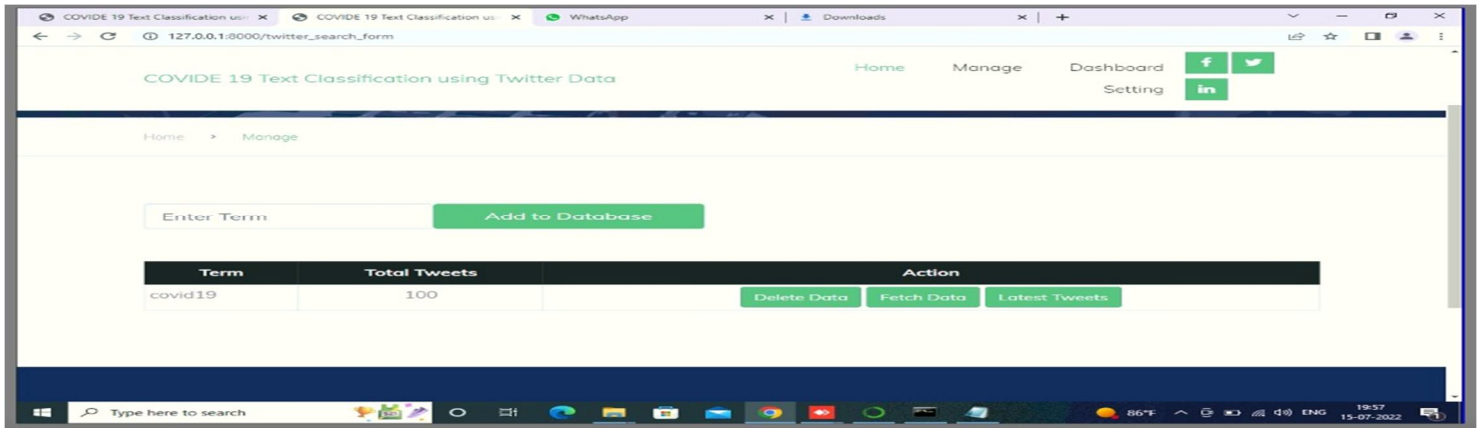


Figure VI.1: System design

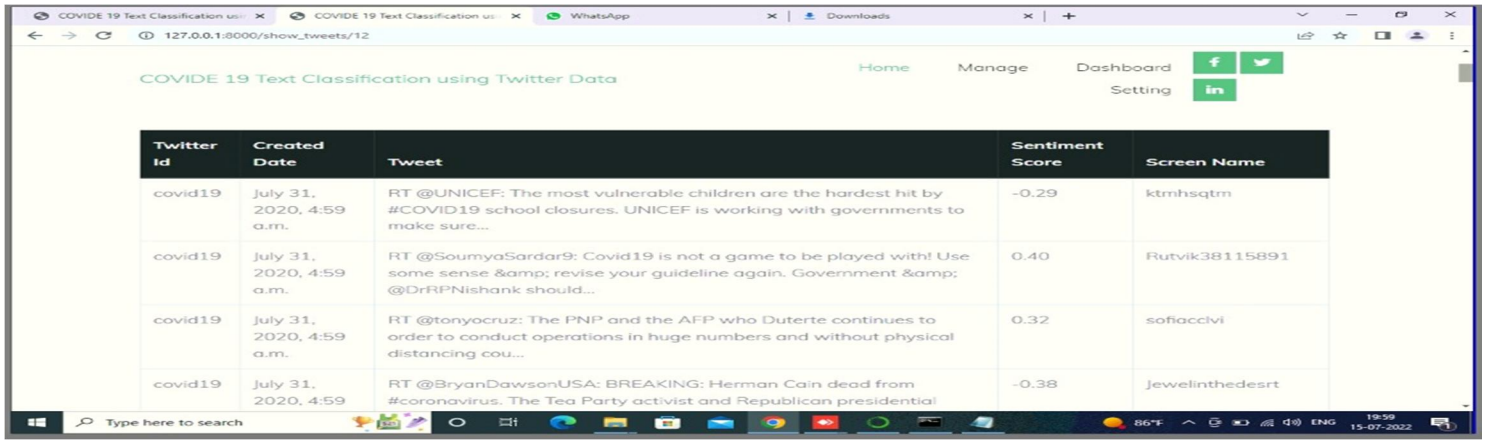
VII. RESULTS



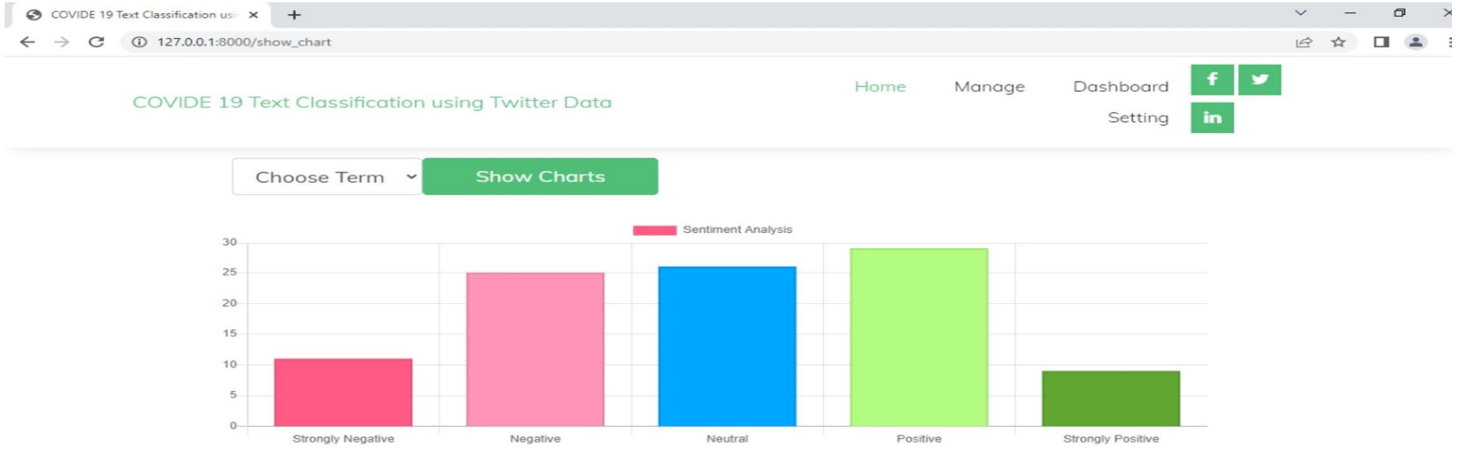
The above screen shot is used for collecting the twitter live data, here we must provide access key api, and secrete key using this we can collect the information.



The above screen shot is used where one can able to fetch the data, these keywords are the main fetures using which the retrival of the data is done.



