NEWS ARTICLE CREDIBILITY TESTER

A PROJECT REPORT
BY

SUBMITTED TO
Dr. Nitin Arvind Shelke

SUBMITTED BY

Arman Beer Kaur	401883004
Vedansh Varshney	401703030
Vybhav Chaturvedi	401703032

LM THAPAR SCHOOL OF MANAGEMENT, DERA BASSI CAMPUS, DISTRICT MOHALI- 140507

(AUG-DEC 2021)

DECLARATION

We hereby declare that the work which is being presented in the report entitled “News Article Credibility Tester” under the subject Social Media and Web Analytics, is an authentic record of our own work carried out during the period from JUNE, 2021 to December, 2021 at LM Thapar School of Management.

The matters and the results presented in this report has not been submitted by us for the award of any other degree elsewhere.

TABLE OF CONTENTS

ABSTRACT

Headlines are the marquees of news stories. These short statements at the beginning of an article frame the information that is to follow and encapsulate a portion of the story, presenting a snapshot of reality. For many outlets navigating the changes associated with digital journalism, news headlines have changed as well. In addition to traditional headlines that highlight the important aspects of the story, headlines are being crafted to arouse curiosity, pose intriguing questions, or lead to specific conclusions. Industry insiders refer to non-traditional headlines that attempt to entice readers to a news page as "Clickbait." The question with these new headline formats is whether the way a title is written has an impact on how readers perceive the news source. In this research, we utilize ensemble modelling in an attempt to assess the reliability of the headlines by determining whether the content matches the headline or is simply outlandish to garner clicks.

1 INTRODUCTION

With the onset of the pandemic, followed by emerging protests, economic distress, and worry among the world's population, a recently released survey suggests that consumers are shifting away from traditional media sources for their news and are increasingly turning to social media and messaging services. With our rapid advancement into the digitalized era, the usage of traditional sources of news are observing a substantial decay and people are turning towards the Internet in order to get their news related information and updates.

1.1. Problem Statement

So far, we have relied on media/news outlets on their face value. However, having a high face value does not always imply accurate and dependable information. Many media outlets, even those with millions of followers or viewers on various social media platforms, can nevertheless produce inaccurate or redundant content in order to get public attention or feature in the latest trends. Our project will assist social media users in overcoming the misconception that "a large following on the Internet always implies a trustworthy source” and will help them in identifying and exploring news stories with more accurate and relevant information/content.

1.2. Motivation

This project was chosen by our team because we believe it is unique and can serve an important purpose. It helps in reducing the likelihood of consumers being "click-baited," as numerous media outlets frequently employ false titles to garner attention. People who are aware of the proportion of relevant information they are likely to find in a news article before clicking on it can make better choices when deciding what to read and so save time. Our project is also useful for screening out spam or misleading items, and it can assist separate trustworthy media outlets that consistently deliver clear, accurate, and comprehensive information and news from those that publish misleading pieces. Media outlets can utilize our project as a beneficial tool to demonstrate their dependability and authenticity to their readers to build trust with them. It can also be used to select or curate articles based on their informativeness and relevance to the user.

2. BACKGROUND RESEARCH

With this initiative, we hope to help users in determining which news headlines are most credible in encapsulating the story contained with an article. There was no equivalent example that we could find that suited our notion, despite the fact that there are numerous applications based on the feasibility of news and headlines. Fake news detection, spam marketing disguised as news, and sentiment analysis of news items are some of the applications or research work that resonated with the topic we are working with. Though the research work surrounding these topics provides the user with a general understanding of news-related trends and observations, none of this checks the reliability of news headlines in relation to news content, which is what we aim to accomplish.

2.1 Related Work

Table 1 Literature Survey

Author	Year	Dataset	Algorithms	Research Approach	Results
[14]	2020	Stanford Sentiment Treebank dataset and IMDB dataset	SentiVec	Deep Learning	85% (Best Case)
[7]	2018	News dataset	Conventional ML and CNN	Both	85%
[1]	2019	Turkish news text dataset	SVM and RNN	Both	98% (Best Case)
[13]	2018	StockTwits dataset	CNN, LSTM	Deep Learning	90.08% (Best Case)
[10]	2017	German SentiWordNet lexicon	RNN	Deep Learning	0.7494b F Score (Best Case)
[6]	2020	COVID-19 Twitter tweets dataset	LSTM	Deep Learning	81.8%
[8]	2018	6 benchmark datasets	Bidirectional LSTM and RNN	Deep Learning	-
[3]	2020	Social media reviews dataset	ConvLSTM Conv	Deep Learning	89.02%

[11]	2019	IMDB dataset & Restaurant Reviews	DNN and LSTM	Deep Learning	78-88%
[15]	2018	Song lyrics, IMDB movie reviews and Amazon Reviews	NgramCNN	Deep Learning	91.2 %
[16]	2020	Twitter tweets dataset	CNN and Graph Structures	Deep Learning	79.8% (Average Accuracy)
[4]	2017	IMDB dataset and Stanford Sentiment Treebank Dataset	ConvLSTM model	Deep Learning	88.3% (Conv LSTM)
[9]	2020	Online Website reviews Dataset	CNN, RNN and LSTM	Deep Learning	-
[12]	2016	Twitter tweets dataset	Naïve Bayes	Machine Learning	-
[2]	2019	TripAdvisor reviews dataset	VADER	Data Visulzation	-

Many researchers have contributed in news sentiment analysis using different approaches. A brief discussion on the work done previously on sentiment analysis is provided in this section.

