SENTIMENT ANALYSIS OF NEWS ARTICLES USING BIDIRECTIONAL RECURRENT NEURAL NETWORKS

ALEXANDER TAN KA JIN

UNIVERSITI TEKNOLOGI MALAYSIA

PSZ 19:16 (Pind. 1/23)

UNIVERSITI TEKNOLOGI MALAYSIA
DECLARATION OF THESIS

Author's full name	:		
Matric No.	:	Academic Session	:
Date of Birth	:	UTM Email	:
Thesis Title	:		

I declare that this thesis is classified as:

☐	**OPEN ACCESS**	I agree that my report to be published as a hard copy or made available through online open access.
☐	**RESTRICTED**	Contains restricted information as specified by the organization/institution where research was done. *(The library will block access for up to three (3) years)*
☐	**CONFIDENTIAL**	Contains confidential information as specified in the Official Secret Act 1972)

(If none of the options are selected, the first option will be chosen by default)

I acknowledged the intellectual property in the thesis belongs to Universiti Teknologi Malaysia, and I agree to allow this to be placed in the library under the following terms :

1. This is the property of Universiti Teknologi Malaysia
2. The Library of Universiti Teknologi Malaysia has the right to make copies for the purpose of only.
3. The Library of Universiti Teknologi Malaysia is allowed to make copies of this thesis for academic exchange.

Signature of Student:

Signature :

Full Name
Date :

Approved by Supervisor(s)

Signature of Supervisor I:	Signature of Supervisor II
Full Name of Supervisor I	Full Name of Supervisor II
Date :	Date :

NOTES : If the thesis is CONFIDENTIAL or RESTRICTED, please attach with the letter from the organization with period and reasons for confidentiality or restriction

This letter should be written by a supervisor and addressed to Perpustakaan UTM. A copy of this letter should be attached to the thesis.

Date:
Librarian
Jabatan Perpustakaan UTM,
Universiti Teknologi Malaysia,
Johor Bahru, Johor

Sir,

CLASSIFICATION OF THESIS AS RESTRICTED/CONFIDENTIAL
TITLE:
AUTHOR'S FULL NAME:

Please be informed that the above-mentioned thesis titled **_________** should be classified as RESTRICTED/CONFIDENTIAL for a period of three (3) years from the date of this letter. The reasons for this classification are
(i)
(ii)
(iii)

Thank you.

Yours sincerely,

SIGNATURE:
NAME:
ADDRESS OF SUPERVISOR:

“I hereby declare that I have read this project report and in my
opinion this project report is sufficient in terms of scope and quality for the award of
the degree of Masters in Data Science”

Signature	:	____________________
Name of Supervisor	:	MY SUPERVISOR
Date	:	JANUARY 15, 2025

Declaration of Cooperation

This is to confirm that this research has been conducted through a collaboration ___________________________ and ___________________________.

Certified by:

Signature :

Name :

Position :

Official Stamp

Date

* This section is to be filled up for theses with industrial collaboration

Pengesahan Peperiksaan

Tesis ini telah diperiksa dan diakui oleh:

Nama dan Alamat Pemeriksa Luar :

Nama dan Alamat Pemeriksa Dalam :

Nama Penyelia Lain (jika ada) :

Disahkan oleh Timbalan Pendaftar di Fakulti:

Tandatangan :

Nama :

Tarikh :

SENTIMENT ANALYSIS OF NEWS ARTICLES USING BIDIRECTIONAL RECURRENT NEURAL NETWORKS

ALEXANDER TAN KA JIN

A project report submitted in partial fulfilment of the
requirements for the award of the degree of
Masters in Data Science

Faculty Of Computing
Universiti Teknologi Malaysia

JANUARY 2025

DECLARATION

I declare that this project report entitled *"Sentiment Analysis of news articles using Bidirectional Recurrent Neural Networks"* is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	______________________________
Name	:	ALEXANDER TAN KA JIN
Date	:	JANUARY 15, 2025

ABSTRACT

Sentiment Analysis has seen recent developments with the use of word embedding and deep learning methods. Despite this, it has been largely limited to short-form textual analysis such as tweets and reviews. This project aims to apply sentiment analysis onto longer forms of text with less opinionated forms of language such as news articles which can be biased by authorial intent but otherwise remain objective. In order to identify both subjective and objective forms of media bias, this project proposes the usage of Bidirectional Recursive Neural Networks (BiRNN) models like Bidirectional Long Short Term Memory (BiLSTM) and Bidirectional Gated Recurrent Units (BiGRU) as methods of accounting for subtle patterns in sentiment. The project uses the AllTheNews 2.0 data for analysis and through initial exploratory data analysis, have revealed that media bias correlates with sentiment scores.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	-	Artificial Intelligence
IR	-	Information Retrieval
ML	-	Machine Learning
DL	-	Deep Learning
NLP	-	Natural Language Processing
tf-idf	-	term frequency-inverse document frequency
W2V	-	Word2Vec
CBOW	-	Continuous Bag of Words
GLoVe	-	Global Vectors for word representation
ANN	-	Artificial Neural Networks
RNN	-	Recursive Neural Network
Bi-	-	Bidirectional model
BiRNN	-	Bidirectional Recursive Neural Network
LSTM	-	Long Short Term Memory
GRU	-	Gated Recurrent Unit
LSTM	-	Bidirectional Long Short Term Memory
BiGRU	-	Bidirectional Gated Recurrent Unit
GPT	-	Generated Pre-trained Transformers
BERT	-	Bidirectional Encoder Representations from Transformers

CHAPTER 1

INTRODUCTION

The subject of Natural Language Processing is the application of machines and computation within the domain of language. Part of it is Sentiment Analysis, which analyses neutrality, opinion strength and positivity/negativity from text and language use. Within recent times, the domain has utilized the tools of Deep Learning and Artificial Intelligence in order to analyse the complicated patterns of semantics and syntax, moving away from older machine learning methods and rules-based lexicon methods. The use of models like Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) allow better performances in the study of language use in everyday life such as within social media sites like Twitter (now X).(Wankhade et al., 2022) Language can be used objectively or subjectively, either a person is stating facts about a subject or giving their emotional opinions or narratives around the subject. One domain that is often not explored within sentiment analysis is the domain of long-form text like news articles.

Politics is a domain that tends to be filled with various opinions within a given subject. Whether a given policy, law, political figure or institution is good or bad is the main form of discussion within politics. Despite their intention of bringing objective news, journalism and news articles are written by people and institutions with certain political opinions and ideologies and may create narratives that reflect their political biases through the use of language. (Baly et al., 2020) These opinions and ideas are often embedded in otherwise objective journalism and Whether intentional or not, this use of language can push narratives that sways people into certain political beliefs and camps and is inconducive to a proper functioning democratic society. Compared to machine learning models, the use of deep learning models allows machines to extract certain features on their own and allows them to potentially detect more complex patterns of sentiment.

This thesis aims to perform sentiment analysis on news of various forms of bias in order to uncover the relationship between sentiment and propaganda. It uses a deep learning approach in order to analyse this relationship and how subjective viewpoints are pushed through objective language.

1.1 Problem Background

Recent studies of Sentiment Analysis (see Literature Review) have increased in their application of Deep Learning models such as LSTM and Transformer Models. This is a trend that is unique to contemporary times where the internet is adopted by billions of people and millions of data are made readily available, which is perfect for the use of large data deep learning models. Often the subject and dataset of analysis is short-form media like Twitter (now X) posts and consumer opinions, less work is done on long form media such as news articles.

The main issue of sentiment analysis in news is the tendency for journalists to phrase words objectively rather than utilizing subjective language. (Alonso et al., 2021) This provide a unique challenge for sentiment analysis as sentiment on news articles is expressed more subtly compared to messages on social media. (Balahur et al., 2013) Lexicon-based techniques rely weighing the importance of certain words and analysing the sentiment of each word which can fail to pick up the finer details of propaganda in news. (Da San Martino et al., 2019) identified 18 methods of language use in propaganda within news articles which are often overlooked by classification models and techniques, the same article also notes that previous studies on propaganda in news tended to label sources as biases and not news.

Due to the recent success in identifying patterns within text using deep learning, deep learning models such as Long Short Term Memory (LSTM) and Bidirectional Encoder Representations from Transformers (BERT) provide promising avenues for sentiment analysis, especially on text that contains more subtle forms of sentiment. The consideration of these insights from previous studies provide an avenue for investigating the correlation between sentiment and political propaganda and therefore political biases within news outlets and articles.

1.2 Problem Statement

This thesis aims to perform an analysis of the relationship between sentiment within text and use of language within news using deep learning models. The research aims to address the lack of sentiment analysis on long-form text data that is published within various media outlets rather than focusing on short-form text data within social media. In particular it aims to uncover and analyse in detail how subjective opinions are embedded within objective text. It analyses the efficacy of sentiment analysis with regards to identifying the degree of opinionated news. The analysis will also determine the relationship between media outlets and biased and opinionated journalism. With the use of Deep Learning technology, this research aims to use models such as LSTMs to better analyse textual data. This research aims to help citizens understand the use of language better within media and helps understanding how journalism can manipulate language to advocate or disagree with certain figures and policies.

1.3 Research Objectives

1. Perform an exploratory sentiment analysis on large news datasets to identify patterns and trends in the language use of journalism.
2. Construct BiLSTM and BiGRU models to uncover complex language patterns and potentially reveal sentiment hidden under objective language.
3. Conduct comprehensive evaluation on the model to ensure it's performance.
4. Visualize article sentiment by each media outlet.

1.4 Research Questions

1. What is the efficacy of utilizing Deep Learning models on long-form sentiment analysis compared to older methods?
2. To what degree does deep learning sentiment analysis properly account for the analysis of propaganda within news outlets and articles?

3. Is there a relationship between news media outlet bias and degree of opinionated language?

1.5 Significance of the Research

The research can help people identify the biases of new sources and how sentiment and opinions are used as forms of propaganda. Which can help people critique media institutions and ensure that people can differentiate opinion and facts within a long article. Additionally, the research contributes to the use of deep learning models within sentiment analysis of long-form textual data and sentiment analysis of objective and neutral language.

1.6 Study Scope

The study use a large set of news media data from recent years and only considers the use of models, BiLSTM and BiGRU. The research will conduct preliminary analysis of the data, extensive data cleaning and preprocessing before it is used to train our choice of model. The model will then be meticulously evaluated for it's robustness and performance before being used to predict emotion sentiment of articles. The scope of the analysis of articles will cover the purported media bias of it's original institution and it's popularity as a source of news along with the sentimental score of the article. All of it will be presented in the study as a visualization using graphs. Finally, the project will examine and address it's limitations for future work and enhancements in both the use of data and the construction of the model.

CHAPTER 2

LITERATURE REVIEW

2.1 Polarity of language within politics

Politics heavily uses language. The subject of politics is mainly communicated using language, politicians utilize certain words for their rhetoric, media can manipulate sentences to paint a different picture for or against a cause, languages and mannerism to shape and influence public support for policies and discourse within politics is frequently a site of analysis for many journalists and political analysts. This language can have various sentiments, ranging from anger to hope, parties, politicians and citizens can have strong opinions on these subjects and is communicated through textual opinions on various forms of media. Given this, the analysis of these opinions is done through the analysis of the language of politics like in (Orellana & Bisgin, 2023) where opinions can be classified in terms of importance to a given political group. A significant amount of these opinions are communicated through online social media platforms and newspaper like Twitter (now also known as X), which provides an opportunity for textual analysis. (Németh, 2023)'s review of political polarity research using Natural Language Processing (NLP) notes that sentiment polarity often uses political polarity as a basis for it's measurement, even though political polarity is not a well-defined term. Such political polarity measures include the four measures by (DiMaggio et al., 1996), 1. variance of opinion, 2. bimodality of opinion, 3. ideological constraint of opinion, 4. differences in in-group opinion. From there, sentiment polarity can be measured as the propensity for political groups to hold to a certain group. That being said, most papers utilize sentiment polarity as the strength and attitude of a given opinion between negative opinion and positive opinion such as (Alshutayri et al., 2022).

Sentence	Polarity
Stupid storm. No river for us tonight	Negative
that's great!! weee!! visitors!	Positive
looking forward to body works today	Neutral

Table 2.1 Examples of sentiment from tweets on Twitter. Gathered from Kaggle Dataset: (https://www.kaggle.com/datasets/abhi8923shriv/sentiment-analysis-dataset Accessed 22:50 14/12/2024)

2.1.1 Natural Language Processing and it's challenges

(Nadkarni et al., 2011) notes in their introduction that NLP originated as the intersection of artificial intelligence (AI). (Liddy, 1998) describes NLP as a range of techniques for processing natural text in hopes of achieving human-like text comprehension. This subject was initially distinct from Information Retrieval (IR), which was concerned with efficiently identifying and querying large corpora of text. It's integration with IR occurred with a need for identifying underlying meanings of text, often in regards to text that contain non-conventional uses of language such as sarcasm or irony that contain semantics outside of it's literal meaning. Similarly in Text Mining, the identification of irony and sarcasm is crucial to yielding accurate results from analysis of data like Twitter data.

Irony and sarcasm has been traditionally the main obstacle to sentiment analysis within social media where sentences may be more opinionated and more subjective. Within news media however, journalists are expected to maintain an objective form of writing that removes the presence of irony. Instead, news media will utilize methods such as loaded language, usage of logical fallacies or obfuscation as a means of introducing bias and persuading people into supporting certain causes. (Da San Martino et al., 2019) Outside of specific language use, media often introduce biases by selective coverage of events or by presenting events in specific manners which is harder to analyse with textual analysis and is better studied via the analysis of headlines. (Rodrigo-Ginés et al., 2024)'s literature review outlayed several methods of measuring media bias categorized by bias authorial intent and contextual biases. Methods that are important to this thesis and are measurable by textual analysis are mostly authorial intent biases like spin biases (sensational coverage of events and emotionally charged labels) and ideological biases (selective usage of adjectives and opinionated statements).

2.1.2 Text Mining

Similar to IR, Text Mining is the discovery of patterns within unstructured textual data, it deals with the processing and comprehension of complex textual patterns in order to uncover hidden trends. Like IR, Text Mining deals with large corporea of text and is concerned with the complexities of language.(Stavrianou et al., 2007) The rules of language can be complex and change frequently to fit the cultural context of it's speakers. (AbuSa'aleek, 2015) notes that internet users have a change in words and expression in English internet discourse ranging from shortening of words to the use of emoticons while (Bengio et al., 2000) mentions that traditional language models suffer greatly from the curse of dimensionality and have decreases in efficiency owing to the number of words in a given vocabulary. The analysis of text has to adjust to these changes in both morphology and semantics while also accommodating the diversity of expressions and words.