The above screenshot is giving the latest tweets related to covid 19 information, it gives the twittered, dates, actual, tweets, then there score information and screen name used for uploading purposes.



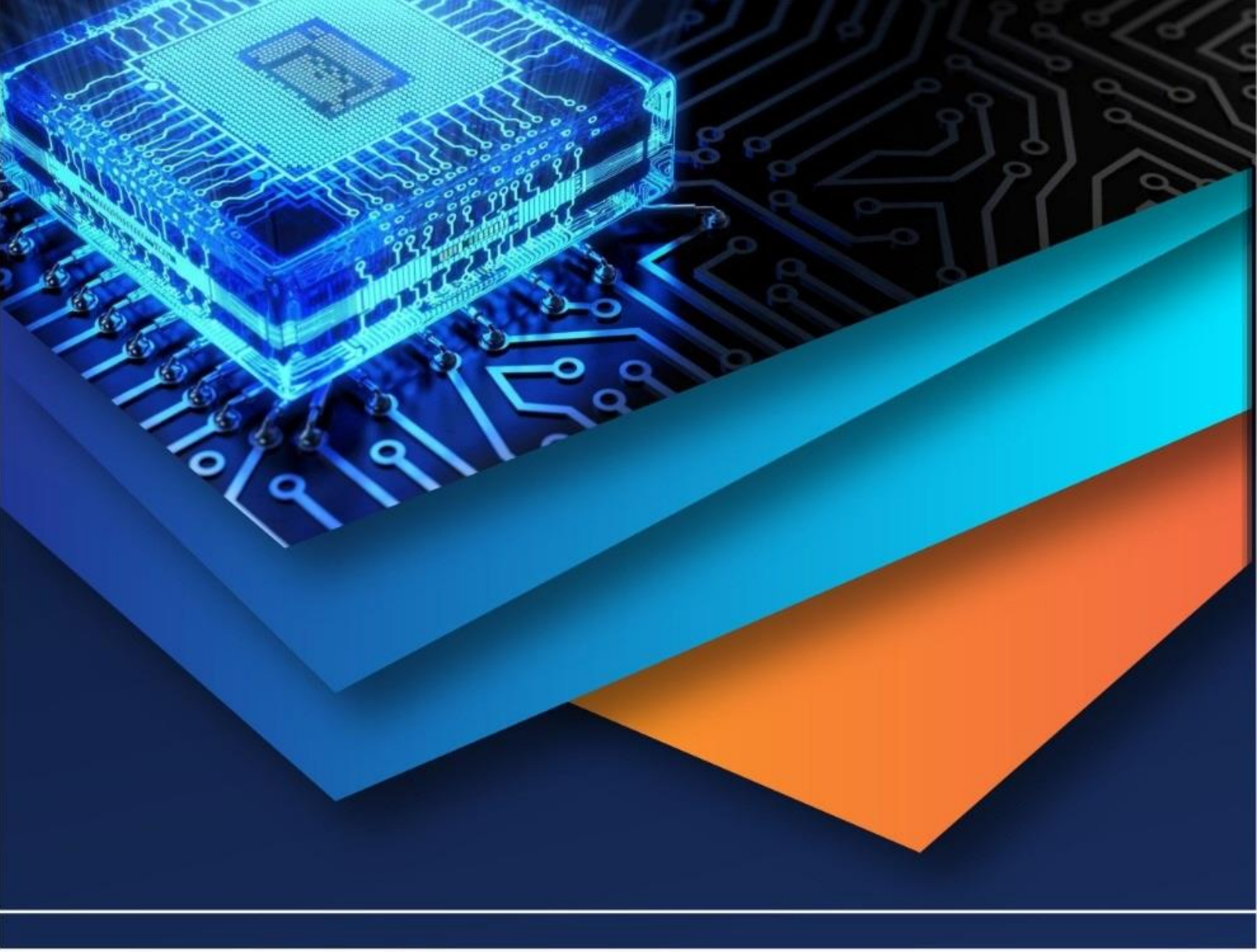
This chart shows the sentiment analysis. It is used to determine weather data is strongly negetive, negative, neutral, positive, strongly positive.