Zhu et al [14] employed deep learning using the SentiVec algorithm on the Stanford Sentiment Treebank dataset and IMDB dataset in 2020 and obtained an accuracy of 85% for their best-case scenario. Kale et al [7] utilized both machine learning and deep learning using conventional ML and CNN algorithms on the News dataset achieving an accuracy of nearly 85% in 2018. Abbas et al [1] also made use of both approaches using the SVM and RNN algorithms on the Turkish news text dataset and achieved an impressive accuracy of 98% in 2019. Imran et al [6] utilized deep learning using the LSTM algorithm on the COVID-19 Twitter tweets dataset in 2020 and obtained an accuracy of nearly 81.8%. In 2018, Cano et al [15] employed deep learning using the NgramCNN algorithm on a variety of input data like song lyrics, IMDB movie reviews, and Amazon reviews and obtained an accuracy of 91.2%. In 2016, Qaisi et al [12] utilized deep learning approach using the Naive Bayes algorithm on Twitter tweets dataset. Aguero Torales et al [2] employed Data Visualization while using the VADER algorithm on a dataset containing reviews on TripAdvisor in 2019.

2.2 Goals and Objectives

We aim to accomplish the following objectives with the help of our project:

1. Exploring and identifying news articles that contain more accurate and relevant information with reference to their headlines.

2. Reducing the likelihood of users getting “click-baited” by articles with bizarre headlines that have been crafted purely for the purpose of garnering views and public attention.

3. PROPOSED METHODOLOGY

This section describes in detail about the design part of the system.

3.1. Overall Description

This section has discussed the proposed model for determining the credibility of news headline with the title. First of all, we will discuss the Dataset, then the discussion will move to Pre-processing and vectorization. Finally, the developed Models would be discussed.

3.2. System Architecture

3.2.1 Block Diagram

Block Diagram is a diagram of a system in which the principal parts or functions are represented by blocks connected by lines that show the relationships between the blocks. In this diagram, we show the process of classifying news source feasibility of the news article accessed by the user.

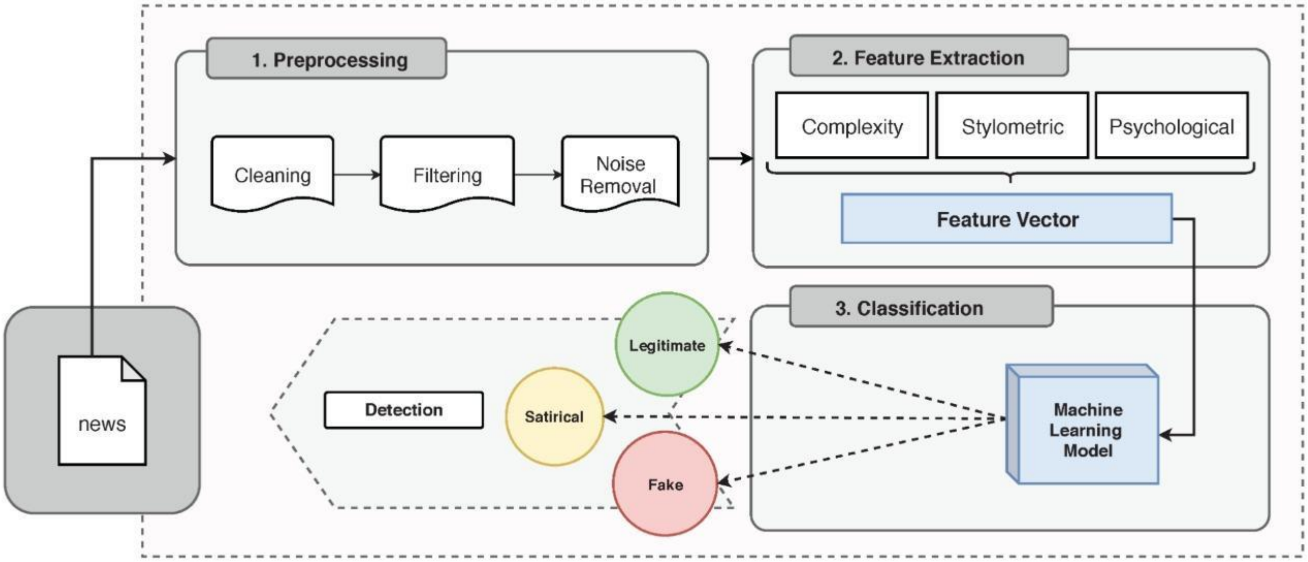

Figure 1: Block Diagram

3.2.2 Use Case Diagram

Use Case Diagram shows some of the use cases in the system, some of the actors in the system, and the relationships between them. Our diagram has only one actor which is the user and the only person who accesses the system and obtains the results in the end from it.

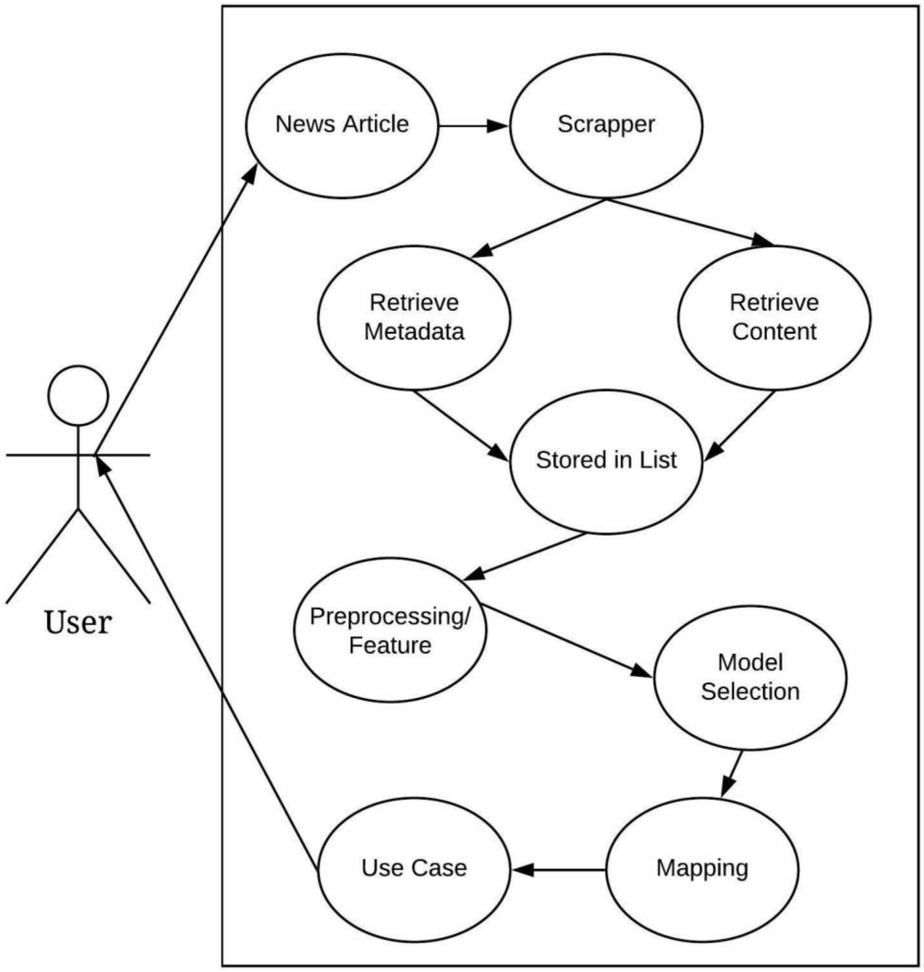

Figure 2: Use Case Diagram

3.2.3 Level 0 Data Flow Diagram

Level 0 data flow diagrams show a single process node and its connections to external entities. User and News article are the two entities here that are connected with our news liability process.

Figure 3: Level 0 DFD

3.2.4 Level 1 Data Flow Diagram

Level 1 DFDs describe more detail than a context diagram as new processes are added, the diagram needs additional data flows and data stores to link them together. We expand our single process node of Level 0 DFD further into more processes in this diagram.

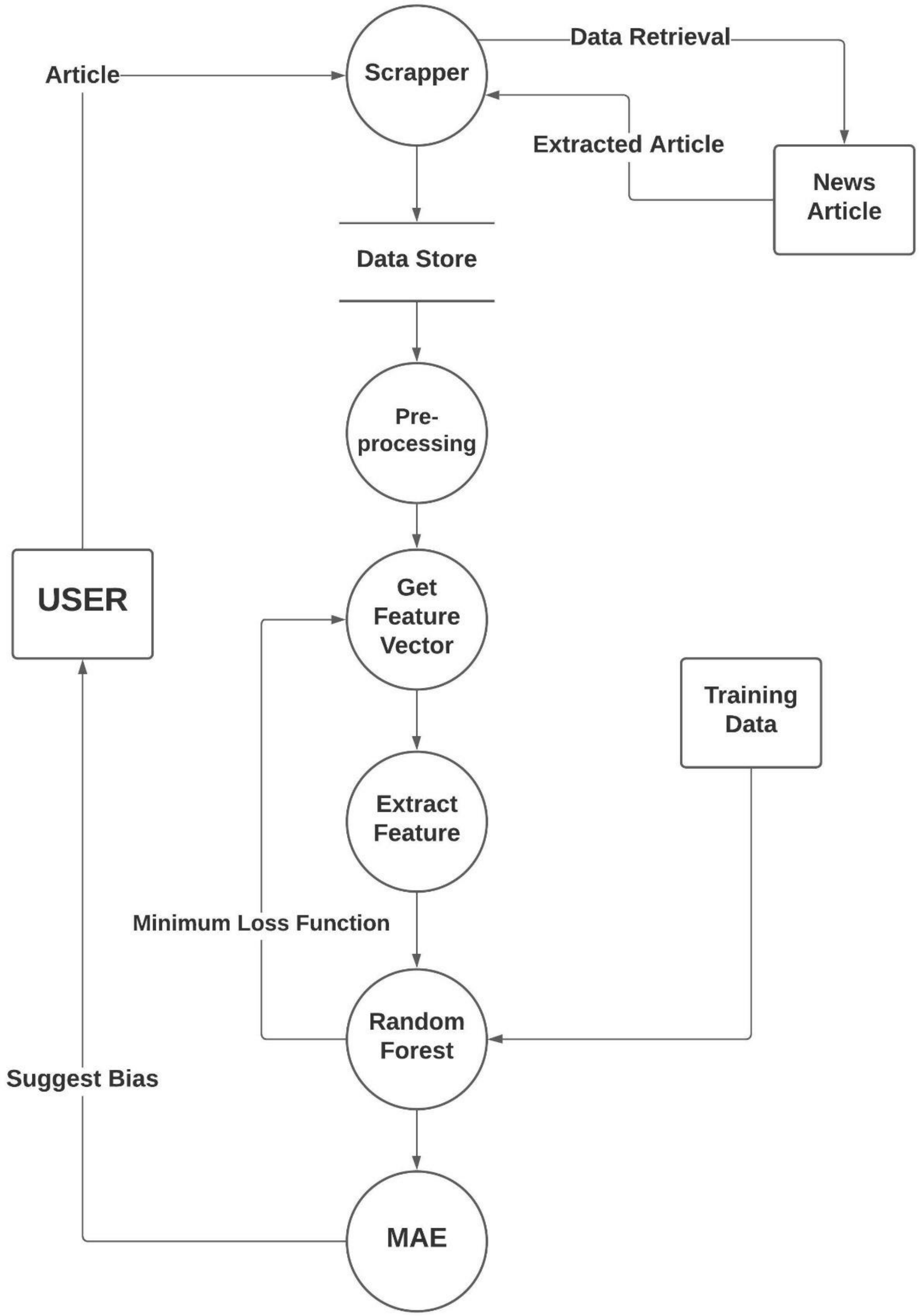

Figure 4: Level 1 DFD

3.2.5 Pipeline Diagram

Pipeline diagram shows the series of instructions divided into its component stages. In this diagram, there are three main components namely Pre-processing, Feature extraction and Classification described individually giving the final result.

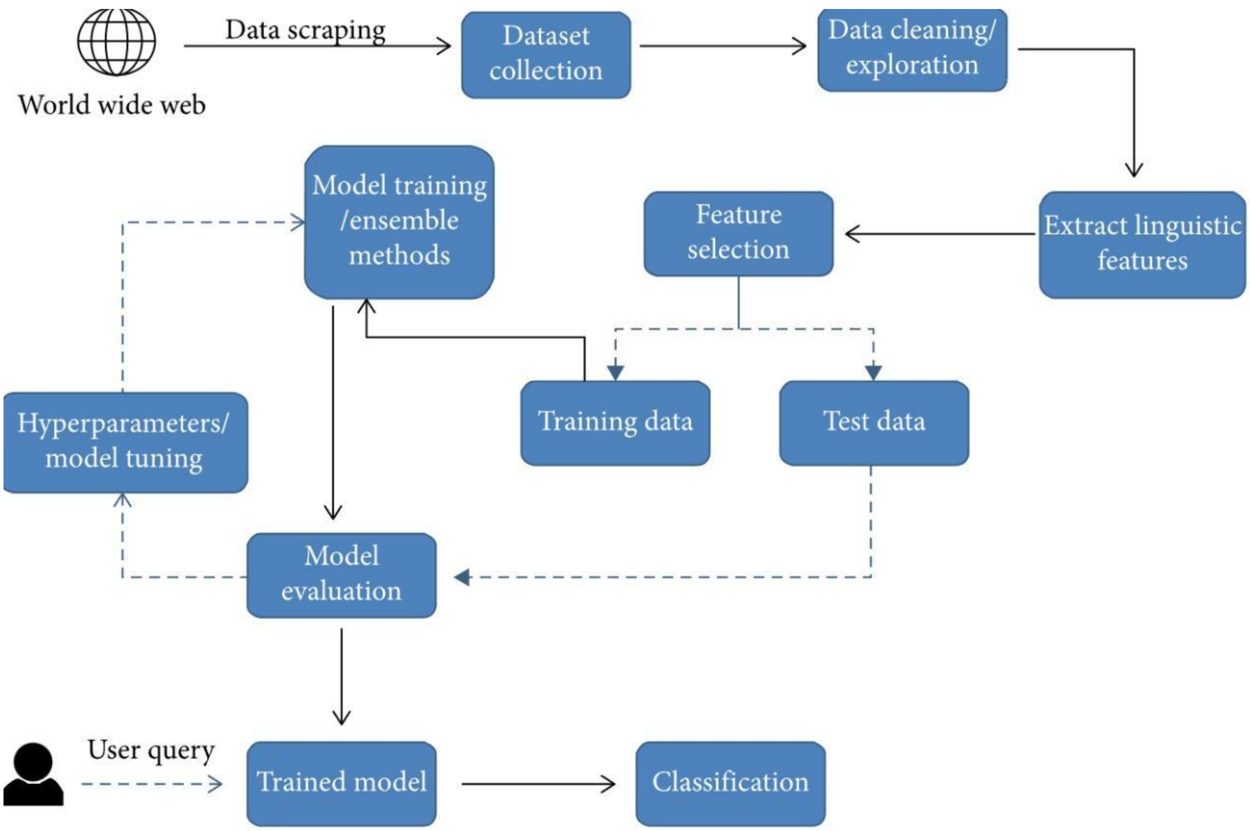

Figure 5: Pipeline Diagram

3.3 Dataset Used

The Dataset used for the study consists of 48000+ News Articles from reputed publication including US Today, ABC News, The Guardian, Bloomberg on multiple topics such as Economy, Politics, General Election, Technology and International relations.

```
In [4]: train.head()
```

Out[4]:

	IDLink	Title	Headline	Source	Topic	PublishDate	Facebook	GooglePlus	LinkedIn	SentimentTitle	SentimentHeadline
0	Tr3CMgRv1N	Obama Lays Wreath at Arlington National Cemetery	Obama Lays Wreath at Arlington National Cemete...	USA TODAY	obama	2002-04-02 00:00:00	-1	-1	-1	0.000000	-0.053300
1	Wc81vGp8qZ	A Look at the Health of the Chinese Economy	Tim Haywood, investment director business-unit...	Bloomberg	economy	2008-09-20 00:00:00	-1	-1	-1	0.208333	-0.156386
2	zNGH03CrZH	Nouriel Roubini: Global Economy Not Back to 2008	Nouriel Roubini, NYU professor and chairman at...	Bloomberg	economy	2012-01-28 00:00:00	-1	-1	-1	-0.425210	0.139754
3	3sM1H0W8ts	Finland GDP Expands In Q4	Finland's economy expanded marginally in the t...	RTT News	economy	2015-03-01 00:06:00	-1	-1	-1	0.000000	0.026064
4	wUbnxgvqaZ	Tourism, govt spending buoys Thai economy in J...	Tourism and public spending continued to boost...	The Nation - Thailand's English news	economy	2015-03-01 00:11:00	-1	-1	-1	0.000000	0.141084

Figure 6: Dataset Description

Additionally, the data comes labeled with the associated topics and the Sentimental value of the Title and the Headline. The Dataset consists of total 11 columns:

```
Data columns (total 11 columns):
#    Column             Non-Null Count  Dtype
---  ------             --------------  -----
0    IDLink             55932 non-null  object
1    Title              55932 non-null  object
2    Headline           55932 non-null  object
3    Source             55757 non-null  object
4    Topic              55932 non-null  object
5    PublishDate        55932 non-null  object
6    Facebook           55932 non-null  int64
7    GooglePlus         55932 non-null  int64
8    LinkedIn           55932 non-null  int64
9    SentimentTitle     55932 non-null  float64
10   SentimentHeadline  55932 non-null  float64
```

This study focuses on the writing of newspaper front-page headlines for some of the most most popular daily English newspaper. The texts were collected for two years, from 1 January 2002 to 31 December 2019, from various newspapers. There are no articles related to COVID-19 pandemic as it would have skewed the results in one direction. Then, the data were pre-processed and analyzed using several Python-packages for text mining, as detailed in the following section.

3.4 Data Preprocessing

Data Preprocessing or Text preprocessing is a method to clean the text data and make it ready to feed data to the model. Text data contains noise in various forms like emotions, punctuation, text in a different case. The most prominent methods of text mining are word cloud and sentiment analysis. The Word Cloud from our data has been attached below.

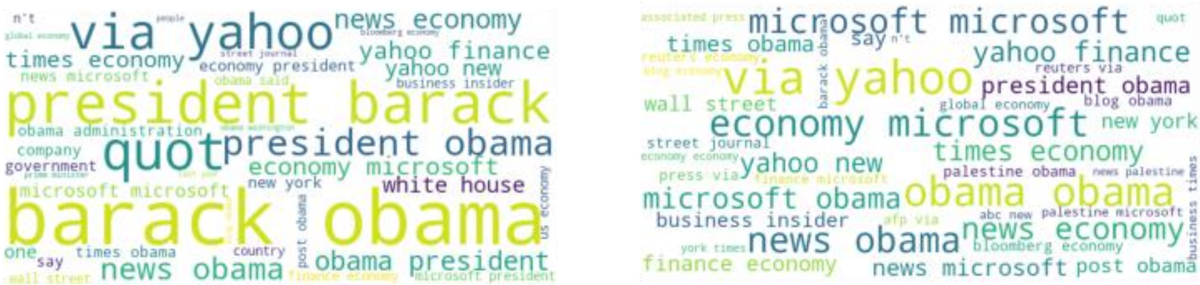

Figure 7: Word Clouds

Text pre-processing, in essence, acts on raw text to eliminate extraneous information. The pre-processing step includes tokenization, standardization, cleaning, and stop word removal, as well as stemming or lemmatization. The method is crucial in acquiring and simplifying information. Prepositions, URLs, numerals, conjunctions, and other unnecessary items that have no relevance on the phrase are often deleted. First of all, initial preprocessing is done, Words are converted into lower case and URLs and symbols are removed. After those words are tokenized, the text is split into smaller units. We used word tokenization based on our problem statement. After Tokenization, Stop Words are removed they are the commonly used words and are removed from the text as they do not add any value to the analysis. These words carry less or no meaning. It is also known as the text standardization step where the words are stemmed or diminished to their root/base form. For example, words like 'programmer', 'programming, 'program' will be stemmed to 'program'. However, the problem of stemming is that it stems the words in such a way that the root form loses meaning or is not reduced to a valid English term. It stems the term yet ensures that it retains its meaning. Lemmatization has a pre-defined vocabulary that saves the context of words and verifies the word in the dictionary as it decreases.

After preprocessing the list of words are vectorized, Word Embeddings or Word vectorization is a methodology in NLP to map words or phrases from vocabulary to a corresponding vector of real numbers which used to find word predictions, word similarities/semantics.

When a human peruses a text, sentiment analysis assists in analyzing the human emotion in order to determine its polarity. Polarity is float which lies in the range of [-1,1] where 1 means positive statement and -1 means a negative statement. while Subjective generally refer to personal opinion, emotion or judgment whereas objective refers to factual information.

Raw text cannot be passed into machines as input until and unless they are converted into numbers, hence encoding is performed. Text encoding is the process of converting meaningful text into a numeric or vector representation while preserving the context and relationship between words and sentences, so that a computer can grasp the pattern associated with any text and determine the context of sentences. The dates and days were also encoded in form of numbers.

Finally, after preprocessing, normalization using was performed and features were standardized by removing the mean and scaling to unit variance.

3.5 Model Development

After preprocessing and normalization, data was split in 80:20 ratio for testing and training. We used the following classifiers.

- Ridge
- Lasso
- Elastic Net
- Decision Trees
- k-Neighbors
- Linear SVM

We used Sci-kit learn, SciPy and Statsmodel for the modeling and fitting. The results obtained from the regression have been discussed in next section.

3.5.1 Ensemble modeling

For a particular dataset, a single algorithm may not generate the best forecast. Machine learning algorithms have constraints and creating a model with high accuracy is difficult. If we create and merge numerous models, we may be able to improve overall accuracy. The combination may be achieved by aggregating the output of each model with two goals in mind: lowering model error while retaining generalization.

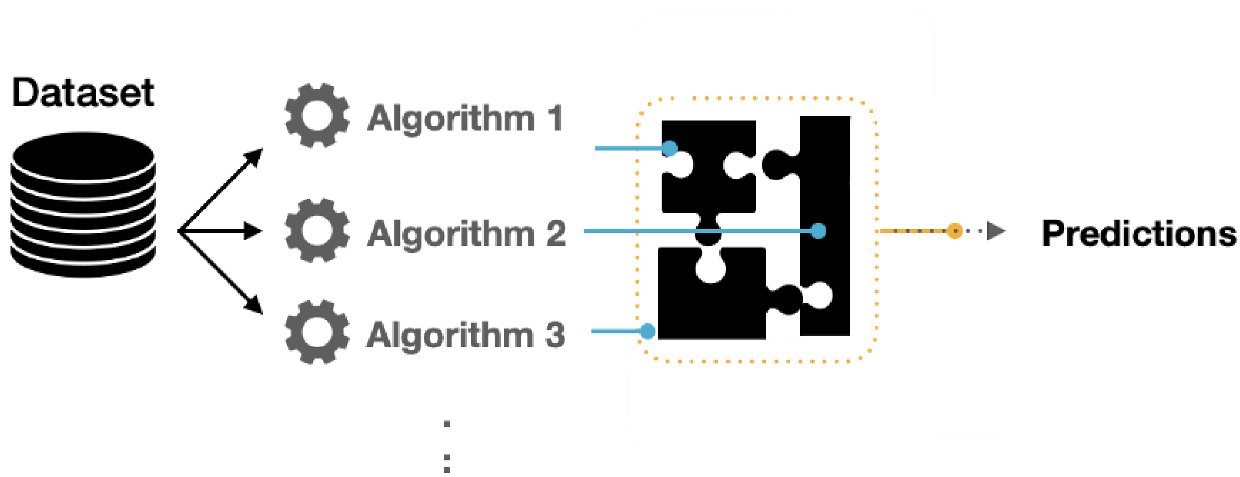

Figure 8: Ensemble Modeling

There are 4 major approaches:

- Bagging
- Boosting
- Stacking
- Voting

We initially used 5 ensemble models:

- Random Forest Bagging variation
- Bagging Regression
- Cat Regression
- Max-Voting Regression
- Average-Stacking regression

When creating ensemble models, the variance of the approach used is not the only factor to consider. For example, we may train a large number of C45 models, each of which learns a unique pattern specialized in predicting a single component. Weak learners are these models, and they may be used to construct a meta-model. In an ensemble learning architecture, the inputs are distributed to each weak learner while their predictions are gathered. An ensemble model may be built using the combined forecast. We are not dealing with yes or no questions in regression problems. We need to determine the best numerical numbers that can be anticipated. We may take an average of the forecasts that have been collected. Ensemble models is an excellent method for machine learning. The ensemble models have a variety of techniques for classification and regression problems. We have discovered the types of such models, how we can build a simple ensemble model, and how they boost the model accuracy. Cat Regression and Bagging Algorithm took a lot of computation time and hence we decided to interrupt the model development.

4. EXPERIMENTAL RESULTS AND DISCUSSION

```
linearModel2=sm.GLS(Yvar,Xvar).fit()
print(linearModel2.summary())
```

```
                            GLS Regression Results
=======================================================================================
Dep. Variable:                 Source   R-squared (uncentered):                   0.000
Model:                            GLS   Adj. R-squared (uncentered):              0.000
Method:                 Least Squares   F-statistic:                              5.568
Date:                Thu, 23 Dec 2021   Prob (F-statistic):                     0.00382
Time:                        18:26:41   Log-Likelihood:                         -79359.
No. Observations:               55932   AIC:                                  1.587e+05
Df Residuals:                   55930   BIC:                                  1.587e+05
Df Model:                           2
Covariance Type:            nonrobust
=====================================================================================
                        coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------
SentimentTitle        0.0361      0.031      1.155      0.248      -0.025       0.097
SentimentHeadline     0.0841      0.029      2.853      0.004       0.026       0.142
==============================================================================
Omnibus:                  1632657.246   Durbin-Watson:                   1.928
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             4200.774
Skew:                          -0.027   Prob(JB):                         0.00
Kurtosis:                       1.659   Cond. No.                         1.22
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

```python
linearModel1=sm.OLS(Yvar,Xvar).fit()
print(linearModel1.summary())
```

```
                            OLS Regression Results
=======================================================================================
Dep. Variable:                 Source   R-squared (uncentered):                   0.000
Model:                            OLS   Adj. R-squared (uncentered):              0.000
Method:                 Least Squares   F-statistic:                              5.568
Date:                Thu, 23 Dec 2021   Prob (F-statistic):                     0.00382
Time:                        18:48:39   Log-Likelihood:                         -79359.
No. Observations:               55932   AIC:                                  1.587e+05
Df Residuals:                   55930   BIC:                                  1.587e+05
Df Model:                           2
Covariance Type:            nonrobust
=====================================================================================
                        coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------
SentimentTitle        0.0361      0.031      1.155      0.248      -0.025       0.097
SentimentHeadline     0.0841      0.029      2.853      0.004       0.026       0.142
==============================================================================
Omnibus:                  1632657.246   Durbin-Watson:                   1.928
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             4200.774
Skew:                          -0.027   Prob(JB):                         0.00
Kurtosis:                       1.659   Cond. No.                         1.22
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

The following parameters have been evaluated for determining the best regressors for our case:

- **Explained Variance Score:** The explained variance score explains the dispersion of errors of a given dataset. The best possible score is 1.0, values lower than 1 are considered to be undesirable.

- **Max Error:** The Max-Error metric is the worst-case error between the predicted value and the true value. Lower the value, the more desirable it is.

- **Mean Absolute Error:** The mean absolute error of a model with respect to a test set is the mean of the absolute values of the individual prediction errors on over all instances in the test set. Lower the value, the more desirable is the value.

- **Mean Squared Error:** tells you how close a regression line is to a set of points. Lower the value of the error, the more closely are the points located with respect to the regression line.

- **Median Absolute Error:** the median of all of the absolute values of the residuals. the median absolute error is useful as it is essentially insensitive to outliers. Lower the value, the more desirable is the value.

- **R-squared Score:** R-squared (R^2) is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model. your R-squared should not be greater than the amount of variability.

Ridge regressor was the best performing regressor and after ensembling, stack regressor provided the best results because of minimum overfitting.

Table 2 Metrics and Scores of Algorithms

	RANDOM FOREST REGRESSION	ELASTIC NETS	RIDGE REGRESSION	LASSO REGRESSION	DECISION TREES REGRESSION
Explained Variance Score:	0.3754663	0.0000000	0.5332692	0.0000000	-0.1049028
Max Error:	0.6453006	0.7695916	0.7202093	0.7695916	0.8780334
Mean Absolute Error:	0.0837758	0.1093500	0.0719510	0.1093500	0.1097854
Mean Squared Error:	0.0126583	0.0202696	0.0094588	0.0202696	0.0223944
Median Absolute Error:	0.0637260	0.0867242	0.0544368	0.0867242	0.0841602
R^2 score:	0.3753906	-0.0001824	0.5332678	-0.0001824	-0.1050257
	k-NEIGHBOURS REGRESSION	**GRADIENT BOOSITNG REGRESSION**	**STACKING REGRESSION**	**VOTING REGRESSION**	**LINEAR SVR**
Explained Variance Score:	-0.2911120	0.2062728	0.5237221	0.3786776	0.3754663
Max Error:	0.8237983	0.6616571	0.7580094	0.5788553	0.6453006
Mean Absolute Error:	0.1238655	0.0974502	0.0721316	0.0850451	0.0837758
Mean Squared Error:	0.0261658	0.0160914	0.0096532	0.0125917	0.0126583
Median Absolute Error:	0.0991882	0.0787630	0.0532696	0.0666789	0.0637260
R^2 score:	-0.2911214	0.2059887	0.5236720	0.3786776	0.3753906

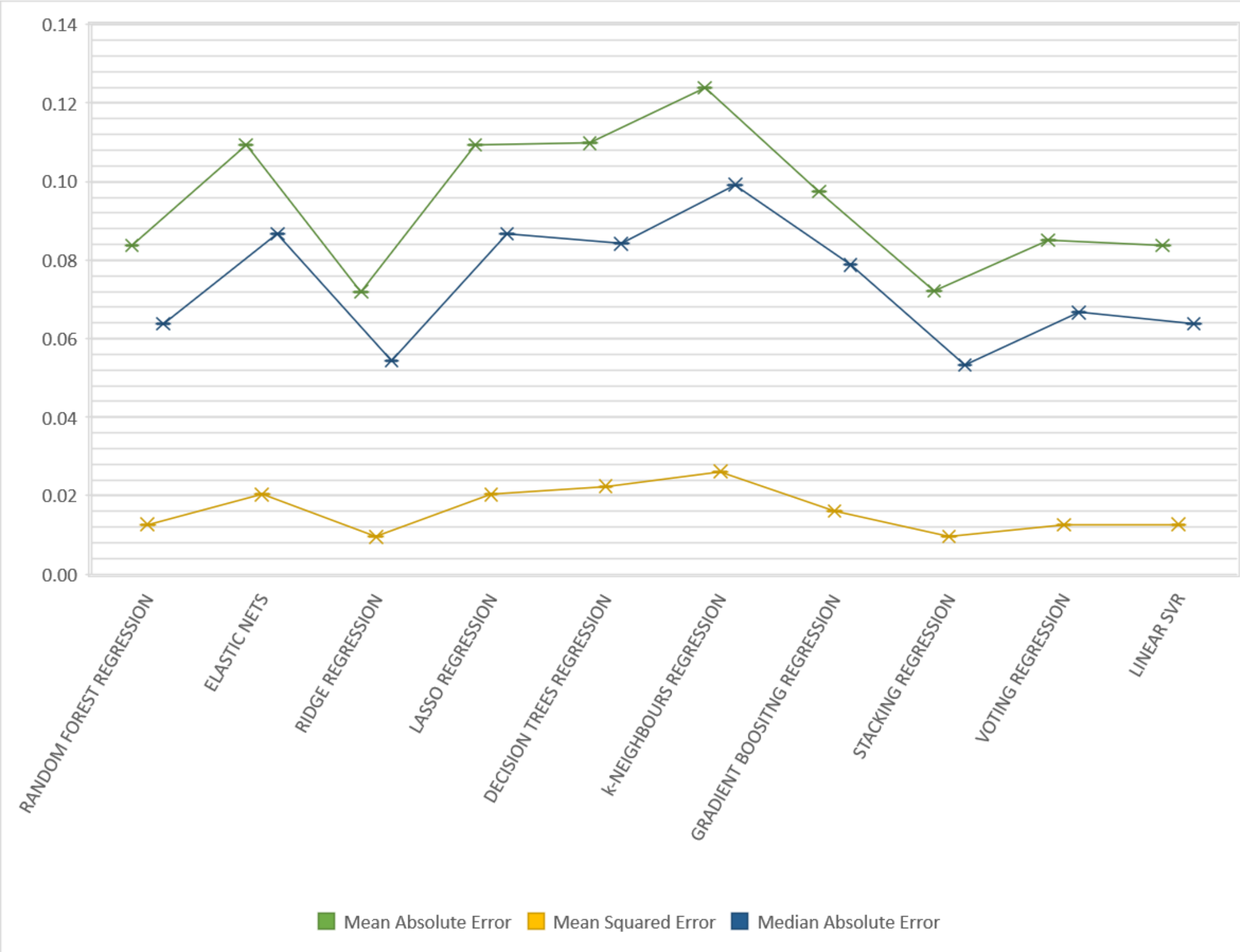

Figure 9: Error Scores vs Regression

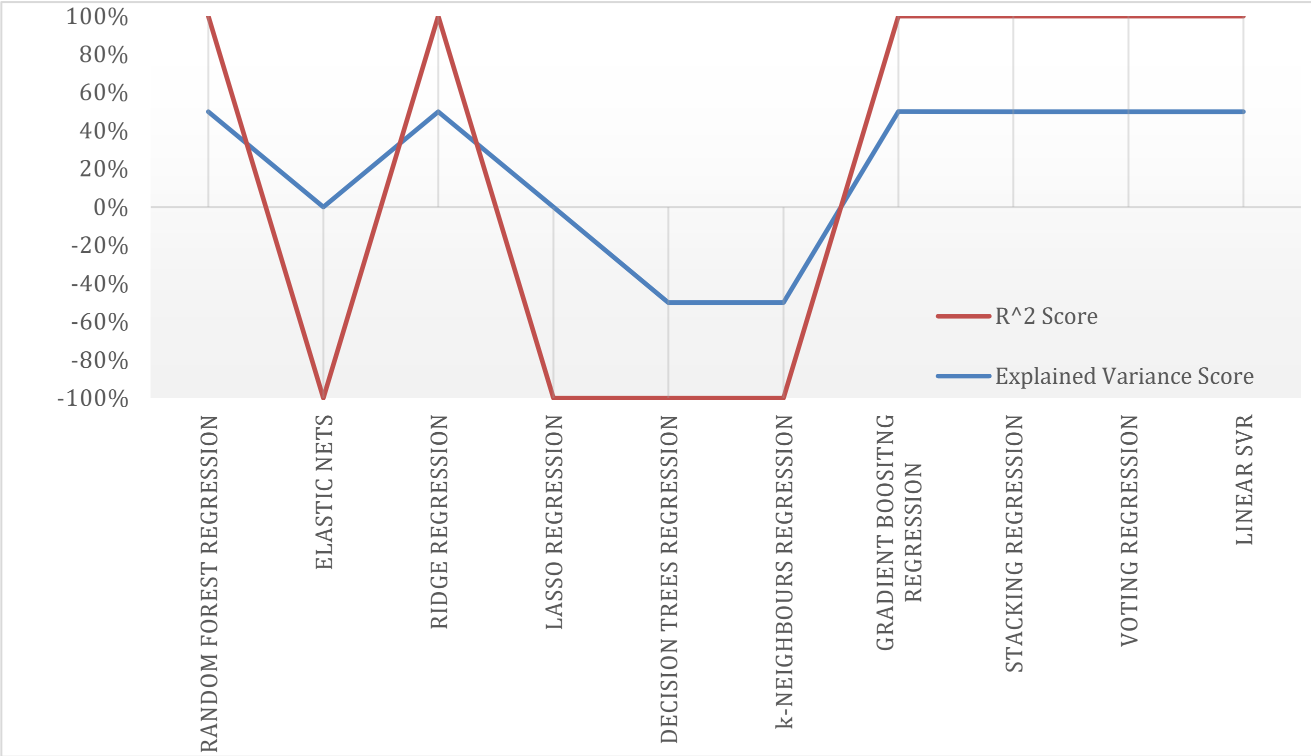

Figure 10: Regression Metrics vs Regression

These two graphs are the graphical representation of Table II and represent the score values for different parameters with respect to the various regressors used.

5. CONCLUSION AND FUTURE WORKS

Text analytics is a rapidly expanding topic that tries to enhance people's views of many situations by discovering key hidden information and patterns in text. The fast development of natural language processing models and implementations enables real-time information prediction, categorization, and identification. The texts were preprocessed using one of the many python preprocessing and text mining programs. However, among the approaches utilized, the word cloud methodology was used to find the most informative word, revealing that phrases connected to Election, Politics, International Relations, and Economy occurred the most frequently in 2018. After performing regression and setting up an ensemble model, we determined the 2 best algorithm for determining the sentiment of News Articles and their corresponding Headlines. By using text blob, spacy, and scikit learn, data analysis and model development was conducted. The algorithms determined not only give better results than the other alternatives but also reduce the time required for processing. To increase the level of predictability, however, as future work, we need to include different approaches and combinations of methodologies. Moreover, data scraping should be used for acquiring articles in various languages and a rigorous ensemble modeling approach should be used for better possible results. An application or extension can be also deployed for end users.

REFERENCES

[1] Abbas, S.S.I., Automatic Text Categorization of Turkish News with Machine Learning and Deep Learning Techniques. 2019, Ankara Yıldırım Beyazıt Üniversitesi Fen Bilimleri Enstitüsü.

[2] Agüero-Torales, M., et al., A cloud-based tool for sentiment analysis in reviews about restaurants on TripAdvisor. 2019. 162: p. 392-399.

[3] Ghorbani, M., et al., ConvLSTMConv network: a deep learning approach for sentiment analysis in cloud computing. 2020. 9(1): p. 1-12.

[4] Hassan, A. and A. Mahmood. Deep learning approach for sentiment analysis of short texts. in 2017 3rd international conference on control, automation and robotics (ICCAR). 2017. IEEE.

[5] Hussain, W., Sohaib, O., Naderpour, M., et al., Cloud MarginalResource Allocation: A Decision Support Model. Mobile NetworksApplications. 2019. 25: p. 1418-1433.

[6] Imran, A.S., et al., Cross-cultural polarity and emotion detection usingsentiment analysis and deep learning on COVID-19 related tweets. 2020. 8: p. 181074-181090.

[7] Kale, M., P. Mankame, and G.J.a.p.a. Kulkarni, Deep learning for digital text analytics: Sentiment analysis. 2018.

[8] Kratzwald, B., et al., Deep learning for affective computing: Text-based emotion recognition in decision support. 2018. 115: p. 24-35.

[9] Li, L., et al., How textual quality of online reviews affect classification performance: a case of deep learning sentiment analysis. 2020. 32(9): p.4387-4415.

[10] Naderalvojoud, B., B. Qasemizadeh, and L.J.P.o.t.G. Kallmeyer, HU-HHU at GermEval-2017 Sub-task B: Lexicon-Based Deep Learning forContextual Sentiment Analysis. 2017: p. 18-21.

[11] Pathak, A.R., et al., Empirical evaluation of deep learning models for sentiment analysis. 2019. 22(4): p. 741-752.

[12] Qaisi, L.M. and I. Aljarah. A twitter sentiment analysis for cloud providers: a case study of Azure vs. AWS. in 2016 7th International Conference on Computer Science and Information Technology (CSIT). 2016. IEEE.

[13] Sohangir, S., et al., Big Data: Deep Learning for financial sentimentanalysis. 2018. 5(1): p. 3.

[14] Zhu, L., et al., SentiVec: Learning Sentiment-Context Vector via Kernel Optimization Function for Sentiment Analysis. 2020.

[15] Çano, E. and M. Morisio. A deep learning architecture for sentiment analysis. in Proceedings of the International Conference on Geoinformatics and Data Analysis. 2018.

[16] Markoski, Filip. (2020). Sentiment Analysis of Twitter Data with Graph Structures and Deep Learning. 10.13140/RG.2.2.14678.88644.