Due to the complexities of text, much of NLP have moved away from statistical models (n-gram, maximum entropy) into the domain of Machine Learning utilizing the presence of larger corpora with the development of models like BERT and GPT-3, achieving higher results than traditional models. (Wei et al., 2023) The use of Machine Learning helps tremendously with pattern recognition and mining of textual data. Still, more research is to be done on more complicated use of words in different languages and context (such as humour or disgust and hate) (Farabi et al., 2024).

2.1.3 Semantics

This thesis concerns mainly with the semantics of language. Semantics is one of the main disciplines of language along with phonetics (individual sounds), phonology (the organization of sounds), morphology (the formation of sound into words), syntax (the structure of sentences), pragmatics (how information is conveyed through language), discourse (use of language in conversations) and socio-linguistics (how society influences language). These components of language are listed in the Handbook of Linguistics (Aronoff & Rees-Miller, 2020), where semantics is listed as the study of meaning developed through words and sentences. Though semantics is mainly thought to come from words and phrases, the meaning of a given sentence can change depending on any of the other components of language like discourse, phonology and socio-linguistics.

(a) Thanks again for the full fries!

(b) Another perfect pizza from <user>!

Figure 2.1 Examples of how social media users may post sarcastic messages. The semantics of the sentence may be literal but is rendered not due to the juxtaposition of the images. Similarly, the discussion surrounding a conversation may render a message sarcastic. (Farabi et al., 2024)

2.1.4 Sentiment Analysis

Sentiment is the emotional reaction of a given person. In text, we often communicate our emotions and opinions using words with specific connotations we learn. Examples of words that communicate strong sentiment may include direct emotions (Happy, Sad, Afraid), objects that evoke emotions (Baby, Rain, Rainbow) or adjectives (Cool, Warm, Bitter). These opinions can be all sorts of different emotions ranging from anger, sadness to joy and surprise and of different polarities from extremely negative to extremely positive (Taboada, 2016).

2.1.5 Goal of Sentiment Analysis

The purpose of Sentiment Analysis (also known as Opinion Mining) is to uncover and summarize the various opinions hidden within large corpora of text in order to discover cultural and societal opinions on subjects. Papers such as (Singh et al., 2022) classified tweets on Twitter during the Covid-19 pandemic with the labels of joy,sad,fear and anger using various machine learning models in order to help organizations adapt to current world events. From sentiment analysis, research can uncover the trends in opinion within given subjects like pandemics or opinions on political/cultural issues such as dementia.(Kong et al., 2022)

Figure 2.2 Examples of various emotional sentiments in Weibo messages and their sentiment labels. (Li & Li, 2023)

2.1.6 Machine Learning based Sentiment Analysis

Of the methods in sentiment analysis, two are most significant, Machine Learning approaches and Lexicon-Based approaches.(Taboada, 2016) A majority of Sentiment Analysis utilizes Machine Learning/Deep Learning methods. These methods utilize a textual dataset that is transformed by Feature Engineering to emphasize relevant information. The machine learning models then train and classify based on the transformed dataset. Machine learning approaches are more popular within the domain. As (Rodríguez-Ibánez et al., 2023) notes, machine learning became more popular when vector embedding was introduced where words are transformed into vector space, allowing a mathematical representation of language. Deep learning, Support Vector Machines (SVM) and Bayesian methods are some of the most commonly used techniques for sentiment analysis with more recent deep learning models like Long Short-term Memory (LSTM) and Transformers like Generative Pre-trained Transformers (GPT) and Bidirectional Encoder Representations from Transformers (BERT) achieving higher results. This thesis will use this branch of sentiment analysis and several deep learning and machine learning models will be discussed in detail later in the thesis.

2.1.7 Lexicon-based Sentiment Analysis

These methods differs from lexicon based approaches that develop rules based on dictionaries and text and separate the data according to the rules. Mentioned in (Saberi & Saad, 2017), a lexicon-based method like semantic orientation would use k-means clustering to group documents and sentences based on a set of positive/negative words to form a dictionary, the text would be then be weighted upon using term frequency-inverse document frequency (TF-IDF) and a voting technique to measure it's importance and fine-tune the dictionary. The key advantage that Lexicon-based approaches have was their ability to be applied to a wide variety of corpora without changes to the dictionary (Taboada et al., 2011) allowing for easy reuse. But with increases in computational capabilities, recent research is heavily skewed towards Machine Learning methods.

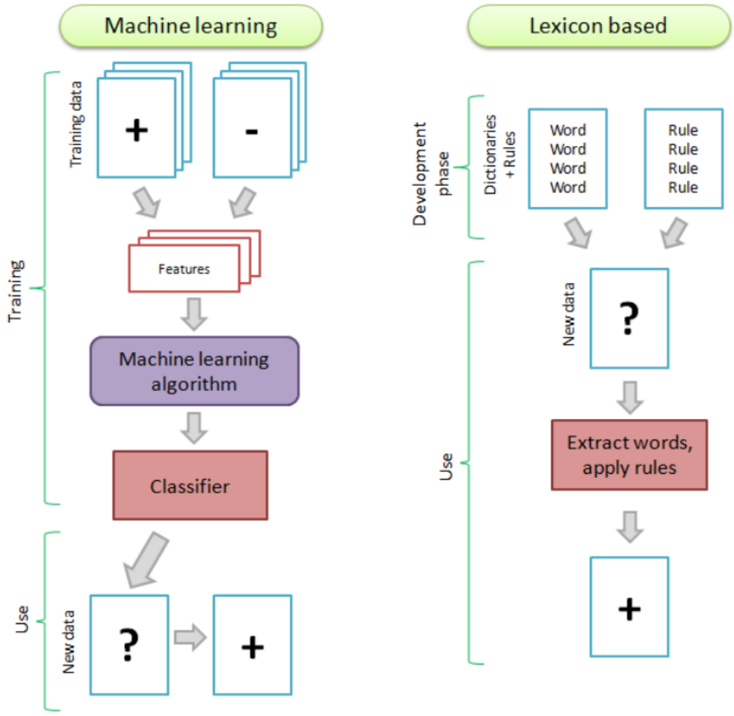

Figure 2.3 Difference between Machine Learning methods and Lexicon-based methods. (Taboada, 2016)

2.1.8 Challenges and Research Gaps of Sentiment Analysis

The main challenge to all sentiment analysis is non-literal use of language. As stated above, sarcasm, irony and satire all use language in a way that communicates the opposite of it's literal meaning. (Maynard & Greenwood, 2014) analysed tweets using labels of polarity, strength of opinion and sarcasm and showed that accounting for sarcasm could have an improvement on Text Mining systems. In terms of political humour, (Frenda et al., 2018) noted that aspects of bigotry like misogyny are often transmitted through sarcastic jokes, identification of sarcasm and correct labelling of sentiment may be important for political jokes that demean groups of people.

2.2 Machine Learning/Deep Learning Classification Models

The domain of Artificial Intelligence (AI) is largely split between two types of classifications. (Janiesch et al., 2021) distinguishes these two classifications as the model type (Machine Learning (ML) methods which includes Artificial Neural Networks (ANN) which includes Deep Learning (DL) methods) and the learning type (supervised, unsupervised, semi-supervised and reinforcement learning).

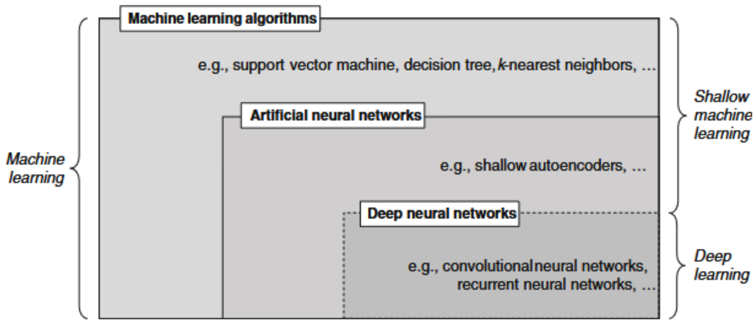

Figure 2.4 Venn diagram grouping on ML Methods. DL is a subset of ML that utilizes complicated neural networks. (Janiesch et al., 2021)

Much of sentiment analysis is done through supervised classification ML/DL such that the model is able to recognize the concept of polarity and emotional labels. This type of learning occurs when models are first trained on a set of data with labelled outputs (positive/negative, emotional labels, etc) and then utilized for use and evaluation in unlabelled data. In general, both supervised ML and DL follow the same basic methodology:

1. Labelled Data Input
   Data is labelled with it's correct output and fed to the model. This directs the model so that it is useful for classification.

2. Model Training
   The model is trained to identify patterns within the data provided. The results of the model are transformed by the loss function into a value that is gradually optimized. For deep learning models this is usually a form of gradient descent. (Andrychowicz et al., 2016) The next sections will go into detail on how

ML/DL models are gradually trained to identify patterns and the specific architecture of these models.

3. Model Evaluation

   The model is evaluated based on it's performance. Common evaluation metrics for classification models include Precision, Recall, Accuracy, F1-score and Receiver Operating Characteristic(ROC)-Curve

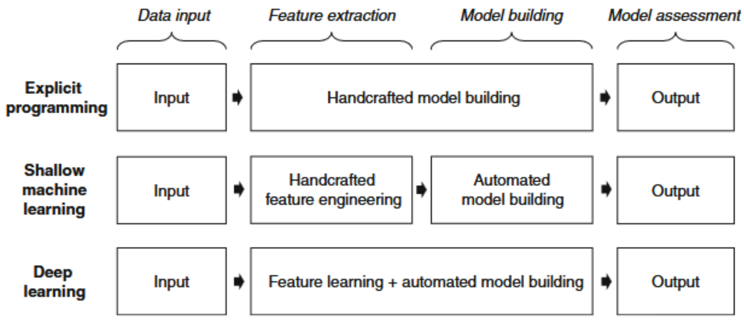

Figure 2.5 Comparison of explicit programming methods, machine learning and deep learning. (Janiesch et al., 2021)

2.2.1 Neural Networks

Artificial Neural Networks (ANN) are at the heart of deep learning. They are composed of several layers of artificial neurons within hidden layers. Each neuron has weight inputs and transforms them via an activation function of choice (sigmoid, tanh, softmax, so on.) The output weight is then fed to the next layer, continuing until it reaches the end of network where the value of the prediction is calculated by a loss function (usually Mean Square Error) against it's training dataset. (Zakaria et al., 2014) The network ”learns” and self-corrects by performing back-propagation. Starting from the output layer, back-propagation derives every layer in order to minimize the loss of the model.

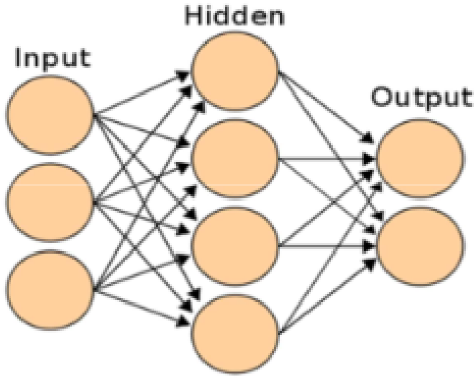

Figure 2.6 Illustration of a simple ANN model. (Islam et al., 2019)

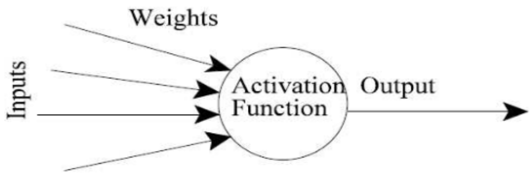

Figure 2.7 Illustration of an artificial neuron. The neuron is fed weights from previous layers and transforms it into an output by an activation function. (Zakaria et al., 2014)

2.3 Feature Extraction Methods

The main distinction of DL methods from other ML methods like Support Vector Machines (SVM) is the reduced need for feature engineering, that is the preprocessing of data in order to reduce the dimensionality of it such that the ML model can achieve suitable accuracy. (Zebari et al., 2020) DL Models do not suffer from the requirement of feature engineering due to it's ability to find features by itself. As (LeCun et al., 2015) notes, DL models operate on successive feature extraction layers which can find significant features of data by continuously abstracting from higher representative patterns. Given that textual data and language contains high dimensional data such that multiple factors can affect the semantics of a given sentence, it is easy to see how the self extraction of significant features in DL can result in improved performances in both Sentiment Analysis and NLP. That being said, DL methods still require the use of feature extraction in order to convert textual data into mathematical representations before the DL model can extract more abstract features.

(Chaudhary et al., 2023)'s overview of Machine Learning and Deep Learning method in Sentiment Analysis for Twitter data noted that a majority of Deep Learning methods utilize Word2Vec, TF-IDF and GloVe in order to transform text into mathematical representations.

2.3.1 Term Frequency-Inverse Document Frequency (TF-IDF)

TF-IDF weights words based on their importance within a given document. This is based on the frequency of the word within the document and within the corpus. The weight of the word is increased based on it's frequency of within the document while simultaneously decreased by it's frequency within the wider corpus.(Lubis et al., 2021) The total equation is given as:

$$TF - IDF_{t,d} = tf_{t,d} \cdot idf_t$$

For word t in document d, $tf_{t,d}$ is the weight of importance based on it's frequency within document d while idf_t is the inverse weight of a term based on df_t, the frequency of documents that contain word t against the number of documents within the corpus, N. There are a variety of different methods to calculate the two terms:

Term Frequency $tf_{t,d}$		Inverse Document Frequency idf_t	
natural	$f_{t,d}$	no frequency	1
logarithmic	$1 + \log(f_{t,d})$	idf	$log\frac{N}{df_t}$
augmented	$0.5 + \frac{0.5f_{t,d}}{mode_d}$	probabilistic	$max\{0, log\frac{N-df_t}{df_t}\}$
boolean	$f(x) = \begin{cases} 1 & \text{if } f_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$		

Table 2.2 List of tf-idf methods documented by (Schütze et al., 2008)

For $f_{t,d}$ being the frequency of term t in document d, $mode_d$ being the count of the most frequent term in document d. N being the total number of documents in the corpus, df_t being the number of documents containing term t.

A common method of calculating $tf_{t,d}$ is the logarithmic method, where the importance of words is not measured linearly like in natural count. Another method is by augmented $tf_{t,d}$ where the weight of importance is measured against the frequency of the most common word in the document.

2.3.2 Word2Vec

Until Word2Vec, most techniques for extracting textual features ignore the surrounding context of a given word like TF-IDF. Word2Vec is a technique for transforming text into representational vectors that utilizes one of two simple neural network models, Continuous Bag of Words (CBOW) and Skip-gram. (Ma & Zhang, 2015) notes that CBOW predicts the appropriate words given adjacent words (word order does not matter) and Skip-gram does the opposite, predicting the adjacent words using the words given. Using either of the two models, Word2Vec constructs a lexicon of words from the dataset and then assigns weight to each word to a point in vector space, such that similar words are placed closed to each other while dissimilar words are placed further apart. Due to it's ability to consider the context of a word, Word2Vec has been popularly adopted as a feature extraction method for many NLP tasks.

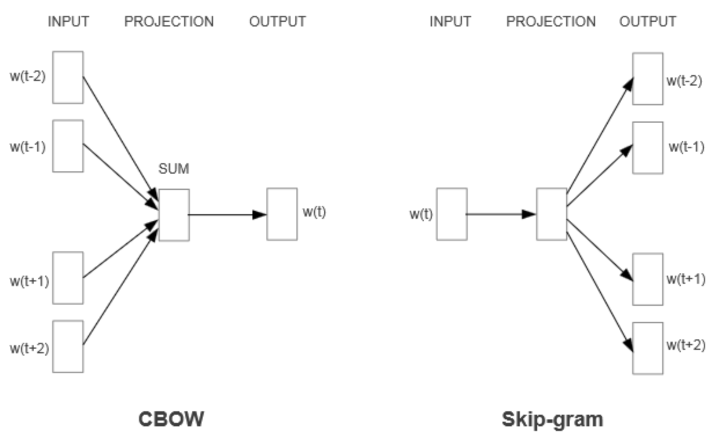

Figure 2.8 Illustration of Word2Vec's two models, Continuous Bag of Words (CBOW) and Skip-gram. CBOW takes words and predicts the most appropriate word in the context. Skip-gram takes a word and generates the most appropriate words outside of the input word. Both of these methods will produce a vector of weights that represents it's similarity to other words in vector space. (Mikolov, 2013)

2.3.3 Global Vectors (GloVe)

Word2Vec is a powerful method for feature extraction of text, but it focuses on small windows of words and ignore the global view of how words interact. As introduced in (Pennington et al., 2014), GloVe is a method that uses a large global matrix to represent the probability of any given word co-occurrence represented by the conditional probability:

$$g_{i,j} = P(w_i|w_j) = \frac{P(w_i \cap w_j)}{P(w_j)}$$

For i-th row, j-th column within the GloVe matrix, $g_{i,j}$ is represented by the probability that words w_i and w_j co-occur in a sentence/phrase over the probability that word w_j is present with a sentence. The matrix is then factorized by the equation to extract the weights of the words in a given sentence. This allows for extraction of the weights through a simple mathematical operation at the cost of utilizing large matrices that scale quickly with the number of words in a given corpus.

$$\begin{array}{l} \text{The} \\ \text{Cat} \\ \text{Sat} \\ \text{On} \\ \text{Mat} \end{array} \begin{bmatrix} 5 \\ 3 \\ 4 \\ 3 \\ 1 \end{bmatrix} = \begin{array}{|c|c|c|c|c|c|} \hline & \text{the} & \text{cat} & \text{sat} & \text{on} & \text{mat} \\ \hline \text{the} & 2 & 1 & 2 & 1 & 1 \\ \hline \text{cat} & 1 & 1 & 1 & 1 & 0 \\ \hline \text{sat} & 2 & 1 & 1 & 1 & 0 \\ \hline \text{on} & 1 & 1 & 1 & 1 & 1 \\ \hline \text{mat} & 1 & 0 & 0 & 1 & 1 \\ \hline \end{array} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Figure 2.9 Demonstration of matrix factorization for GloVe Matrices. The importance of each word is extracted given that words ”the” ”cat” ”sat” (represented as the vector on the right) appear within a sentence. This example may be misleading as it is preferable to understand co-occurrence matrices as a matrix of conditional probabilities. Example GloVe co-occurrence matrix given by (Hindocha et al., 2019)

2.4 Recurrent Neural Networks (RNN)

Although the utilization of ANNs within NLP and consequently, sentiment analysis, have been massive benefits in identifying patterns within unstructured textual data. That being said, ANNs do not have the ability to remember information from previous contexts. This makes the recognition of patterns in highly dynamical systems of information (such as language or speech recognition) achieve less than desirable accuracy. Due to this, the use of RNN is deemed to be an improvement on these subjects. The main modification of RNNs is the presence of prior knowledge from previous layers, which allows RNNs to have limited knowledge of previous words and phrases and making use of the order of sentences and words. (Tarwani & Edem, 2017)

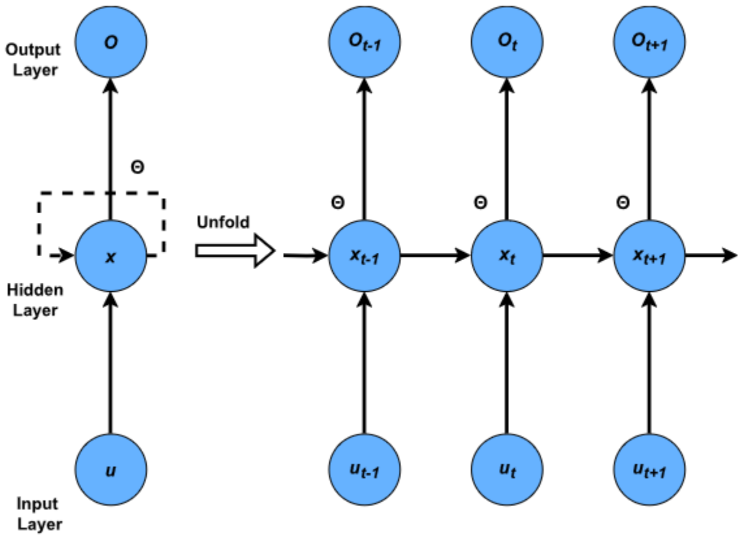

Figure 2.10 Illustration of RNN architecture. The left image depicts the neural network containing a hidden layer that loops back information from the previous step. When unfolded, information from time $t-1$ is fed to the RNN at time t. Information at time t is fed to time $t+1$ and so on. (Alhajeri et al., 2024)

Some recent studies have been using bidirectional RNNs that promised increased results compared to single directional RNNs, at the expense of increased computational time during back-propagation. (Mahadevaswamy & Swathi, 2023)

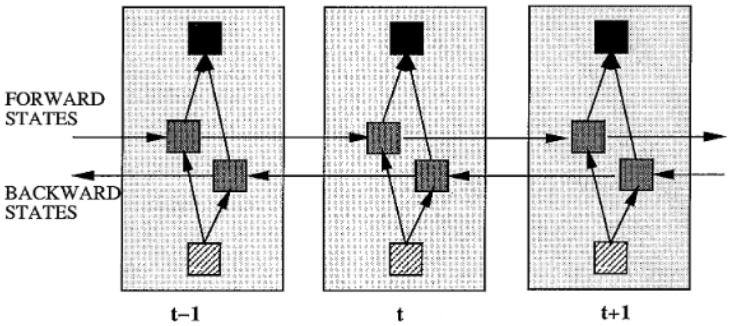

Figure 2.11 Illustration of Bidirectional RNNs, this applies to subsequent models like LSTM and GRU. The information is passed through two RNNs understanding context from two directions. (Schuster & Paliwal, 1997)

2.4.1 Long Short-Term Memory (LSTM)

(Noh, 2021) noted that the standard RNN model has difficulty learning information in the long run. The training improvement will gradually vanish and long term training of RNNs will bring about increasingly diminishing returns. Similarly, the RNN may explode in learning gradient that equally leads to a lack of training progress. These two issues were described by (Bengio et al., 1994). The problem comes with the use of back-propagation, the gradient during back-propagation. (Pascanu et al., 2013) examined the gradient of back-propagation in RNNs and highlighted that the cause of vanishing/exploding gradients was due to repeated multiplications in the differential between RNN states. A common solution to this is to utilize a variant of RNN called

$$\frac{\partial \mathcal{E}}{\partial \theta} = \sum_{1 \leq t \leq T} \frac{\partial \mathcal{E}_t}{\partial \theta} \quad (3)$$

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{1 \leq k \leq t} \left(\frac{\partial \mathcal{E}_t}{\partial \mathbf{x}_t} \frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} \frac{\partial^+ \mathbf{x}_k}{\partial \theta} \right) \quad (4)$$

$$\frac{\partial \mathbf{x}_t}{\partial \mathbf{x}_k} = \prod_{t \geq i > k} \frac{\partial \mathbf{x}_i}{\partial \mathbf{x}_{i-1}} = \prod_{t \geq i > k} \mathbf{W}_{rec}^T diag(\sigma'(\mathbf{x}_{i-1})) \quad (5)$$

Figure 2.12 The issue with back-propagation in RNNs as highlighted by (Pascanu et al., 2013). In the general back-propagation chain, $\frac{\partial \epsilon}{\partial \theta}$, it is required that it also account for the gradient between two states of a RNN, $\frac{\partial x_t}{\partial x_k}$. This differential is comprised of only multiplications of gradient between the two states, making it susceptible to greatly amplifying/diminishing the gradient of the network over time.

LSTM (Hochreiter, 1997). LSTM does not address with the exploding gradient, again as noted by (Pascanu et al., 2013), but it does adequately address the vanishing gradient problem by providing a long term memory channel, allowing the results of older states to leak into new states. The model utilizes two channels, a long term memory channel c_t and a short term memory channel, h_t (illustrated in Figure 13 as the top channel and bottom channel respectively). The model is designed such that the results of long term memory will leak into short term memory and short term memory will modify long term memory. This allows for a broader view than RNNs and allows significant words and phrases in the beginning of a sentence/document to have greater effect on the prediction overall. (Lakretz et al., 2019) notes that LSTMs are able to capture specific linguistic information, that models contain units that are able to capture information such as subject-verb dependency.

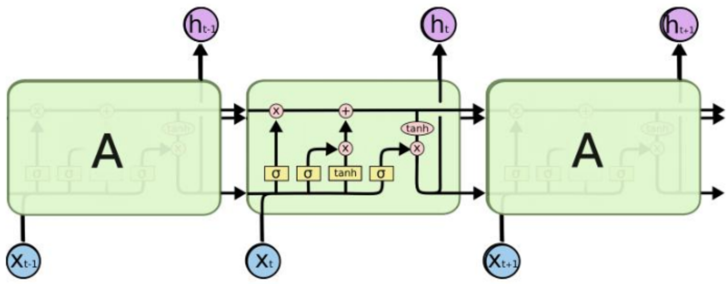

Figure 2.13 Simplified illustration of the LSTM model. (Tarwani & Edem, 2017)

2.4.2 Gated Recurrent Units (GRU)

GRUs are a subset of LSTMs that are simpler than the traditional LSTM model. The main difference is the combination of the long-term memory channel and short-term memory channel into a single channel. The model is worse than LSTM by a bit, but has demonstrated improved running times. (Shewalkar et al., 2019)

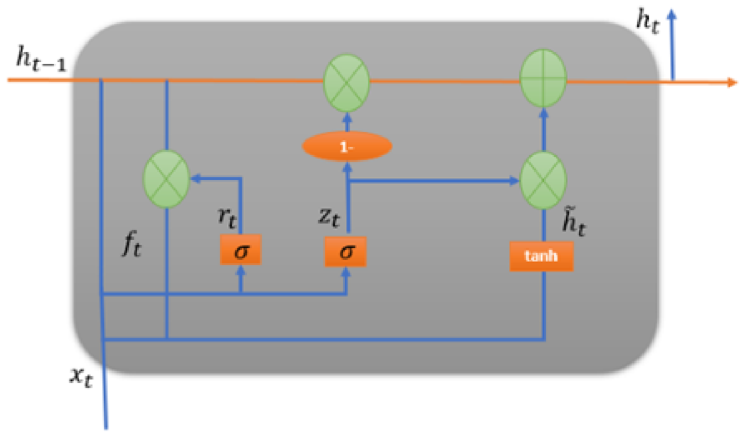

Figure 2.14 Simplified illustration of the GRU model. (Mateus et al., 2021)

Paper	Dataset Language	Model	Accuracy	Precision	Recall
(Singh et al., 2022)	English	LSTM-RNN with attention mechanism	0.8456	0.8234	0.8212
(Said et al., 2023)	English	LSTM-2BiGRU	0.9246	0.9297	0.9194
(Mu et al., 2024)	English	IDBO-CNN-BiLSTM	0.8033	0.8235	0.8207
(Eyvazi-Abdoljabbar et al., 2024)	Farsi	Ensemble of CNN and LSTM Hybrid models	0.7234	0.72069	0.72198
(Halawani et al., 2023)	English	ABiLSTM	0.8425	0.8583	0.8637
(Yin et al., 2023)	English	DPG-LSTM		0.842	0.845
(Li & Li, 2023)	Chinese	CNN-BiLSTM	0.8778	0.8946	0.8841
(Nguyen, 2024)	English	BiLSTM, with GloVe	0.8218	0.8219	0.8218
(Kastrati et al., 2024)	English with Emojis	BiLSTM with attention mechanism	0.7022	0.7093	0.7091
(Vernikou et al., 2022)	English	LSTM with BERT tokenizer	0.91	0.91	0.91
(Vanam & Raj, 2023)	English	BET Tokenizer and H-LSTM	0.99	0.98	0.97
(Lin et al., 2023)	Chinese	BiGRU-Att	0.9758	0.98	0.98
(Tabinda Kokab et al., 2022)	English	BERT embedding and BiLSTM	0.95	0.96	0.97

Table 2.3 List of LSTM/GRU models used in literature.

2.4.3 Literature of LSTM use in sentiment analysis

From the table, we can see that LSTMs are effective at classifying text into sentiment with most models reaching 80% to 90% in evaluation metrics. In a survey of sentiment analysis methods, (Wankhade et al., 2022) noted the effectiveness of BiLSTM models in comparison to other types of models but the training cost and time is higher compared to other models, the same paper also noted that semi-supervised training is the best at handling ambiguous text while the more commonly used supervised training is effective at handling subjectivity. That being said most of the recent models utilize a hybrid model with the only non-hybrid model being BiGRU and BiLSTM. From the table, some of the best performing models are relatively simple LSTM/BiLSTM and BiGRU models.

CHAPTER 3

METHODOLOGY

3.1 Methodological Framework

A research methodological framework allows a researcher to illustrate the core steps in conducting a research and helps the reader to understand the given steps the researcher took to receive their end results. This aids in replication of the research and helps identify potential criticisms of the research. This thesis will follow four main steps.

1. Research Gap Identification
2. Data Collection
3. Data Preprocessing
4. Exploratory Data Analysis
5. Model Construction
6. Model Performance Evaluation
7. Model deployment

Through literature review, research gaps are found and the purpose of this study is constructed. We determined that this research will encounter the following challenges in it's methodology:

1. Determining the correct dataset and ensuring that the proper features are extracted from the data prior to model training.
2. Choosing the correct model and training method. Which can greatly impact the final behaviour of the model and the results of this research. If training

is supervised or semi-supervised, then the data labelling method must be considered too.

3. Choosing a dataset to analyse relationships between politics opinions and emotional sentiment.
4. Determining the proper visualizations such that relationships between data are made clear.

3.2 Data Collection

The subject of this study is long-form text in the form of news articles, therefore the data used has to be same. Most articles are freely available on internet and are often scraped by users on the net and published through Kaggle and Mendeley Data or on independent websites. The main obstacle to data collection is the diversity and size of the new article dataset. Preferably, the project requires a large dataset of atleast 1 million articles and sufficient diversity of data such that all kinds of news outlets are covered regardless of bias and regardless of the factuality of it's contents. This makes the All The News 2.0 Dataset (Thompson, n.d.) suitable for analysis. All the News 2.0 is a large dataset containing 2.6 million articles from news of different websites including author, title, main text and publication. The size and diversity makes it perfect for use in analysis of news media.

3.3 Model and training Choice

Prior to model construction, it is important to both chose the model and chose the training method that is being used for research. For this research, BiLSTM is chosen as it has been proven in the literature review to have promising results in sentiment analysis. BiGRU is similar and offers slightly lower performances but is more lightweight and would also be chosen alongside BiLSTM to compare performances. The bidirectionality of these models provide them the capability to analyse sentences in context of the sentences before and after and has been proven in Literature Review to have great results in sentiment analysis for shorter forms of text.

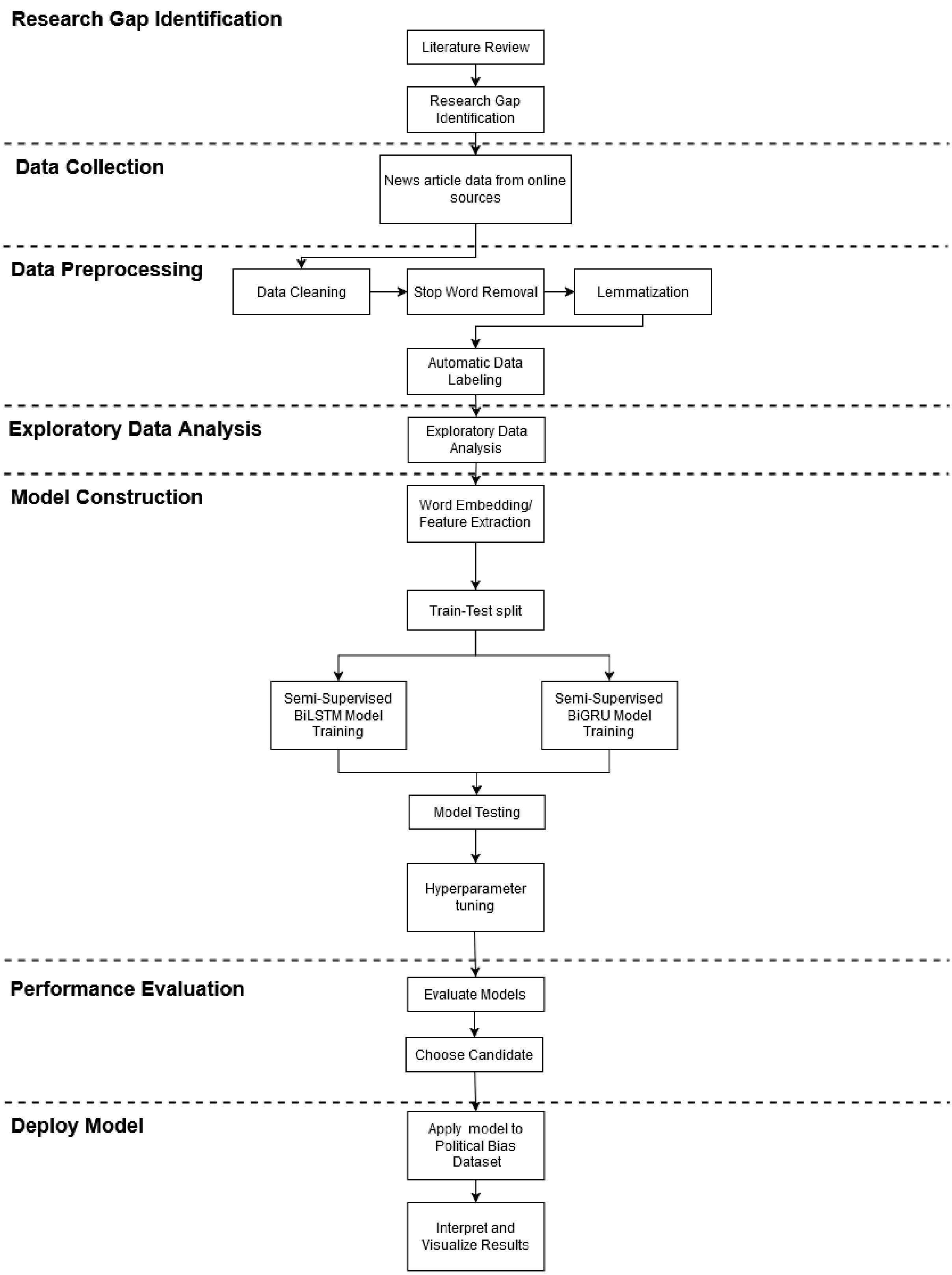

Figure 3.1 Illustration of the research framework

The type of model greatly determines the methods of data preprocessing and how much of the data is labelled. For most of previous sentiment analysis research, supervised analysis is used. (Wankhade et al., 2022) noted that semi-supervised training may be more effective at handling ambiguity (such as an ambiguous sentiment score) while supervised training is better at handling subjectivity. Since most of the data will contain objective use of language, large amounts of subjective language is not of concern. Ultimately, semi-supervised training will be used as to account for more equivocal sentences and sentiment that is harder to evaluate under supervised training.

3.4 Data Preprocessing

Data Preprocessing for text involves feature extraction so that the accuracy of the model is increased and sentiment labelling in order to direct the model into doing sentiment analysis.

3.4.1 Data Cleaning

Prior to preprocessing the data, we must ensure that the training/testing data does not contain any irrelevant noise that may decrease the accuracy of the model. For this research, the model must properly interpret professional English journalism and so flaws that may impede this process are:

1. Hyperlinks (https:// or http://)
2. Stop words (a,the,in)
3. Punctuations (, . @ !)
4. Unusually capitalized words/ lowercase words (sUCh aS THIS)
5. Non-Ascii Characters
6. Word contractions (-’s, -’re, -’d)

7. Duplicated words

In addition, missing values, invalid values (such as a text being too short), satirical articles (such as The Onion and The Babylon Bee) and tabloids/pop culture magazines (TMZ, The Sun,...) are removed. Words that are not the root of a word should then be lemmatized (eg. Happily becomes happy, mournful becomes mourn). To prepare the data for word embedding and sentiment labelling, the data will then be split into individual sentences.

3.4.2 Data Labelling

For the training/testing dataset, the data is labelled as to prepare for supervised training using automatic labelling software. This project will utilize TextBlob, a library for text mining and NLP, it utilizes a rule-based approach to labelling individual sentences. Compared to competitors like SentiWordNet 3.0 and VADER, TextBlob is able to output emotional polarity scores and sentence objectivity scores while simultaneously handling the context of surrounding words. During exploratory data

```
>>> from textblob import TextBlob
>>> blob = TextBlob("The food was awfully delightful")
>>> blob.sentiment.polarity
1.0
>>> blob.sentiment.subjectivity
1.0
>>>
```

Figure 3.2 Demonstration of TextBlob's sentiment labelling. TextBlob is able to probably handle the phrase ”awfully delightful” and accurately output a positive score for polarity and subjectivity.

analysis, all of the data will be labelled and utilized for a rules based analysis of the corpus. However, training will be semi-supervised and a majority of the labelled data will be obscured so as to facilitate the pseudo-labelling during training.

3.5 Word Embedding

The main methods of transforming words into vectors are documented within the literature review. Between the three options, Word2Vec is preferable as to avoid potential computational constraints that may be met when running GloVe. Word2Vec is comparably lightweight compared to GloVe while still extracting features based on context.

The vectors from the result of Word Embedding will then be normalized with min-max scaling as to ensure that data is standard and uniform while maintaining the relationships between vectors. Because Word2Vec is an unsupervised clustering model like K-means, it is difficult to evaluate it's accuracy.

3.6 Model Training and Data Split

The data will be split between Training, Testing and Validation on a 80/10/10% split. Training data is used to help the model learn and encode patterns within data, the testing data is used to evaluate the performance of the model and the validation set is used to tune hyper-parameters of the model. Within semi-supervised training, only a small part of the training data is labelled while the rest is pseudolabeled. Pseudolabelling is the process of labelling unlabelled training data and then adding it back to the training data on the next session. Only 10% of the training data will labelled this way and the remaining are pseudolabelled.
When a model is done training, it moves to the hyper-parameter tuning section of the phase where the validation set is used to adjust variables like learning rate and optimiser momentum. After which, it loops back to training until gradient change of the loss function is too little for any noticeable improvement.

As noted in model choice, the models used for this research will be BiLSTM and BiGRU. The two will be trained simultaneously and at evaluation, the better performing model will be chosen to be deployed.

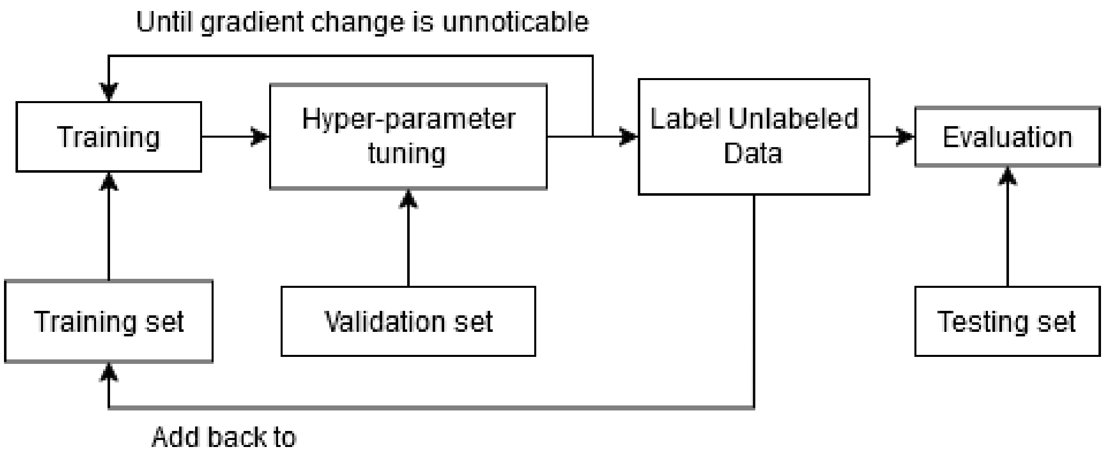

Figure 3.3 Semi-supervised training and evaluation process

3.7 Evaluation

The end model will be a classification model. A classification model labels data between two or more categories. Classification models are often evaluated using a confusion matrix which measures the amount of labels that were predicted correctly and incorrectly. The confusion matrix lists down the true positive (top left), false positive (top right), false negatives (bottom left) and true negatives (bottom right). Subsequent evaluation metrics like Recall, Precision, Accuracy and F1 score is derived from the initial confusion matrix. Due to the research's use of SentiWordNet, the confusion matrix used is 3x3 to evaluate positive, negative and neutral responses.

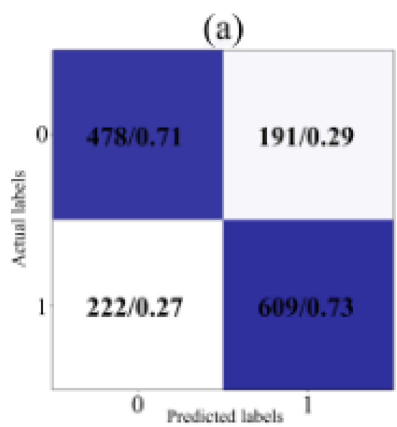

Figure 3.4 Example of a Confusion matrix(Mu et al., 2024)

Below are the formulae for Accuracy, Precision, Recall and F1 scores respectively within a nxn confusion matrix M for a given element α and given that columns are predicted values and rows are actual values. The diagonals are true positives and every other element is a mismatch.

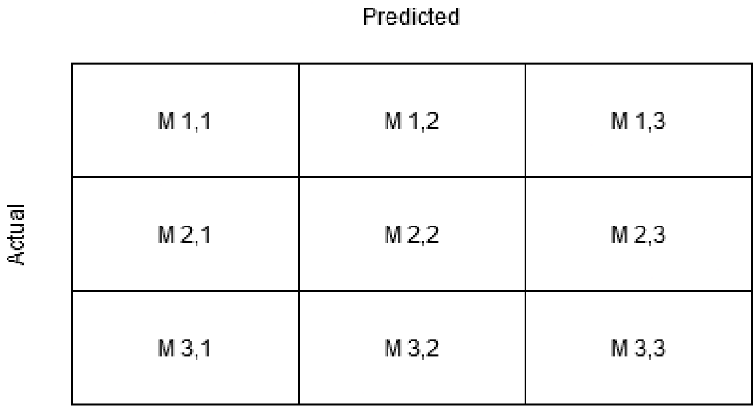

Figure 3.5 Reference for 3x3 confusion matrix

$$\text{Accuracy} = \frac{\sum_{i=0}^{n} M_{i,i}}{n} \tag{3.1}$$

$$\text{Precision}_\alpha = \frac{M_{\alpha,\alpha}}{\sum_{i=0}^{n} M_{i,\alpha}} \tag{3.2}$$

$$\text{Recall}_\alpha = \frac{M_{\alpha,\alpha}}{\sum_{i=0}^{n} M_{\alpha,i}} \tag{3.3}$$

$$\text{F1}_\alpha = \frac{2\text{Precision}_\alpha\text{Recall}_\alpha}{\text{Precision}_\alpha + \text{Recall}_\alpha} \tag{3.4}$$

$$\tag{3.5}$$

Accuracy is the general measurement of the model's ability to label text correctly. Precision of a given label is the amount of data that are correctly classified, recall of a given label is the amount of points classified under that classification. F1 is the mean of both precision and recall.

3.8 Model Deployment and Visualization

After the model is evaluated, the model is used for analysis for polarity within the analysis dataset (All The News 2.0) in order to measure the polarity of articles within various news publications. Prior to deployment, the dataset must also be cleaned in a similar manner to the data cleaning section as to ensure accurate results. The dataset is also labelled via AllSides (*AllSides*, n.d.) in order to categorize them by media bias labels ranging from far left to far right. The resulting statistics will be noted and published in the form of various graphs such as a scatter plot or bar chart which is frequently used for visualizing multiple dimensions of data at once. These visualizations will be made using matplotlib and the results of the analysis will be meticulously interpreted.

CHAPTER 4

EXPLORATORY DATA ANALYSIS

4.1 Data set

All-The-News 2.0 is a collection of 2.6 million news articles from US based news sites such as Reuters, The New York Times and CNN.(Thompson, n.d.) It also includes tabloids and magazines such as TMZ and Vox within it's data. The news articles are scraped from news sites using python webcrawlers and is free to be downloaded and used for analysis outside of use for training generative AI. (Thompson, n.d.) The data is compiled from 26 news sites with the majority of data originating from Reuters. A huge amount of articles are on the shorter side being less than 22479 characters long.

The initial data contained 10 columns detailing the data, year, month, day, author,

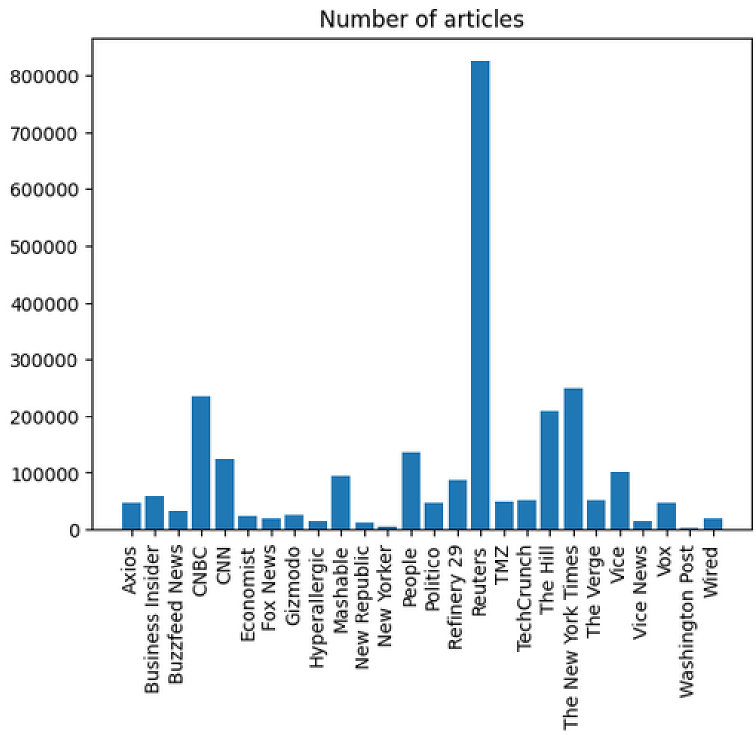

Figure 4.1 Data article count sorted by publication.

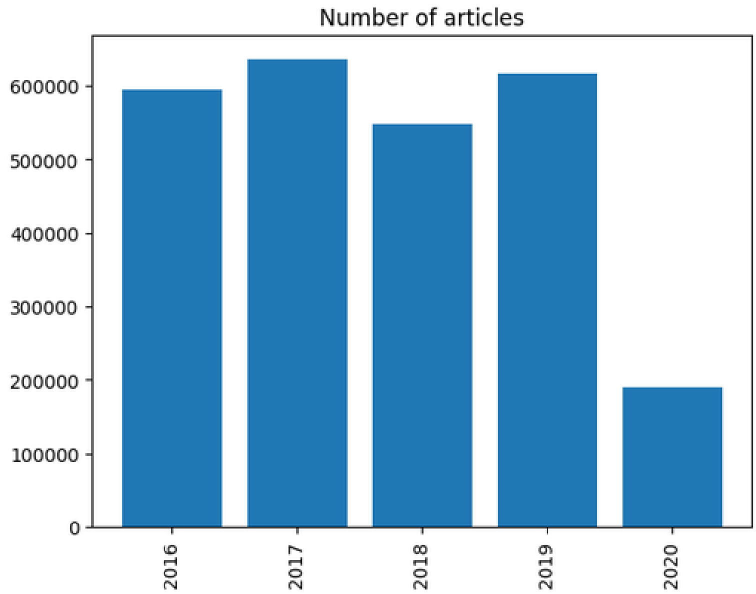

Figure 4.2 Data article count sorted by year.

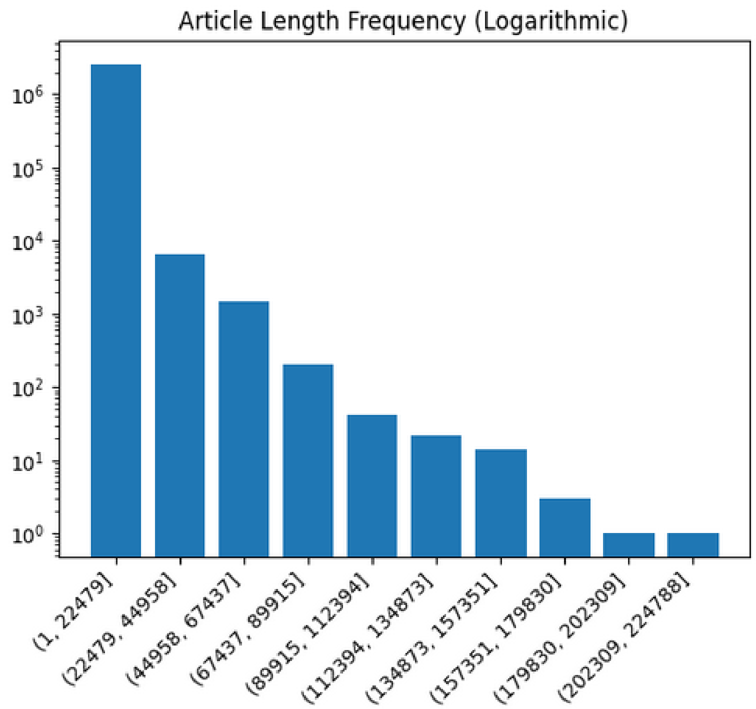

Figure 4.3 Frequency of articles by amount of characters. (Logarithmic scaling)

title, article text, url, publication section and publication sites. It includes news detailing current events and politics like Reuters and also tabloids/pop culture and tech magazines like TMZ or TechCrunch which the latter is not relevant to the thesis. The median text length is different across each news sites, with the New York Times, CNN, Vox and The New Yorker being the longest while Axios, Wired, Reuters and TMZ being the shortest. It's important to note that the length of articles are subject to large variances, the articles collected come from a large variety of sources that are not strictly political news and different journalist may have different methods of writing. Vox magazine and Vice magazine are both publications that contains many articles detailing global politics but also contains articles on opinions on pop culture.

Out of the 26 news sites, 10 of these sites focus more on pop culture and technologies rather than politics and were excluded. (Business Insider, Gizmodo, Hyperallergic, Mashable, People, Refinery 29, TMZ, TechChrunch, The Verge, Wired) Axios is also excluded because the structure of the publication summarizes news into short lists and sentences and is not suitable for the scope of the project.

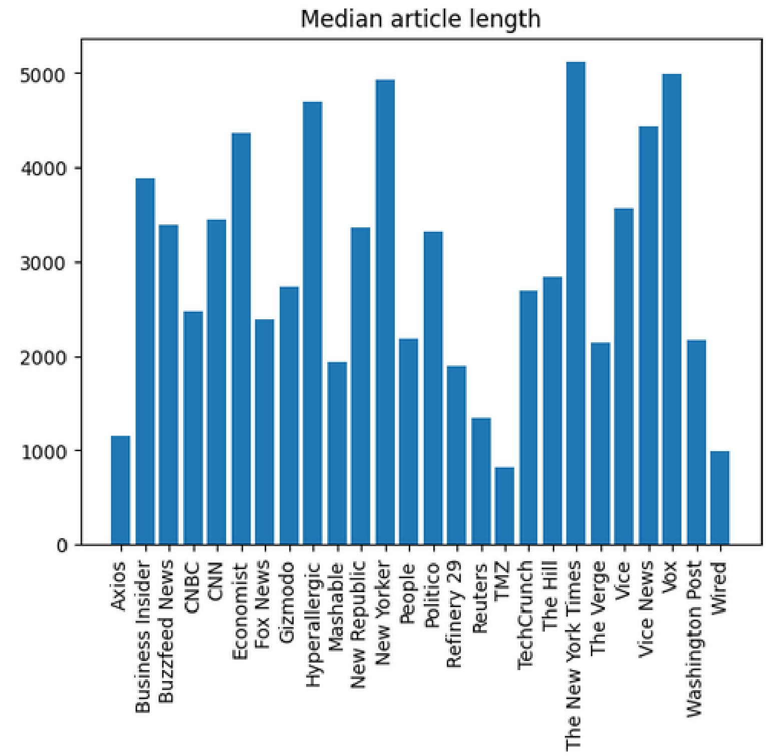

Figure 4.4 Median Article length by Publication.

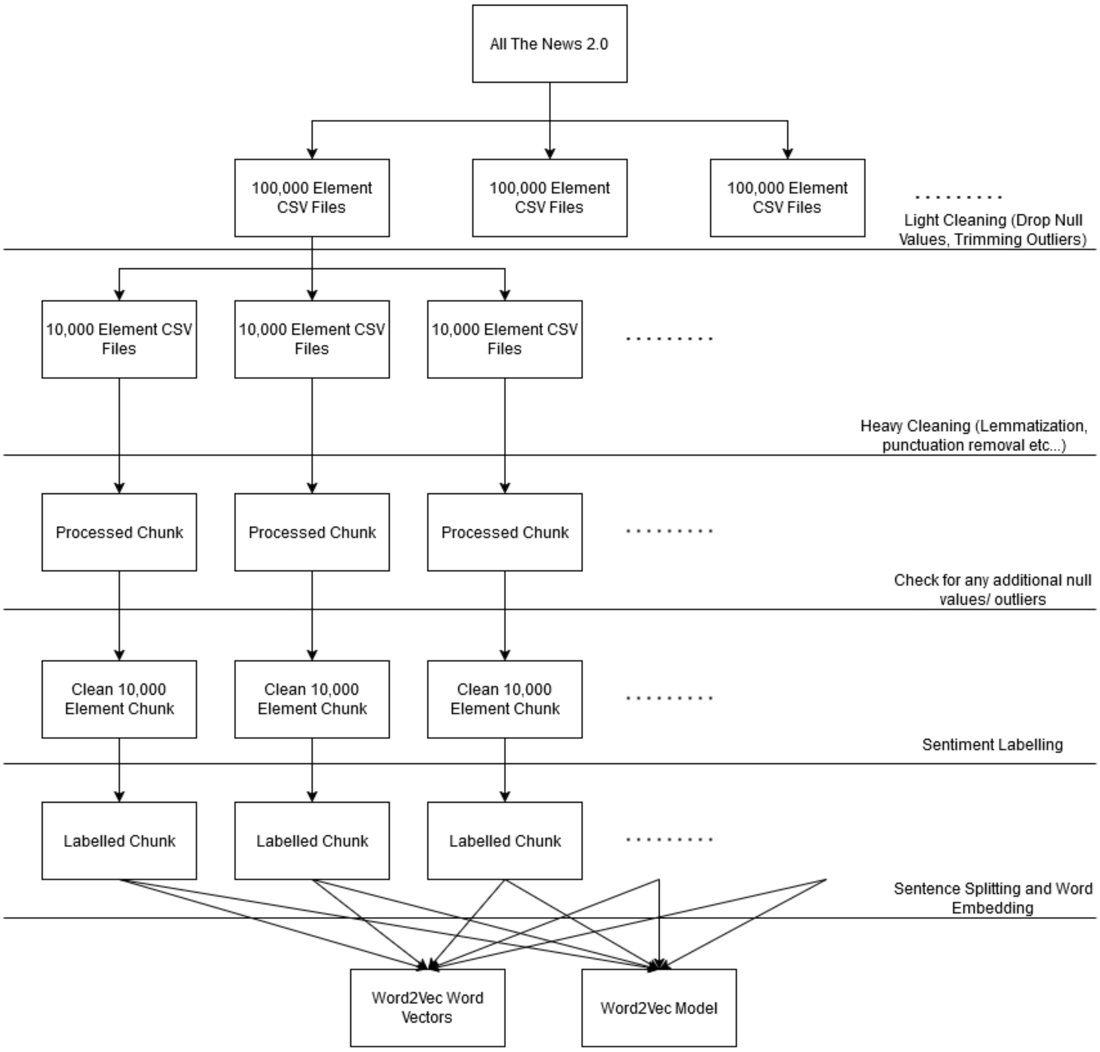

Figure 4.5 Data Cleaning process.

4.2 Data Cleaning

In order to compensate for computing limitations, the data was split into 100,000 sized chunks before processing lighter data cleaning processes like finding null data, then split into 10,000 sized chunks for deeper text cleaning. The data was reduced to only the year, article text and publication. Then, null values within the publication and article text were dropped. If the text length was less than the 1% quartile of its current chunk, it would be considered too short for use and removed. 1% quartile text length is a tiny amount of the data and most chunks have a 1% quartile text length between 200-250 characters. The text is then processed for more rigorous

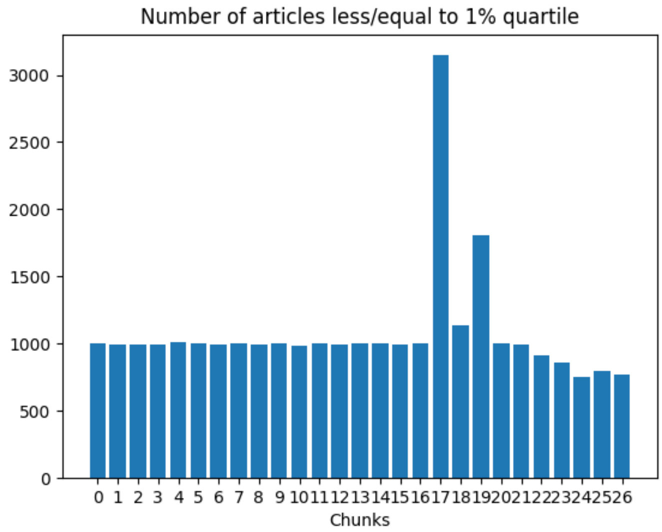

Figure 4.6 Amount of articles less than 1% quartile in text length.

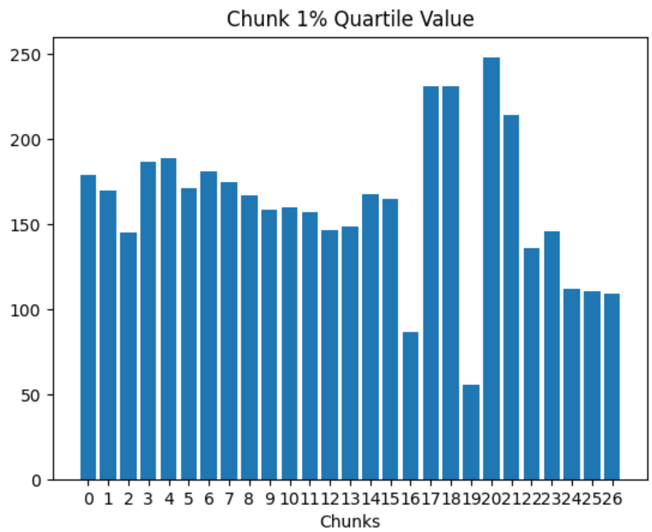

Figure 4.7 1% quartile character length for each chunk

data cleaning. The individual 100,000 sized chunks are split into 10,000 sized chunks to overcome memory limitations and goes through these removal processes in order:

1. Byte Order Mark (BOM, \\ufeff) removal
2. Duplicate word removal
3. Hyperlink removal

4. Expanding Contractions
5. Removing Non-Ascii/Non-Latin characters
6. Removing Punctuations
7. Convert to Parts of Speech (POS) Tags
8. Lemmatization
9. Stop Word Removal
10. Transform to lowercase
11. Remove POS tags
12. Split into individual sentences

Most methods requires the use of regular expressions (Regex, a form of string matching language) on some level and in total required more than 13 hours of computing to completely clean out the data.

4.2.1 BOM Removal

The first method is removing the Byte Order Mark (BOM) which is a character that signals how Unicode should handle the text either in Big-Endian or Little-Endian. By externally marking this, bytes and characters are properly rendered. This is an invisible character to rendered text and is usually handled but can be annoying for the processing of raw text. (Unicode, 2024)

4.2.2 Duplicate word removal

Duplicated words are removed via the Regex function: re.sub(”\\b(\w+)(\s+\\1)+\\b”,”\\1”,str) which replaces any instance of two words (”data data”, ”sing-sing”, ”waka waka”) into one word (”data”, ”sing”, ”waka”) this can get rid of words in languages that rely on reduplication to modify meanings like Austronesian Languages or some proper nouns but operating on mostly English articles, this is not as much of an issue.

4.2.3 Hyperlink Removal

Hyperlinks are addresses to websites and may distract from the training and labelling processes. The following Regex is used to detect potential hyperlinks and remove them: ”\S+\.\S+\/?\w+.”. This also detects words that have .net or .org in them which can either be potential website links or organization names that use the styling of domain names for aesthetic/branding reasons.

4.2.4 Expanding Contractions

Contractions are expanded using Regex substitution. The table below details common contractions in the English language and it’s replacement. ”’s” is not

Original Word	Expansion
”won’t”	”will not”
”can’t”	”can not”
”-n’t”	” not”
”-’re”	” are”
”-’s”	””
”-’d”	” would”
”-’ll”	” will”
”-’t”	” not”
”-’ve”	” have”
”-’m”	” am”

expanded into any word because it two major ways of expanding based on it’s syntactic part of speech, either as ”’s” to denote possession of a noun or as a truncation of is. This project will not differentiate between the two and instead removes the ”’s” entirely under the assumption that most occurrences of ”’s” as ”is” appears before a verb (most cases of verbs followed by ”’s” end in the easily identifiable suffix ”-ing”) or adverb (like here, there, up, yesterday, most, etc...) that can easily identified. The possessive

”’s” also frequently appears before a noun. Therefore, the project assumes there to have no significant reason to expand ”’s” instead of deleting it.

4.2.5 Removal of Non-Ascii Characters

The corpus is mostly in English. Any Non-Ascii character is removed to ensure that we are working in English. Individual characters instead of words are removed as most of these words are proper nouns like names and places in places that don’t use English as a first language, the word is still identified and preserved as a proper nouns during POS-tagging. The Regex used is ”[ˆ\x00-\x7F]”, which checks for all non-Ascii letters.

4.2.6 Removal of Punctuations and Numbers

Punctuations are used to denote a modification to a word or sentence in English but misplaced punctuations can be a distraction to the analysis, especially numbers where there sentiment value requires more context than the surrounding text. Due to this, it is often best to remove as much punctuation and numbers as possible. This project uses a multi-step Regex substitution to eliminate punctuations (” ” Indicates Regex matching string used):

1. Eliminate decimal full stops (1.2, 3.14,...) ”\d+\.\d+”
2. Numeral suffixes (-th, -k, -M,...) ”(\d+\w+)”
3. Duplicated !, ? or . (!!,...) ”\!{2,}”,”\?{2,}”,”\.{2,}”
4. Hyphens ”-”
5. Every non-alphabet except !,?,. ”[ˆa-zA-Z\s\.\!\?]”

The project still retains at least one ! ? or . as to split the individual sentences later.

4.2.7 Part Of Speech(POS) Tagging

POS tags are used to denote the syntactic category such as verbs, nouns and adverbs along with sub-categories like proper nouns. POS tags originated from the expansion of the work of Noam Chomsky's Syntactic Structures (Chomsky, 1957) and have been in wide use for NLP methods due to it's logical construction of sentences. This step is important for lemmatization as the lemmatized word is dependent on

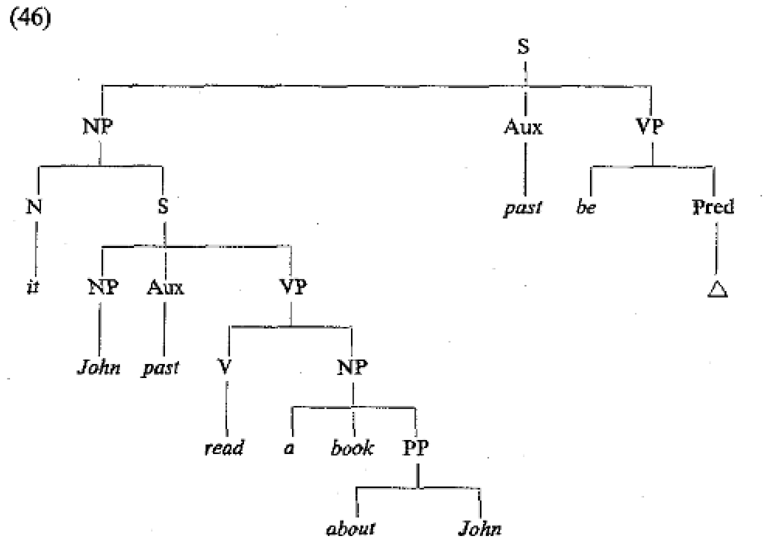

Figure 4.8 Example of a syntax tree deconstructing a sentence into it's POS categories (Chomsky, 1970)

context it has on the sentence. The sentence is first tokenized with WordTokenizer in NLTK, then converted into a tuple that represents both the word and it's POS tag using NLTK.

4.2.8 Lemmatization

Lemmatization is the act of converting a word into it's root word. This differs from stemming which removes the suffixes and prefixes. By converting a word into it's root, the vocabulary of a NLP model is reduced and any extra meaning that could be carried within the word through it's suffixes and prefixes is removed leaving only analysis of the root word. However, not all words should be stemmed/lemmatized as some non-root words may contain crucial information being in a certain POS category. As such the POS tag is required for this section.

4.2.9 Stop-word Removal

Stop-words are common words used within a language such as ”I”, ”You”, ”There”, ”Is”, ”Will”, ”Most” and so on. These words are largely uniformative to textual analysis and can often be removed. Important to note that the stop-words used excluded negations and quantifiers like ”not”, ”some” and ”most” as they are important in understanding the sentiment of a sentence. These stop words are excluded from removal. From the NLTK stop word list, they are depicted in the figure below. Articles that contain less than 25 words (outside of stop-words) are removed as they

```
["not","nor","no","few","some","more","most","all","can","will","don","don't","should","should've","now","ain","aren",\
 "aren't","couldn","couldn't","didn","didn't","doesn","doesn't","hadn","hadn't","hasn","hasn't","haven","haven't","isn",\
 "isn't","mightn","mightn't","mustn","mustn't","needn","needn't","shan","shan't","shouldn","shouldn't","wasn","wasn't",\
 "weren","weren't","won","won't","wouldn","wouldn't"]
```

Figure 4.9 Stop words that were excluded from removal.

may contain data that is not useful enough for analysis or are null after stop word removal.

4.2.10 Lowercase and POS tag removal

After the text is converted to lowercase, the pos tags are removed as the labelling method used (TextBlob) does not rely on POS tagging.

4.2.11 Sentence Splitting

Using the remaining !,? and . Each article is split into individual sentences to prepare the data for labelling and word embedding. A number is assigned to each sentence as to indicate which article it originated from.

After this second round of cleaning, 815,963 articles were removed which constitutes 30.345% of the original data.

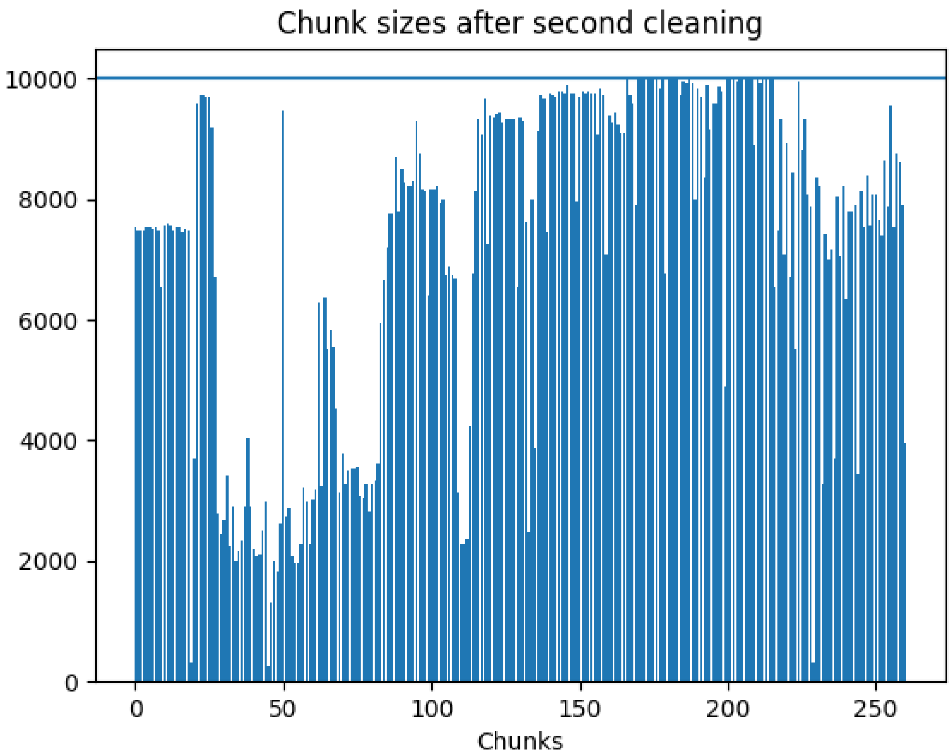

Figure 4.10 Chunk sizes after cleaning with line indicating the original 10,000 articles per second order chunk.

4.3 Rules-based Sentiment Analysis

4.3.1 Word Frequencies

Here are the most frequent nouns, verbs and adjectives present within the data: Much of the news focused around the Republican Party of the United States and

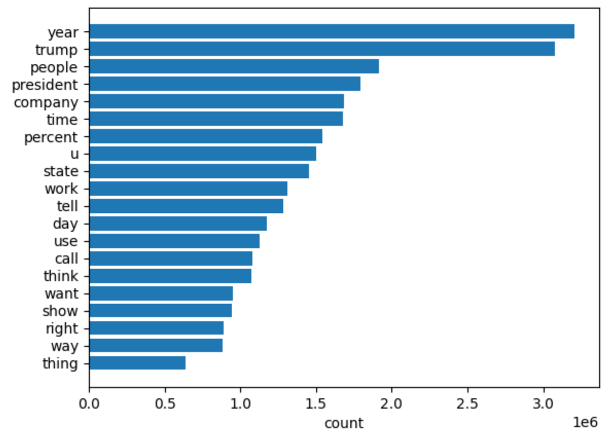

Figure 4.11 Frequently used nouns within the AllTheNews 2.0 Dataset. (U represents the abbreviation for United States of America, U.S. which was erroneously truncated to u during data cleaning.

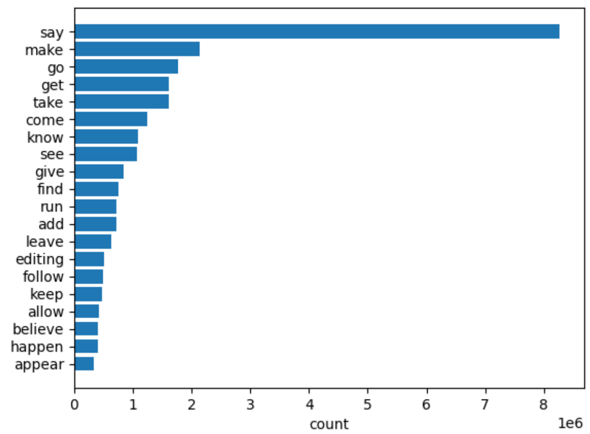

Figure 4.12 Frequently used verbs within the AllTheNews 2.0 Dataset..

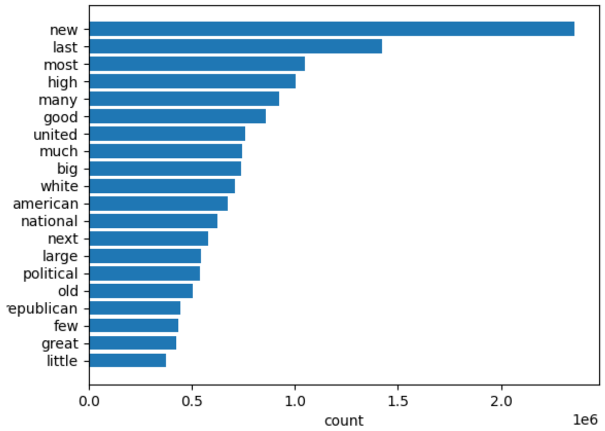

Figure 4.13 Frequently used adjectives within the AllTheNews 2.0 Dataset.

particular interest was garnered around (as of writing) President-Elect Donald J. Trump and his family with his surname appearing 2nd in most frequent nouns. A lot of the adjectives used are modifiers like high, most, great and big, some of which can be tagged as having positive polarity.

4.3.2 Text Labelling

The data is labelled using TextBlob. TextBlob does not require POS Tags in order to classify sentences which is why the POS tags are removed and the data is rewritten to contain individual sentences rather than paragraphs. The subjectivity score and objectivity score of the sentence is stored along with the original article index. Any empty fields that may have been produced from the sentence expansion are removed.

From the corpus, the polarity scores are labelled in the range between -1 to 1, where 0 is neutral, -1 is highly negative and 1 is positive, while the subjectivity score is labelled between 0 to 1, where 0 is most objective and 1 is most subjective. From the text-blob labelling, the sum sentiment score in each sentence and total sentence count per publication is collected and it's average is calculated by dividing the sum score with the total sentence count. Any objective sentences (that is subjective score == 0) is removed to emphasize the effects that non-objective sentences have. Mean polarity

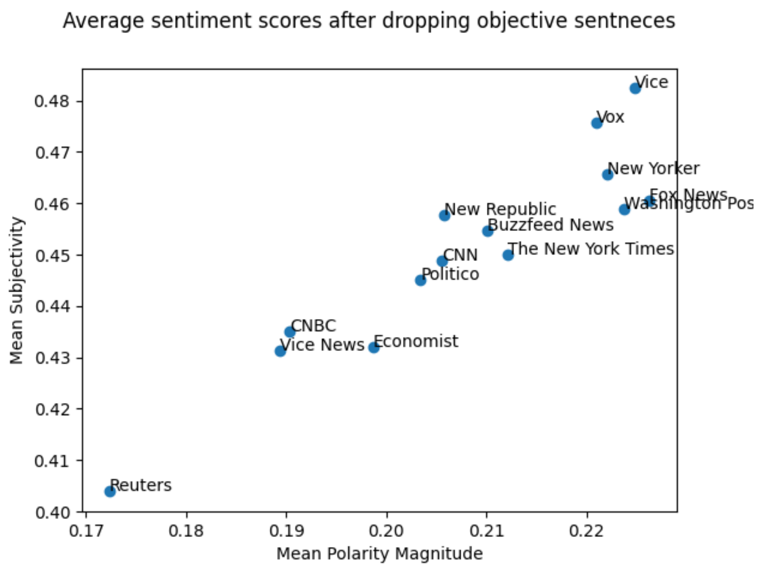

Figure 4.14 Average sentiment score per publication after accounting for objective sentences. The strong correlation between polarity magnitude and subjectivity scores is expected

magnitude, is also collected by making all polarity positive.

$$N = \text{sentence\_count} \tag{4.1}$$

$$\text{average\_polarity} = \frac{\sum_{i=0}^{N} \text{sentence\_polarity}_i}{N} \tag{4.2}$$

$$\text{average\_subjectivity} = \frac{\sum_{i=0}^{N} \text{sentence\_subjectivity}_i}{N} \tag{4.3}$$

$$\text{average\_polarity\_magnitude} = \frac{\sum_{i=0}^{N} \mid \text{sentence\_polarity}_i \mid}{N} \tag{4.4}$$

Reuters is noted to have the most objective news as it is also most neutral sounding news outlet. The second most objective publication is Vice News which is stark difference to Vice magazine articles which have high subjectivity scores. Publications like Vice, Vox and The Washington Post have higher scores may be due to them having more opinion pieces and pop cultural news rather than focusing on solely politics. The Spearman's correlation coefficient between mean polarity magnitude and subjectivity is a strong $r = 0.93263013$ which is expected given that objectivity is measured by how neutral a sentence is. Analysing the polarity magnitude, most news publications maintain less than 0.25 in terms of polarity scores which means that most sentences in news articles maintain more neutral language. Analyzing the ratio between mean

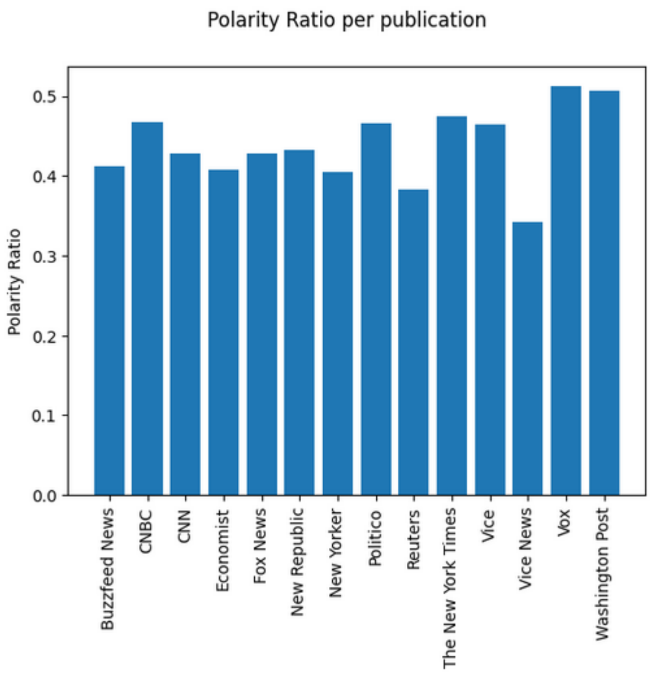

Figure 4.15   Polarity ratios by publication.

polarity scores and polarity magnitude, the mean polarity is only half of the polarity magnitude and are positive. This means that most publications between 2016 and 2020 have a positive bias in their sentences and are on average around 44% more positive. This may be explained by the usage of quantitative adjectives like high or great which can be tagged as positive. Polarity ratio is given by the following equation:

$$\text{polarity\_ratio} = \frac{\text{average\_polarity}}{\text{average\_polarity\_magnitude}} \tag{4.5}$$

4.4 Media Bias Labelling

The media bias labels are manually annotated using AllSides (?, ?) (Accessed 11/1/2025), a service that quantifies media bias as left(negative)/center(zero)/right(positive) leaning based on analysis of media publications and community feedback. The labels are applied to the publications and squared to emphasize measure the degree of bias they have. Publications containing more than one section within AllSides are averaged out. (Such as CNN, which has scores for CNN Opinions, CNN Business and CNN Digital are averaged out). Vice news and Vice magazine are treated as a singular entity in AllSides and thus have the same media bias score applied. Most of the publications outside of Fox News are labelled as either center or left leaning according to AllSides

When plotted onto a scatter plot there is a clear correlation between sentiment scores

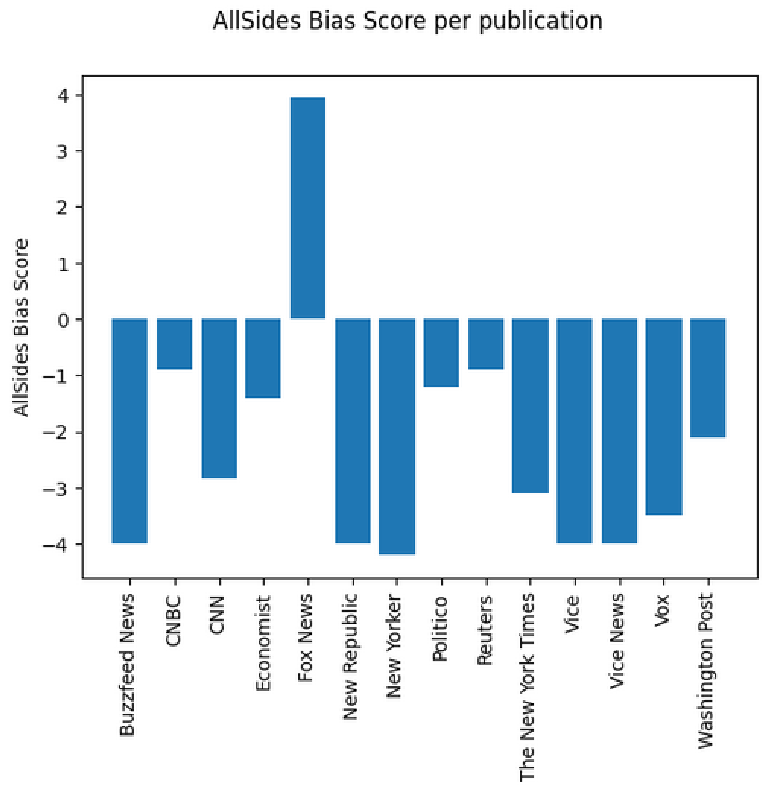

Figure 4.16 Bias scores provided by AllSides. (Negative indicates left leaning; Positive indicates right leaning;)

and media bias scores. The higher the squared media score, the more subjective and polarizing the journalism is, outside from the outlier of Vice News (As it shares the same bias label as the more opinionated Vice Magazine) and the extremely objective and neutral journalism of Reuters.

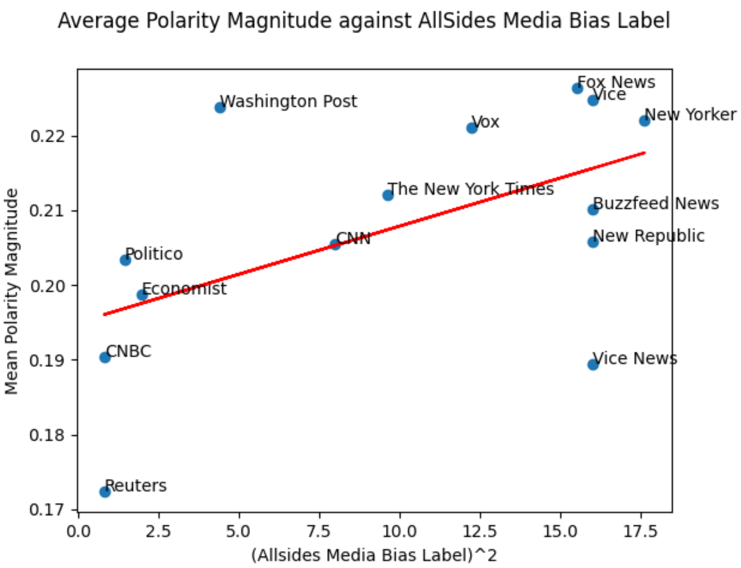

Figure 4.17 Average polarity magnitude score per publication against AllSides Media Bias label. Regression line provided ($r = 0.53845958$)

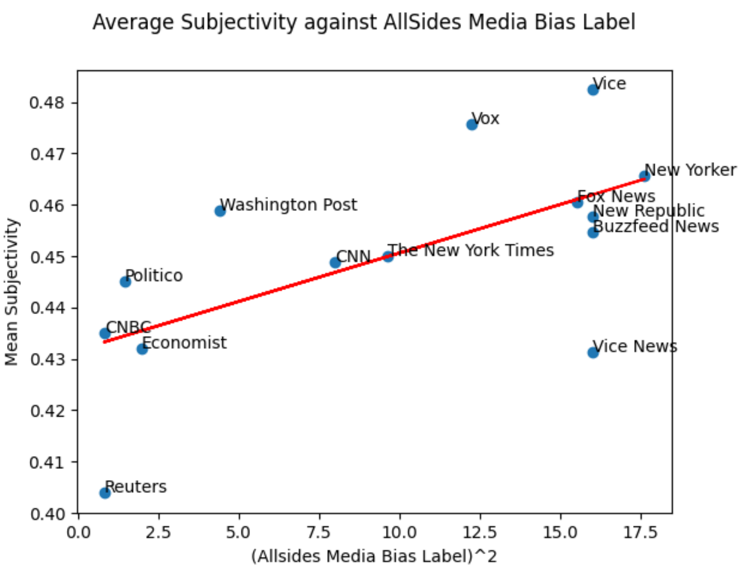

Figure 4.18 Average subjectivity score per publication against AllSides Media Bias label. Regression line provided ($r = 0.6250412$)

4.5 Word Embedding

Words are embedded using a simple Word2Vec model trained using the All-The-News 2.0 data using the GenSim library. Word2Vec is a duo of simple machine learning models both comprising of one hidden layer, continuous bag-of-words (CBOW) and skip-gram. By default, gensim uses CBOW with the use of the sum of word vectors during training. The model produces vectors of size 300 and takes account of 4 surrounding words when training on a specific word (window size 5). The model also ignores words that appear less than 3 times throughout the corpus. Due to memory limitations the model cannot train on the whole corpus at once, but rather must train using the 10,000 sized chunks. The total training epochs is 1. All the word vectors and model files are saved for future use. The embeddings are visualized

```
model_file = Path(f"./word2vec2.model")
model = None
if model_file.is_file():
    model = Word2Vec.load("word2vec2.model")

for file in file_paths:
    print(file)
    df = pd.read_csv(file)
    sentences = df['sentences'].progress_apply(word_tokenize)
    if model == None:
        model = Word2Vec(sentences = sentences, vector_size=300, window=5, min_count=1, workers=2)
    else:
        model.build_vocab(sentences,update=True)
        model.train(sentences,total_examples=len(df['sentences']),epochs = 1)
    model.save("word2vec2.model")
```

Figure 4.19 Training algorithm.

using t-distributed stochastic neighbor embedding (tsne), a dimensionality reduction technique that maps non-linear data onto 3 dimensional/2 dimensional maps. (Belkina et al., 2019) A 2 dimensional map is produced using tsne. A vast majority of words are clustered in a large blob with smaller oblong clusters. There are several smaller clusters that indicate rarer but highly interconnected words. There are many words that fall between clusters possibly due to these words appearing in a large amount and variety of sentences that can fall in either clusters. A 3d mapping may reveal more detailed clusters but due to limitation within this thesis, cannot be made.

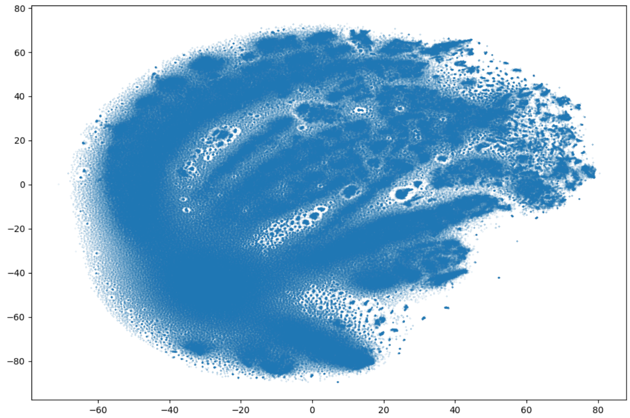

Figure 4.20 2d mapping of the Word2Vec embedding results.

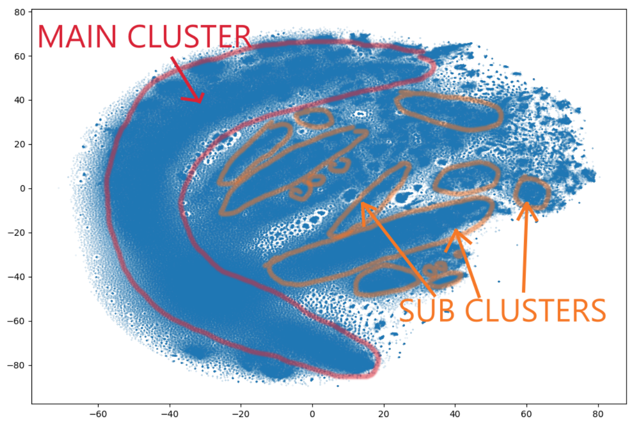

Figure 4.21 Highlighted clusters. There is a clear main cluster that wraps around the sub clusters, much of the words fall into several of these clusters. Only some of the more clearer sub clusters are highlighted. A significant amount of words also fall in between clusters possibly due to the variety of the sentences that feature such words.

CHAPTER 5

CONCLUSIONS

5.1 Summary and Future Work

The study introduced it's goals for uncovering media bias and propaganda with the usage of Bidirectional Recurrent Neural Networks (BiRNN). The study was done under the assumption that rules-based techniques is insufficient in detection of finer patterns within news and media and inspired by the recent success of utilizing BiRNNs models for Natural Language Processing. Using rules-based sentiment analysis, the study was able to get an general overview of the corpus and was able to draw correlations between media bias and sentiment scores.

From the initial findings in the exploratory data analysis, there is a linear correlation between media bias scores from AllSides and sentiment scores in both the polarity and subjectivity areas. Using TexBlob sentiment labelling, the study was able to account for surrounding word contexts within a particular sentence and identify sensational and opinionated language within news media. The initial findings has demonstrated a moderate linear relationship between the use of sensational/opinionated language and sentences and AllSides media bias scores, which emphasizes how emotional and opinionated coverage plays an important part in media bias. In addition to the initial findings, the data was embedded for future analysis using BiLSTM and BiGRU models.

This project only targets the possibility that deep learning driven sentiment analysis has on the identification of media bias. Further research can be done using a more finer grained labeller that specifically targets the form of propaganda technique and media bias that is present within the sentences of news articles like in the study done by (Da San Martino et al., 2019).

5.2 Limitations

The thesis was limited in time and computation. The authors did not have sufficient time in doing a more comprehensive literature review and as mentioned in the Exploratory Data Analysis section, needed to split the data into smaller chunks in order to handle computation of the data. Sentiment Analysis is not sufficient in analysing the overall biases of a news media outlet, further research could be done in the coverage of news and more specific tactics of propaganda utilized in media.

References

AbuSa'aleek, A. O. (2015). Internet linguistics: A linguistic analysis of electronic discourse as a new variety of language. *International journal of English linguistics*, *5*(1), 135.

Alhajeri, M. S., Ren, Y. M., Ou, F., Abdullah, F., & Christofides, P. D. (2024). Model predictive control of nonlinear processes using transfer learning-based recurrent neural networks. *Chemical Engineering Research and Design*, *205*, 1–12.

Allsides. (n.d.). Retrieved 21-12-2024, from `https://www.allsides.com`

Alonso, M. A., Vilares, D., Gómez-Rodríguez, C., & Vilares, J. (2021). Sentiment analysis for fake news detection. *Electronics*, *10*(11), 1348.

Alshutayri, A., Alghamdi, A., Nassibi, N., Aljojo, N., Aldhahri, E., & ABOULOLA, O. (2022). Sentiment analysis for arabic tweet about the covid-19 worldwide epidemic. *Romanian Journal of Information Technology and Automatic Control*, *32*(2), 127–136.

Andrychowicz, M., Denil, M., Gomez, S., Hoffman, M. W., Pfau, D., Schaul, T., ... De Freitas, N. (2016). Learning to learn by gradient descent by gradient descent. *Advances in neural information processing systems*, *29*.

Aronoff, M., & Rees-Miller, J. (2020). *The handbook of linguistics*. John Wiley & Sons.

Balahur, A., Steinberger, R., Kabadjov, M., Zavarella, V., Van Der Goot, E., Halkia, M., ... Belyaeva, J. (2013). Sentiment analysis in the news. *arXiv preprint arXiv:1309.6202*.

Baly, R., Martino, G. D. S., Glass, J., & Nakov, P. (2020). We can detect your bias: Predicting the political ideology of news articles. *arXiv preprint arXiv:2010.05338*.

Belkina, A. C., Ciccolella, C. O., Anno, R., Halpert, R., Spidlen, J., & Snyder-Cappione, J. E. (2019). Automated optimized parameters for t-distributed stochastic neighbor embedding improve visualization and analysis of large datasets. *Nature communications*, *10*(1), 5415.

Bengio, Y., Ducharme, R., & Vincent, P. (2000). A neural probabilistic language model. *Advances in neural information processing systems*, *13*.

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, *5*(2), 157–166.

Chaudhary, L., Girdhar, N., Sharma, D., Andreu-Perez, J., Doucet, A., & Renz, M. (2023). A review of deep learning models for twitter sentiment analysis: Challenges and opportunities. *IEEE Transactions on Computational Social Systems*.

Chomsky, N. (1957). *Syntactic structures*. Mouton de Gruyter.

Chomsky, N. (1970). Remarks on nominalization. *Readings in English transformational grammar*.

Da San Martino, G., Seunghak, Y., Barrón-Cedeno, A., Petrov, R., Nakov, P., et al. (2019). Fine-grained analysis of propaganda in news article. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 5636–5646).

DiMaggio, P., Evans, J., & Bryson, B. (1996). Have american's social attitudes become more polarized? *American journal of Sociology*, *102*(3), 690–755.

Eyvazi-Abdoljabbar, S., Kim, S., Feizi-Derakhshi, M.-R., Farhadi, Z., & Abdu-lameer Mohammed, D. (2024). An Ensemble-Based Model for Sentiment Analysis of Persian Comments on Instagram Using Deep Learning Algorithms. *IEEE ACCESS*, *12*, 151223–151235.

Farabi, S., Ranasinghe, T., Kanojia, D., Kong, Y., & Zampieri, M. (2024, August). A survey of multimodal sarcasm detection. In *Proceedings of the thirty-thirdinternational joint conference on artificial intelligence* (p. 8020–8028). International Joint Conferences on Artificial Intelligence Organization.

Frenda, S., et al. (2018). The role of sarcasm in hate speech. a multilingual perspective. In *Proceedings of the doctoral symposium of the xxxivinternational conference of the spanish society for natural language processing (sepln 2018)* (pp. 13–17).

Halawani, H., Mashraqi, A., Badr, S., & Alkhalaf, S. (2023). Automated sentiment analysis in social media using Harris Hawks optimisation and deep learning techniques. *Alexandria Engineering Journal*, *80*, 433–443. (Publisher: Elsevier B.V.)

Hindocha, E., Yazhiny, V., Arunkumar, A., & Boobalan, P. (2019). Short-text semantic similarity using glove word embedding. *International Research Journal of Engineering and Technology*, *6*(4).

Hochreiter, S. (1997). Long short-term memory. *Neural Computation MIT-Press*.

Islam, M., Chen, G., & Jin, S. (2019). An overview of neural network. *American Journal of Neural Networks and Applications*, *5*(1), 7–11.

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets*, *31*(3), 685–695.

Kastrati, M., Kastrati, Z., Shariq Imran, A., & Biba, M. (2024). Leveraging distant supervision and deep learning for twitter sentiment and emotion classification. *Journal of Intelligent Information Systems*, *62*(4), 1045–1070. (Publisher: Springer)

Kong, D., Chen, A., Zhang, J., Xiang, X., Lou, W. V., Kwok, T., & Wu, B. (2022). Public discourse and sentiment toward dementia on chinese social media: Machine learning analysis of weibo posts. *Journal of medical Internet research*, *24*(9), e39805.

Lakretz, Y., Kruszewski, G., Desbordes, T., Hupkes, D., Dehaene, S., & Baroni, M. (2019). The emergence of number and syntax units in lstm language models. *arXiv preprint arXiv:1903.07435*.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436–444.

Li, C., & Li, F. (2023). Emotion recognition of social media users based on deep learning. *PeerJ Computer Science*, *9*, e1414.

Liddy, E. D. (1998). Enhanced text retrieval using natural language processing. *Bulletin of the American Society for Information Science and Technology*, *24*(4), 14–16.

Lin, W., Zhang, Q., Wu, Y., & Chen, T.-C. (2023). Running a Sustainable Social Media Business: The Use of Deep Learning Methods in Online-Comment Short Texts. *Sustainability (Switzerland)*, *15*(11). (Publisher: MDPI)

Lubis, A. R., Nasution, M. K., Sitompul, O. S., & Zamzami, E. M. (2021). The effect of the tf-idf algorithm in times series in forecasting word on social media. *Indones. J. Electr. Eng. Comput. Sci*, *22*(2), 976.

Ma, L., & Zhang, Y. (2015). Using word2vec to process big text data. In *2015 ieee international conference on big data (big data)* (pp. 2895–2897).

Mahadevaswamy, U., & Swathi, P. (2023). Sentiment analysis using bidirectional lstm network. *Procedia Computer Science*, *218*, 45–56.

Mateus, B. C., Mendes, M., Farinha, J. T., Assis, R., & Cardoso, A. M. (2021). Comparing lstm and gru models to predict the condition of a pulp paper press. *Energies*, *14*(21), 6958.

Maynard, D. G., & Greenwood, M. A. (2014). Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Lrec 2014 proceedings.*

Mikolov, T. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, *3781*.

Mu, G., Li, J., Li, X., Chen, C., Ju, X., & Dai, J. (2024). An Enhanced IDBO-CNN-BiLSTM Model for Sentiment Analysis of Natural Disaster Tweets. *Biomimetics*, *9*(9). (Publisher: Multidisciplinary Digital Publishing Institute (MDPI))

Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, *18*(5), 544–551.

Németh, R. (2023). A scoping review on the use of natural language processing in research on political polarization: trends and research prospects. *Journal of computational social science*, *6*(1), 289–313.

Nguyen, H.-H. (2024). Enhancing Sentiment Analysis on Social Media Data with Advanced Deep Learning Techniques. *International Journal of Advanced Computer Science and Applications*, *15*(5), 970–980. (Publisher: Science and Information Organization)

Noh, S.-H. (2021). Analysis of gradient vanishing of rnns and performance comparison. *Information*, *12*(11), 442.

Orellana, S., & Bisgin, H. (2023). Using natural language processing to analyze political party manifestos from new zealand. *Information*, *14*(3), 152.

Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. *arXiv preprint arXiv:1211.5063*.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)* (pp. 1532–1543).

Rodrigo-Ginés, F.-J., Carrillo-de Albornoz, J., & Plaza, L. (2024). A systematic review on media bias detection: What is media bias, how it is expressed, and how to detect it. *Expert Systems with Applications*, *237*, 121641.

Rodríguez-Ibánez, M., Casánez-Ventura, A., Castejón-Mateos, F., & Cuenca-Jiménez, P.-M. (2023). A review on sentiment analysis from social media platforms. *Expert Systems with Applications*, *223*, 119862.

Saberi, B., & Saad, S. (2017). Sentiment analysis or opinion mining: A review. *Int. J. Adv. Sci. Eng. Inf. Technol*, *7*(5), 1660–1666.

Said, H., Tawfik, B., & Makhlouf, M. (2023). A Deep Learning Approach for Sentiment Classification of COVID-19 Vaccination Tweets. *International Journal of Advanced Computer Science and Applications*, *14*(4), 530–538. (Publisher: Science and Information Organization)

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, *45*(11), 2673–2681.

Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval* (Vol. 39). Cambridge University Press Cambridge.

Shewalkar, A., Nyavanandi, D., & Ludwig, S. A. (2019). Performance evaluation of deep neural networks applied to speech recognition: Rnn, lstm and gru. *Journal of Artificial Intelligence and Soft Computing Research*, *9*(4), 235–245.

Singh, C., Imam, T., Wibowo, S., & Grandhi, S. (2022). A Deep Learning Approach for Sentiment Analysis of COVID-19 Reviews. *Applied Sciences (Switzerland)*, *12*(8). (Publisher: MDPI)

Stavrianou, A., Andritsos, P., & Nicoloyannis, N. (2007). Overview and semantic issues of text mining. *ACM Sigmod Record*, *36*(3), 23–34.

Tabinda Kokab, S., Asghar, S., & Naz, S. (2022). Transformer-based deep learning models for the sentiment analysis of social media data. *Array*, *14*. (Publisher: Elsevier B.V.)

Taboada, M. (2016). Sentiment analysis: An overview from linguistics. *Annual Review of Linguistics*, *2*(1), 325–347.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational linguistics*, *37*(2), 267–307.

Tarwani, K. M., & Edem, S. (2017). Survey on recurrent neural network in natural language processing. *Int. J. Eng. Trends Technol*, *48*(6), 301–304.

Thompson, A. (n.d.). *All the news 2.0.* Retrieved 21-12-2024, from `https://components.one/datasets/all-the-news-2-news-articles-dataset/`

Unicode. (2024). *The unicode standard version 16.0 core specification.* Retrieved 10-1-2024, from `https://www.unicode.org/versions/Unicode16.0.0/core-spec/`

Vanam, H., & Raj, J. (2023). Novel Method for Sentiment Analysis in Social Media Data Using Hybrid Deep Learning Model. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, *32*(1), 272–289. (Publisher: Penerbit Akademia Baru)

Vernikou, S., Lyras, A., & Kanavos, A. (2022). Multiclass sentiment analysis on COVID-19-related tweets using deep learning models. *Neural Computing and Applications*, *34*(22), 19615–19627. (Publisher: Springer Science and Business Media Deutschland GmbH)

Wankhade, M., Rao, A. C. S., & Kulkarni, C. (2022). A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*, *55*(7), 5731–5780.

Wei, C., Wang, Y.-C., Wang, B., & Kuo, C.-C. J. (2023). An overview on language models: Recent developments and outlook. *arXiv preprint arXiv:2303.05759*.

Yin, Z., Shao, J., Hussain, M., Hao, Y., Chen, Y., Zhang, X., & Wang, L. (2023). DPG-LSTM: An Enhanced LSTM Framework for Sentiment Analysis in Social Media Text Based on Dependency Parsing and GCN. *Applied Sciences (Switzerland)*, *13*(1). (Publisher: MDPI)

Zakaria, M., Mabrouka, A., & Sarhan, S. (2014). Artificial neural network: a brief overview. *neural networks*, *1*, 2.

Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D., & Saeed, J. (2020). A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. *Journal of Applied Science and Technology Trends*, *1*(1), 56–70.