VIII. CONCLUSION

Extracting knowledge and providing meaningful insights from an extensive collection of literature remains a non-trivial task, especially when time is a constraint under circumstances such as the COVID-19 crisis. Hybrid clouds are well suited for these scenarios because of its scalable yet cost-effective nature. In this paper, we proposed a system architecture for indexing, analysing and extracting insights from scholarly articles. In particular, we conducted our experiment with the CORD-19 dataset. We choose many technologies from academia and open-source community to create an scalable, highly available, stable, high-performance and portable application. By using the Aneka PaaS solution, parallel data processing application can be effortlessly developed. It significantly reduces entry barrier for a developer to develop such a distributed application.

REFERENCES

[1] L. L. Wang, K. Lo, Y. Chandrasekhar, R. Reas, J. Yang, D. Eide, K. Funk, R. Kinney, Z. Liu, W. Merrill, P. Mooney, D. Murdick, D. Rishi, J. Sheehan, Z. Shen, B. Stilson, A. D. Wade, K. Wang,
[2] "COVID-19 Open Research Dataset Challenge (CORD-19) — Kaggle." [Online]. Available: https://www.kaggle.com/allen-institute-for-ai/ CORD-19-research-challenge/data
[3] R. N. Calheiros, C. Vecchiola, D. Karunamoorthy, and R. Buyya, "The Aneka platform and QoS-driven resource provisioning for elastic applications on hybrid Clouds," Future Generation Computer Systems, vol. 28, no. 6, pp. 861–870, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.future.2011.07.005
[4] C. Vecchiola, X. Chu, and R. Buyya, "Aneka: A Software Platform for .NET Based Cloud Computing," Advances in Parallel Computing, vol. 18, pp. 267–295, 2009.
[5] A. NadjaranToosi, R. O. Sinnott, and R. Buyya, "Resource provisioning for data-intensive applications with deadline constraints on hybrid clouds using Aneka," Future Generation Computer Systems, vol. 79, pp. 765– 775, Feb 2018.
[6] "CORD-19 Search." [Online]. Available: https://cord19.aws/{#}
[7] "COVID-Miner." [Online]. Available: https://dain.research.cchmc.org/ covidminer/
[8] "TEKStack Health - COVID-19 Research Portal." [Online]. Available: https://covid-research.tekstackhealth.com/
[9] "COVID-Miner." [Online]. Available: https://dain.research.cchmc.org/ covidminer/
[10] F. Wolinski, "Visualization of Diseases at Risk in the COVID-19 Literature," May 2020. [Online]. Available: http://arxiv.org/abs/2005. 00848
[11] "COVIDSeer." [Online]. Available: https://covidseer.ist.psu.edu/search? query=covid
[12] "COVID Explorer." [Online]. Available: https://coronavirus-ai.psu.edu/ database
[13] J. McCaffrey, "ML.NET: The Machine Learning Framework for .NET Developers." MSDN magazine, no. 13, p. 24, 2018.
[14] "MinIO — High Performance, Kubernetes Native Object Storage." [Online]. Available: https://min.io/
[15] "Elastic Stack: Elasticsearch, Kibana, Beats &Logstash — Elastic." [Online]. Available: https://www.elastic.co/elastic-stack
[16] T.-H. K. Huang, C.-Y. Huang, C.-K.C. Ding, Y.-C. Hsu, and C. L. Giles, "CODA-19: Reliably Annotating Research Aspects on 10,000+ CORD-19 Abstracts Using a Non-Expert Crowd," May 2020. [Online]. Available: http://arxiv.org/abs/2005.02367.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY