National College of Ireland

Co-pilot widget for assisting the public in processing US presidential political candidate tweets from Twitter in 2024 US elections candidate choice through sarcasm and stance detection

MSc Research Project
MSCAITOP

Arthur Ryan
Student ID: x23333138

School of Computing
National College of Ireland

Supervisor: Professor Sheresh Zahoor

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Arthur Ryan
Student ID:	2333138
Programme:	MSCAITOP **Year:** 2024
Module:	Practicum Part 2
Supervisor:	Professor Sheresh Zahoor
Submission Due Date:	12/8/2024
Project Title:	**"Co-pilot widget for assisting the public in processing US presidential political candidate tweets from Twitter in 2024 US elections candidate choice through sarcasm and stance detection"**
Word Count:	6851 (including references) **Page Count:** 19 of paper itself

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Arthur Ryan

Date: 29/7/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Co-pilot widget for assisting the public in processing US presidential political candidate tweets from Twitter in 2024 US elections candidate choice through sarcasm and stance detection

Student Arthur Ryan
x23333138
MSCAITOP - Masters in AI Research in Computing CA1
National College of Ireland

15/2/2024

Abstract

With 50 percent of the world's population going to the polls to elect new leaders in 2024 it was thought to be relevant to examine what can be done to lessen polarisation and improve quality of discourse online. The gap the paper intends to fill is the prior lack of combined natural language processing (NLP) techniques collected together to raise understanding of media. NLP results were 0.9960 for truth-fake detection, 0.9835 for sarcasm detection and 0.9978 for stance. Additionally, a set of two versions of graphic outcome to make understanding simpler and faster.

Keywords — NLP, truth-fake detection, sarcasm detection, stance detection, understanding communications, lower polarization

1 Contributions

The main contributions of the research are the following:

1. Improvement in detection of sarcasm to 0.9835 versus a benchmark of 0.97
2. Improvement in in detection of stance to 0.9978 versus a benchmark of 0.662
3. Faster understanding of presence of Fake-News, Sarcasm, Stance through two versions of a graphic but particularly the second graphic.

2 Introduction

The problem being examined is fake news, which is found by (Barthel et al. (2016)) in US presidential elections as damaging to opinion forming. Fake news if left unchallenged will lead to political outcomes, election choices, that are not reflective of the facts but instead are the result of manipulation via fake news. This fake news problem (Cantarella et al. (2023)) leads to polarization in online discussions. In turn, generating the need to create a computer aid that will assist readers while they read US political election tweets on Twitter / X Platform.

To solve this problem and enable consumers of online media to be more aware of the

manipulation in fake news, this paper uses text analysis and classification in AI and natural language programming (NLP) to better inform the reader of US election political tweets on Twitter / X Platform in order to reach a more balanced and informed opinion.

The solution in this paper proposes is to combine the techniques, truth-fake detection, sarcasm and stance detection, and a set of graphics. The result of the solution is to reduce time to understand and raise quality of understanding. Through the use of the models Logit Regression, Artificial Neural Network (ANN), Long Short erm Memory (LTSM), Gensim model's Word2Vec, Glove2Word2Vec, FastText, Truth-Fake news, Sarcasm, Stance in Twitter / X Platform tweets will be analysed.

For the data, Twitter / X Platform, reddit, academic reference database were the sources for collecting the data. The tweets were gathered using hashtags and keywords. They will be cleaned with preprocessing tools. Then lemmatization, stemming, tokenized will be applied. The tokenized data was used to create truth-fake, sarcasm and stance scores from the tweets. With the above context we are led to the research question.

Research Question.
The research question to answer is, can NLP test classification models, covering fake news, sarcasm and stance, plus graphic output, improve classification, awareness, and understanding in detecting fake news, sarcasm and stance in tweets on Twitter (X Platform) during the 2024 US presidential elections.

The remainder of the document is in the following structure. Firstly, a literature review, table of results of the literature review, broken down into subsections on sarcasm, stance, polarization. Following the literature review is the research niche, then the research methodology, design specifications, network diagram, resource datasets, evaluation, discussion, generalisation, conclusion, future work, project plan, finally the bibliography.

3 Related Work

In the review of the literature I found (Chebolu et al. (2022)) using an aspect-based sentiment analysis (ASBA) NLP technique to identify the targets of media posts, using the aspect family, and sentiment. (Chebolu et al. (2022)) created a database filling a gap in the field in the ASBA field while overviewing the existing 98 databases across 25 domains available, with data gathering approaches that were listed with their advantages and disadvantages. The authors' advocacy for standardisation is helpful in the research question of my own paper. Further combining datasets could be used to extend my own research. Through expanding on the above work in to more applications better sentiment values can be achieved in the new domain. Similar to (Yue et al. (2019)) and (Kowsari et al. (2019)) the authors call for more comprehensive methodologies and diverse datasets for sentiment and text classification. (Chebolu et al. (2022)) found a research gap in that the inter task connections need to be researched further and the source of the low performance observed needs to be determined. The authors speculate that it may be due to inherent complexity of the task, the element ambiguity, or the shortcomings in the data representation. Another gap is the limitation of many datasets being private and requiring requests to authors for permission for use. The authors call for both respect for copyrights and the greater sharing of the materials in the databases. Further related work examined more media specific applications.

(Gelles-Watnick Risa (2024)) Focusing on media, describe and determine the penetration level of media through the various branches of society. PEW measured internet and mobile phone ownership in the USA, finding 95 percent of adults have internet, 80 percent high speed

internet and 90 percent own a smart phone. This is relevant in ascertaining the relevance of the research question to current technology usage versus assuming, in a non-fact-based manner, that everyone has a channel from president of the United States (POTUS), Presidential Candidates to individual citizens in the US. Also relevant to my research is confirming anecdotal thoughts on broad smartphone and internet ownership. (Gelles-Watnick Risa (2024)) find the rate of access and ownership varies markedly across age bands, income of the household and across educational levels, if time permits this is an interesting layer to add to my own analysis.

3.1 Sarcasm

Sarcasm is defined as an act of conveying a viewpoint with an opposite emotion, advocating the opposite of what they mean. It is a communication technique used to express an emotion that is opposite or distinct from the literal meaning of the words being used, typically for satire, criticism, or amusement. Sarcasm detection is very important in the field of affective computing and sentiment analysis because expressions of sarcasm can reverse the polarity of sentences. Sarcasm, purely context-based and a common phenomena in social media, is inherently difficult to detect, which makes it sometimes difficult for humans to interpret.

Related to sarcasm, the paper (Sharma and Joshi (2024)) detects sarcasm in social media tweets using neural networks and machine learning (ML) models. Using logistic regression, Naive Bayes (NB) classifier, linear Support Vector Machine (SVM), decision tree, an ensemble classifier. The authors model achieved a significant accuracy around 0.75 to identify sarcasm in tweets. Full evaluation was done over the metrics precision, accuracy, recall, and the F-measures. Performance by model was Logistic regression 72.01 percent, Linear SVM 74.82 percent, Decision tree 61.47 percent, Naïve Bayes 70.98 percent, Ensemble classifier 71.87 percent

Leveraging features within sarcasm text, namely through feature extraction content related to sentiments and punctuation, the paper (Gupta et al. (2020)) uses chi-square tests to identify the useful features. Following the chi square feature identification two hundred top ranked features from tf-idf are extracted and combined with the sentiment related features to close in on the content in the tweet that is sarcastic. The authors found that voting classifier created the highest accuracy at 86.53 percent with SVM algorithms produced ranking in second place at 74.59 percent.

Sarcasm via pattern detection was explored by (Pawar and Bhingarkar (2020)) through exploring and mining Twitter / X Platform data. Using four sets of features, including many specific sarcasm examples, it was proposed to classify tweets as sarcastic and non-sarcastic. Models used are SVM, KNN and Random Forest (RF) classifiers with RF performing the best. Though the authors did not explore AI patterns of sarcastic detection to identify patterns.

Most of the existing works focused on using lexical features for identifying sarcasm. In the author's work, (Sundararajan (2021)), hyperbolic and pragmatic features are added to lexical feature extraction to identify sarcasm. Once extracted the authors developed a model to detect sarcasm based on stacking- based feature ensemble algorithms. Results are that feature-based approaches reached an accuracy of 62 percent, emotion-based features attained 59 percent, respectively. However, the proposed ensemble learning algorithm attains an overall accuracy of 83 percent, surpassing the feature set-based approaches.

In the next study (Arifuddin et al. (2019)) aims to detect the text of sarcasm in the language Bahasa. Data consist of 480 training data and 120 test data collected from Twitter / X Platform. Preprocessing and feature extraction stages were carried out. SVM was used to classify sarcasm and non-sarcasm sentences. Through comparing the accuracy of N-gram, part of speech(POS) Tag, Punctuation, Pragmatic experiments are ranked. The authors reached the highest accuracy of 91.6 percent with a precision of 92 percent with all features combined.

Sarcasm detection using Evidential deep learning was employed by the authors (Chy et al. (2023)) using uncertainty estimations for identifying the sentiments from news headlines dataset, with LSTM and GRU used with Evidential deep learning approach. The purpose of using LSTM is that it can classify texts from headlines in order to analysis the sentiments. The authors used GRU (gated recurrent unit), a recurrent neural networks (RNN), which models sequential data. GRU architecture and network is ideally suited for identifying dependencies and extended contextual relationships in news data.

Research on comparative analysis of multiclass classifiers was carried out by (Damaraju and Rao (2023)) for detecting irony and sarcasm in short texts. With rising use of social media, it has become important to develop accurate and efficient algorithms for detecting irony and sarcasm that is expressed implicitly in texts. The authors evaluated five multiclass classifiers, Naive Bayes, linear classifier, XG Boost, KNN, and SGD, using a variety of text representations, such as count vectors, word-level TF-IDF, and hash vectors.

Results showed that the linear classifier using word-level TF-IDF achieved the highest accuracy of 74.39 percent, while the KNN classifier using hash vectors achieved the highest accuracy of 75.16 percent. The authors found all classifiers had sensitivity to certain keywords and phrases, which indicates the need for further research to improve robustness of the classifiers. The study provides insights into the strengths and weaknesses of different multiclass classification approaches for detecting irony and sarcasm in short texts, which can guide future research in this field.

Relevant to my own paper on news flow, the authors deal particularly with sarcasm in the paper of (Bharti et al. (2022)) for detection of sarcasm in News Headlines. Bag of words is used in the analysis using term frequency and n-grams frequency along with voted classification. The authors compared different features-based approach and the experimental results generated by a voted classifier consisting of seven different classifiers.

Result metrics were accuracy, precision and recall. The authors conducted research on news headlines dataset using LSTM and GRU. The LSTM-based model achieves a level of accuracy with 82 percent while GRU has an accuracy of 78 percent. LSTM's AUC score is also higher than that of the GRU model. In addition, the readability score of the LSTM algorithm demonstrates LSTM superior performance in readability.

The authors found that methods in their (Sinha and Yadav (2023)) paper employing in sarcasm detection techniques using a range of deep learning (DL) and machine learning (ML) algorithms, including logistic regression, random forests, decision trees, long short-term memories (LSTM), convolution neural networks (CNN). There were limitations to these methods in accurately detecting sarcasm, especially when dealing with large datasets. The authors aimed to provide a better method for sarcasm identification in news headlines that combines LSTM and CNN algorithms enhancing sarcasm detection's accuracy and here the authors managed to achieve nearly greater than 97 percent of accuracy by leveraging the strengths of both algorithms. Using a dataset of news headlines. The authors also compared their hybrid model to other models to see how effective the LSTM and CNN approach compared.

3.2 Stance

(Hardalov et al. (2021)) presents stance detection to aid in the detection of false postings online, intentional i.e., disinformation / fake, versus misinformation i.e., unintentionally false. The authors examine the relationship between misinformation and disinformation through surveying existing works to identify gaps and future directions. For this paper's research it will incorporate stance detection as a component in the development of code to aid in assisting readers to discern true from fake political twitter postings. This is similar to the (Hardalov et al. (2021)) paper, and to the paper from (Lillie and Middelboe (2019)) in so far as both examine stance detection. It is planned to build on sarcasm detection, stance detection to alert readers to fake news in a more nuanced way i.e., not a binary yes/no, the current paper furthers my research idea that the erroneous news signaling can be higher refined in its flagging by discerning if it is actual disinformation or just misinformation.

(Lillie and Middelboe (2019)) survey paper presents academic work carried out within the field of stance classification and fake news detection. The paper includes a proposal of a system implementation based on the presented survey. (Lillie and Middelboe (2019)) shares the concern about quality of information that (Hardalov et al. (2021)) does for the use of stance detection with misinformation detection. Crowd stance (Dungs et al., 2018) using Hidden Markov Models (HMM) and feature engineering have significant importance in several approaches, which is shown in (Aker et al., 2017). Challenges for analysing data from microblogs, such as Twitter / X Platform, is the model organism problem, as found by (Lillie and Middelboe (2019)) that causes a prevalent issue of representation and visibility when continuously using the same platforms in stance classification. (Lillie and Middelboe (2019)) found a particular approach of the HMM model in analysing rumours in microblog data, achieving very promising results. Decision tree model classification also produced good results though a criticism would be decision tree is too simplistic and not intuitive to all consumers of media.

3.3 Polarization

Turning to Polarization (Iandoli et al. (2021)) and their examination of 121 papers on social media in the context of polarisation. Where polarisation is a dysfunctional group dynamic whereby participants become more extreme in their beliefs and views on topics debate. This is similar to (Vecchi et al. (2021)) in his paper in terms of examining quality of discourse in a digital democracy and its impact on societal conversations.

Preprocessing the corpus, text-document cleaning helps improve the accuracy, latency and robustness of an application. (Kowsari et al. (2019)) find dimensionality reduction methods, principal component analysis (PCA), linear discriminant analysis (LDA), non-negative matrix factorization (NMF), random projection, Autoencoder, and t-distributed Stochastic Neighbour Embedding (t-SNE), to be aids to reduce the time and memory complexity of algorithms in text classification. Some of these stages and (Kowsari et al. (2019)) research findings on feature extraction, preprocessing, dimensionality will guide the development of the pipeline for the app in my own paper's research, particularly as the survey paper by (Kowsari et al. (2019)) is so comprehensive.

(Kowsari et al. (2019)) used existing classification algorithms, Rocchio algorithm, bagging and boosting, logistic regression (LR), Naïve Bayes Classifier (NBC), k-nearest Neighbour (KNN), Support Vector Machine (SVM), decision tree classifier (DTC), random forest, conditional random field (CRF), and deep learning. Again, for this paper's research the above benchmark techniques will be of interest.

For evaluation methods, accuracy, F-Beta, Matthew correlation coefficient (MCC), receiver operating characteristics (ROC), and area under curve (AUC), were used to evaluate the

classification of text highlighting a strength of the (Kowsari et al. (2019)) paper i.e., its evaluation methods. The (Kowsari et al. (2019)) evaluation methods in particular are of interest as the report needs to demonstrate quantified improvement in performance.

Critical limitations of each component of the text classification pipeline (from feature extraction, dimensionality reduction, existing to classification algorithms, and evaluation) are addressed. The (Kowsari et al. (2019)) found limitations can aid dealing and overcoming obstacles met in my own paper's research. Next a comparison is made across all the classification algorithms. Then the use cases for text classification as an application and/or support are reviewed.

In terms of true-false marking a text, (Rohith et al. (2023)) developed a system to simultaneously summarise a text and indicate if it is true or false. The authors reported an improvement of 38 percent in accuracy to 98 percent using a LTSM with a hundred node and forty feature vector architecture. Although in a similar area to my own work as described in the methods and specification section my approach will add more information to guide the user and in a more digestible form.

For quality of narrative and argument in a text, (Vecchi et al. (2021)) addresses argument mining (AM) and its relation to quality of argument as transferred from the social sciences: the contribution itself, its preceding context, and the consequential effect on the development of the upcoming discourse. (Vecchi et al. (2021)) define an application of AM for Social Good: (semi)automatic moderation, a highly integrative application.

Employing a pipeline (Zhan (2021)) presented the pipeline for the NLP task of classification: list preprocessing, feature engineering, dimension decomposition, model selection, and model evaluation. The paper provided an overview of each stage, while surveying and comparing popular classification algorithms. All to understand how different models perform on text datasets of varying size. The algorithms performed quite well, achieving more than 80 percent accuracy. Support vector machine (SVM) was the best-performing classifier using a linear kernel. Methods such as deep neural networks performed comparable to SVM. Though both of these were outperformed by (Rohith et al. (2023)) paper's LTSM model.

Paper	**Description & Drawbacks**	**Accuracy %**
Rahul, Jitendra, Harsh and Kunal (2020)	Sarcasm - Chi-square test to identify features from sentiment, punctuation, then combined with tf-idf, SVM algorithm, voting classifier	SVM 74.59% ; voting classifier 83.53%
Neha and Sukhada (2020)	Sarcasm - pattern approach, four feature	
Spriha and Yadav (2023)	Sarcasm - LSTM and CNN	97%
Chy, Mahin, Hossain and Rasel (2023)	Sarcasm - GRU network, evidential DL based	LSTM and GRU
Damaraji and Rao (2023)	Sarcasm - Naive Bayes, linear classifier, XG Boost, KNN and SGD, sensitive to keywords and phrases	Linear classifier 74.39% ; KNN 75.16%
Bharti, Gupta, Pathik, Mishra (2022)	Sarcasm - Bag of words, term frequency, n-grams frequency, voted classification	Naive Bayes 78.45%, Multinomial Naive bayes 77.63%, Bernoilli Naïve Bayes 78.31%, Logistic Regresion 79.1%, SGD classifier 78%, Linear SVC 77.7%, NuSVC 78.6%, voted classifier 86.4%
Zhou, Elejalde (2024)	Stance - prediction based on collaborative filtering and graph convolution networks	9% outperformance over baseline / further details paywalled by Springer
Alturayeif, Lugman and Ahmed (2024)	Stance — multi-task learning (MTL), joint neural architecture integrating different opinion dimensions, parallel multi-task learning (PMTL), sequential multi-task learning (SMTL), four weighted techniques, best model MTL with hierarchical weighting (SMTL-HW)	F1 78.89%
Ernst (2024)	Stance - large language models to detect stance	
Zhang, Ding, Xu, Guo, Huang, Huang (2023)	Stance - neural production system for stance detection (NPS4SD)	66.2%

Cantarella, Fraccaroli and Volpe (2023)	Truth - text mining, fake news amplifies populism	R-squared 0.906 Rohith, Sooda, Karunakara and Truth - Long Short Term Memory (LSTM) 99% Srinivas, Poria, Hazarika, Majumder and Survey - still open questions in sentiment IMDB 97.4%
Michalcea (2020)	Sentiment analysis (SA) subtasks such as aspect-level SA sarcasm analysis, multimodal SA, sentiment aware dialogue generation	
Zhan (2021)	Survey - of classification steps, preprocessing feature engineering, dimension decomposition, model selection, model evaluation, algorithms NN88% Naive Bayes, SVM, KNN, Random Forest, Neural Networks	NB 86%, SVM 90%, KNN79%, RF 85%,
Onyshchenko and Daniel (2023)	Sarcasm - LSTM in emotion classification using F178 NLP	

Table 1: Table: Summary of Literature Review Results

4 Research Methods & Specifications

4.1 Research Methodology

The paper proposes to use NLP to build an AI copilot-like tool that can operate in near time with the users' reading of media to not only indicate the truth or fake status of a news snippet but add colour to aid both the depth of understanding and speed of understanding to the user i.e., using graphics, charts, colours, shapes, direction, and labels.

The steps to bring the research to completion is to preprocess the two datasets of fake and true news, write an algorithm to cope with human present grammatical anomalies which are confusing to a computer, followed by list preprocessing, feature engineering, dimension decomposition, model selection, and model evaluation.

The output of the research will be quantitative results for probability of truth or fake, sarcasm, stance in the sample data processed, a clear colour coded True / Fake in the adjoining box of the table, then graphics to indicate degree of sarcasm, stance.

The aim is to aid the general reader to the level of manipulation used in information flows that are underhandedly trying to steer the reader to give power, which is what voting for a person as leader is an act of. Once informed through positive AI assistance the reader can then alter and or at least make a more broadly informed decision given the better information she /

he would then hold.

Truth-Fake detection – raw tweet / X platform data related to US presidential elections is used to train the weights in the Logit Regression algorithm. Then the trained model is used to predict Truth-Fake status of new data on the same topic. A score is created for each tweet and then saved to a csv file for passing to the next detector.

Sarcasm detection – reddit data is used to train an LSTM model. Then passing the new twitter / X platform data to the trained LSTM model scores are created for each tweet. The csv file from Truth-Fake detection is then updated to record the Sarcasm detection.

Stance detection – using the academic reference dataset of labelled Stance data an LSTM model is trained. Next the new Twitter / X platform data are passed through the trained LSTM model and the output scores are added as a new column in the csv file that was built up in the Fake-Truth and Sarcasm stages.

Visualisation – with the three columns of scores, Truth-Fake, Sarcasm, Stance, two sets of visualisations are created. Version 1 (Fig 3) is a side by side four column chart of each of the metrics. Version 2 (Fig 4) (Fig 5) is a spatial and colour usage chart, also containing labels at each terminal point.

4.2 Design Specification

The flow of data is shown in figure below (Fig 1). Data moves from raw form through preprocessing and exploration, then splitting into train and test sets, before being passed to the Logit Regression for Truth-Fake, and LSTM model for each of Sarcasm and Stance after which the output from the model is evaluated, compared and visualized, finally a report is written and a presentation made.

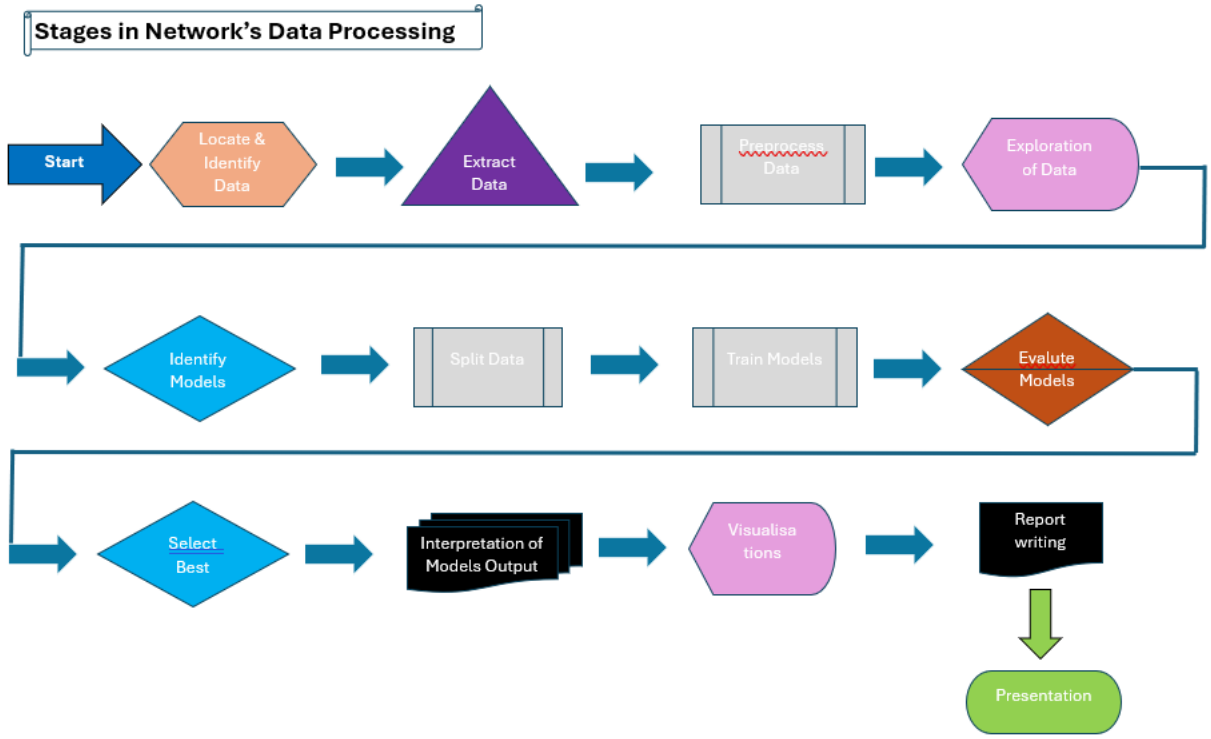

Figure 1: Processing Steps for Paper's Data

4.3 Network Diagram

In the figure (Fig 2) below we see the network diagram. One column per model. First column is the Truth v Fake news. Then the second column is Sarcasm. Finally, the third column is Stance. Truth-Fake value is determined through Logit regression and saved to a csv file. Next Sarcasm value through LSTM is added to the csv file. Lastly Stance value through LSTM are determined and saved to the csv file before the final charting of the three columns metrics in the two versions of the paper's graphic, (Fig 4, Fig 5, Fig 6)

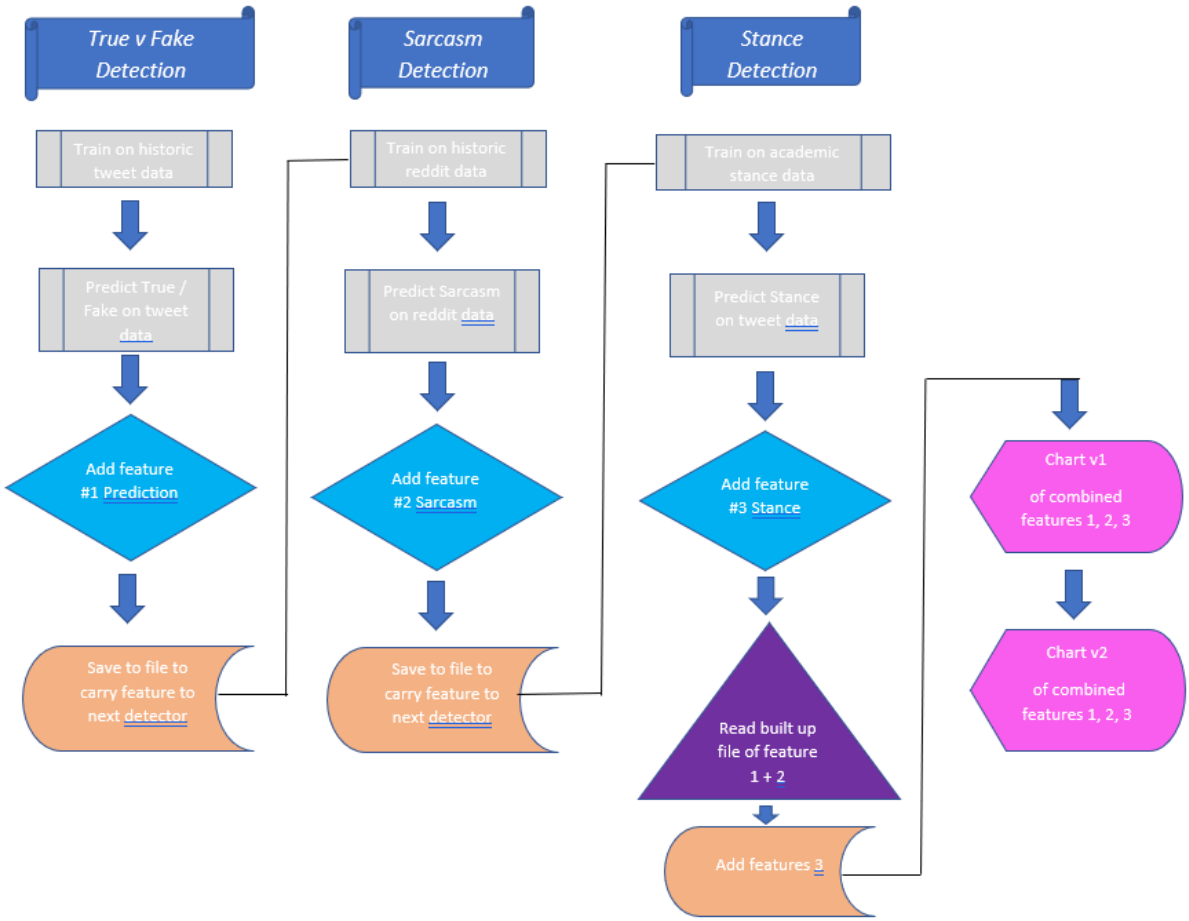

Figure 2: Network Diagram to production of three metrics

4.4 Research Resources

The tools and test data are the following:
Tool: Python 3

4.5 Data Sets

Datasets:

Truth-Fake dataset[1]

The data for Fake and True tweets are in csv format from Kaggle. The tweets relate to the US Presidential election of 2016 for Donald Trump.

Sarcasm dataset[2]

The dataset based on one million reddit comments on a range of topics on discussion boards of reddit.

Stance dataset[3]

The dataset is based on stance position from an academic labelled reference dataset library that was created to address the lack of reliable clean datasets on seven NLP topics, being emoji, emotion, hate, irony, offensive, sentiment, stance.

All the above datasets were available under 2.0 license (Apache License, Version 2.0)

5 Evaluation

The paper evaluates the approach through the use of numeric metrics from the classification report, machine analysed visual media, accuracy score, latency, correlation matrix. Visual output will be the graphs and colours used to better explain the AI numeric values. As seen below in table (Table 2). The paper's models met and exceeded the benchmark accuracy scores. In the follow-on table (Table 3) we see the combined chart (Fig 4, Fig 5) is 0.214545 seconds faster than the benchmark side by chart (Fig 3). Proportionally in percentage terms the combined chart (Fig 4, Fig 5) is 30.13% faster than the benchmark side chart (Fig 3). Although human testing of the output chart i.e., visual media testing (Fig 3, Fig 4, Fig 5) was not conducted the figures were fed into Open AI's Chat GPT4.0-o for evaluation by its Large Language Model (LLM) network. The results returned were that ChatGPT split chart analysis by the chart element, points and labels, lines and colours, legend, then ChatGPT produced an interpretation and context. ChatGPT Interpretation included observations that:

- *"Misinformation" is positioned negatively on both axes, suggesting a negative connotation or impact.*
- *"True News" and "Direct Talk" are positively positioned, indicating positive connotations or impacts.*
- *The red lines might indicate a negative impact or spread of misinformation affecting both "True News" and "Direct Talk."*
- *The green lines connecting "True News" and "Direct Talk" suggest a positive relationship or reinforcement between these two.*

1 - https://www.kaggle.com/code/paramarthasengupta/fake-news-detector-eda-prediction-99
2 - https://www.kaggle.com/datasets/deepnews/fakenews-reddit-comments/data
3 - https://www.kaggle.com/datasets/arashnic/7-nlp-tasks-with-tweets

ChatGPT summarising as

"Overall, the chart illustrates the contrasting relationships between misinformation and credible sources of information, highlighting the negative impact of misinformation and the positive correlation between true news and direct talk."

Accuracy	Truth-Fake News	Sarcasm	Stance
Benchmark	0.99	0.97	0.662
Paper	0.9960	0.9835	0.9978

Table 2: Table of accuracy scores

Chart	Time (seconds)	Proportional % Delta
Side by Side chart	0.712028	
Combined chart	0.497483	
Delta	-0.214545	
Proportional % Delta		30.13%

Table 3: Table of latency scores

6 Discussion

Numerically, examining table (Table 1) on Literature Review findings and table (Table 2) on accuracy scores we can see firstly the matching of the best accuracy score from the Literature Review for the metric Truth-Fake news i.e., Literature Review accuracy of 0.99 matched by my paper's accuracy score of 0.99.
Next for Sarcasm, Literature Review (Table 1) finding was a highest score of 0.97 for Sarcasm score versus a higher accuracy score in my research paper of 0.99. Lastly for Stance, Literature Review (Table 1) found a highest score for accuracy of 0.662 which was bettered by my research to arrive at 0.99.

Turning to visual output, it is trialed in two versions. Version one, truth/fake news, sarcasm, stance, side by side across the three metrics, note the third metric is not binary, absent v present, but has three values, absent, misinformation, disinformation. Consequently, there are four subplots on the one row, not three, the third and fourth subplot being misinformation and disinformation, see figure below (Fig 3).

Developing further, it was thought possible to improve on the version one graphic and combine the four subplots into a single plot to increase speed of understanding. By using spatial and colour heuristics, for example facts such as 85-90 percent of world is right-handed[4] and associates right with positive value[5], versus left with negative value[5].

This subliminal right v left direction value association combined with a similar positive to green colour, for nature and growth, vs red for stop and danger, directed me to develop

4 - https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7058267/
5 – chrome-xtension://efaidnbmnnnibpcajpcglclefindmkajes

the plot as shown in below figures (Fig 4) and (Fig 5). To reduce ambiguity, I label the terminal points and go further to add a colour coordinated to end point and plot legend.

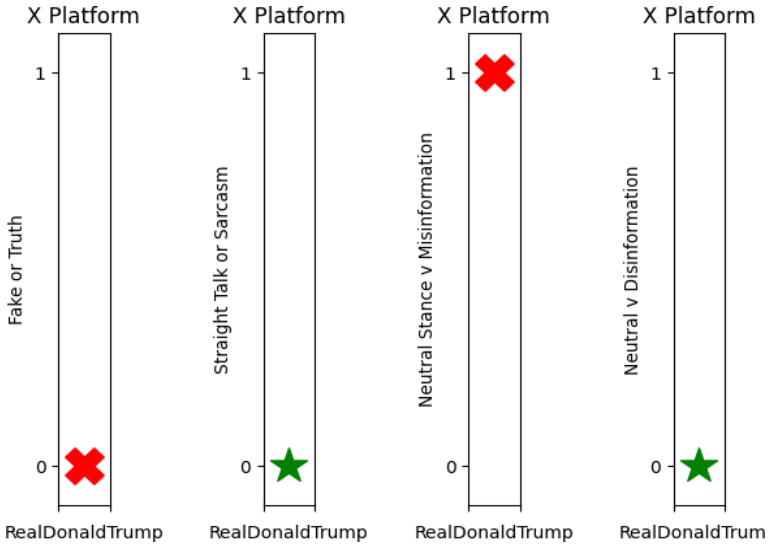

Figure 3: Side by side graphic version1

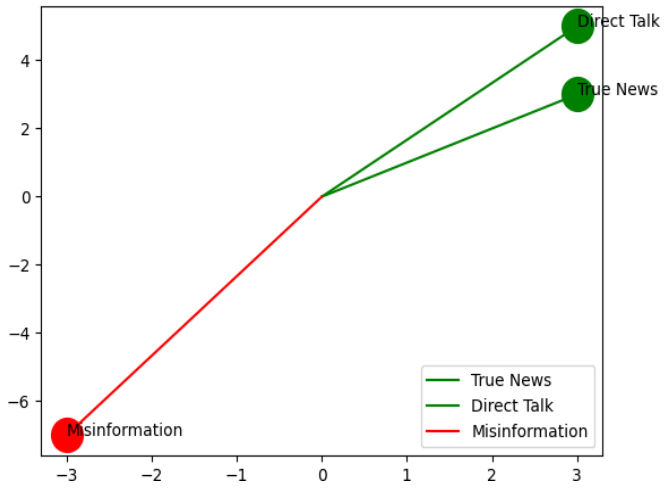

Figure 4: Combined graphic version2a

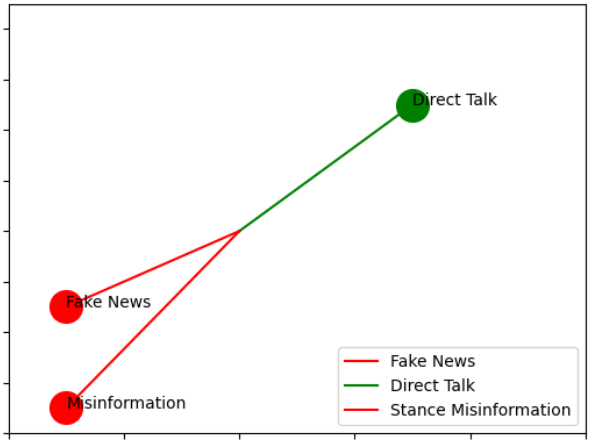

Figure 5: Combined graphic version2b

As detailed in the conclusion the NLP models for truth-fake detection, sarcasm detection, and stance detection compared to the benchmarks favourably. Numerically my truth-fake detection model scored accuracy of 0.99 which compared to (matching) the benchmark of 0.99 ; for sarcasm detection my NLP model scored 0.99 accuracy versus a benchmark of 0.97, so an improvement of 0.02 on the benchmark. Finally for the stance detection my NLP model scored 0.99 versus the benchmark 0.662.

Challenges for the paper were in getting the three data sets to be clean enough to be analysable by the numerical algorithms of NLP, while checking for and ensuring balance in the data, preprocessing the data to be clean enough to reach the benchmark score for truth-fake detection, 0.99, an improvement by 0.02 for sarcasm detection, and largely improve on stance detection to achieve 0.99, a 0.37 accuracy improvement, which is a 49.5% proportional improvement.

Research Generalization

Looking to the future with an aim to preserve performance of the models this generalization can be achieved through the two routes of Data Centric and Model Centric approaches. In Data Centric terms, the techniques of changing to better validation datasets (using K-fold and stratified K-fold cross validation) could be used. Furthermore, data augmentation could be employed to expand the range of data in the test dataset.

Then for Model Centric, there are the options to improve generalization through regularization techniques of L1, L2, and drop out (where nodes in the network are randomly selected for removal). Lastly, as the optimization to an objective function is done through gradient descent, early stopping can be activated, so that by a certain rule, say number of iterations without a set amount of improvement, the processing can be halted early to avoid overfitting and to raise generalisation.

To further aid in avoiding overfitting / and to raise generalisation, inductive bias, whether preference or restricted, can be introduced to the model so that simpler hypothesises are selected versus over fitted ones that fit to include noise.

Lastly, if a generalisation issue remains, domain adaptation (DA), particularly for the stance element of the detection, can be built up with domain adaptations by supervised, semi-supervised, or unsupervised learning using the three techniques divergence-based DA, adversarial-DA, reconstruction-based DA from source to target domain.
With the above steps in generalization, I think the models can generalise to future elections.

Graph Interpretation

I determined the features to use in visualisations through reviewing the outputs of the AI NLP models i.e., use of truth-fake detection accuracy score, similarly for sarcasm detection, and the four classes of stance detection. The combined output in graph form of the record by record / prediction scores for each of truth-fake detection, sarcasm detection, stance detection.

Critically assessing the two versions of the visualization / graphical outputs, the version 1 (side by side graph), although representing the same data, is in fact four charts presented on one line. The consequence is that the data is cluttered, with 24 in number of elements on the side-by-side chart, and the human eye has to decide where to start, middle or left side or right side, additionally what is the meaning of the placement of the side by side charts i.e., does the placement of indicate a prioritisation or strength of performance. On the plus side the vertical

and repetitive structure of the version 1 (side by side graph) produces a uniformity that aids interpretation of the data represented.

In the version 2 (combined graph) presents the same data as the version 1 (side by side graph) but there are only 6 graph elements producing a far less busy / cluttered graph for human reading. Through this reduction in number of elements from 24 to 6, a 75 % reduction, the human reader has lower stress and processing required in understanding the version 2 (combined graph). Additionally, the leverage of directional and colour proclivities present in the public speeds up the time to understand the version 2 (combined graph). Finally, the labelling of the terminal points in the version 2 (combined graph) copper fastens the meaning of the directional and colour messages of the version 2 (combined graph)

User interpretation could be changed with alterations in colour choices, say using temperature related colours (blue for cool and orange for hot), could produce a different effectiveness in interpretation.

Data

Consistency in data means the quality of the data as measured by its level of uniformity, accuracy, coherence across datasets. To achieve this I pre-processed the data in similar manners across the three sources. I ensured consistency through using the same cleaning and preprocessing across the three datasets. Using the libraries gensim I cleaned the data through the following steps. Removing stop words standard to the genism library, extended to include 'from', 'subject', ' re', 'edu', 'use', also words of size less than 2 characters in length. Removed duplicates, corrected spellings, standardising naming conventions transforming names 'politics' to 'PoliticsNews', 'politicsNews' to 'PoliticsNews' to ensure data was consistent

7 Conclusion

The problem examined in the paper was to achieve better informed consumers of US political media transmitted on X Platform, so as to raise understanding, and make readers aware of manipulations through fake news, sarcasm, stances of misinformation and disinformation. Additionally, to reduce polarization in posts.

We can conclude the combined chart, figures (Fig 4) and (Fig 5), is most effective. This is found through a comparison of figures above, the side-by-side graph (Fig 3) versus the later combined style graph figures (Fig 4) and (Fig 5). It is clear the less cluttered figures in the combined style (Fig 4) and (Fig 5) are superior in conveying the data visually. This being achieved through use of direction and colour and augmented with terminal point labelling. Further the same data is visualized in each figure (Fig 3), the side-by-side graph, and figures (Fig 4) and (Fig 5), the combined style graph. So there is no loss in information while the aforementioned gain within figures (Fig 4) and (Fig 5), the combined style grpahs, the information in each set of visualisations is retained. Tweet media consumer understanding is assisted with figures (Fig 4) and (Fig 5), the combined style graphs. Thereby aiding the users' awareness of Fake-News, Sarcasm, Stance within the tweet data.

In terms of time and latency, the combined style graph, as displayed in figures (Fig 4) and (Fig 5) is fastest to produce, has lowest latency, a 30.13% time to generate improvement vs the side-by-side graph which is a further aid to consumers of tweet media given the high velocity of tweet data. Accuracy is held at benchmark for True-Fake-News detection 0.9960 vs benchmark 0.99, while accuracy scores for the underlaying data for each of Sarcasm detection 0.9835 vs 0.97 for the benchmark, and Stance detection 0.9978 vs benchmark 0.67.

On setbacks and challenges not overcome. The graphic output was not tested on human

test subjects. Further there was not time to test for an improvement in polarization.

Future work

It would be in the direction of converting the graphs to code language used within the X Platform itself. Additionally, to expand the graphics, perhaps in to the area of clearly designed emojis along with testing of efficacy of graphics with a human test panel.

It would be interesting to conduct tests of the visualisation with humans. I think splitting the group in to control and test groups would be the first step. Then providing the graphs as listed in the paper (version 1 (side by side), then version 2(combined graph)) so a comparison to the interpretation from the Open AI ChatGPT4.0o can be made. Additionally, a second group set where the text is read and opinions recorded then compared to side-by-side graph for the control group, and combined graph for the test group. A comparison from human text interpretation to each graph can be made and scored, then the control v test group accuracy scores compared.

Furthermore, for research tests with human participants

The testing method would be in-person. The reason for this is cost, and control of the process of the test. Other options of testing for the testing format would be

- **Guerrilla user testing** —informal in nature with available colleagues
- **Remote panel testing** — through apps like *What Users Do*
- **Moderated remote test** — online with participants I would recruit, a software for this option is Lookback.io but Zoom or Microsoft Teams can be used, which would have the benefit of the ability to generate a written transcript of a recorded meeting
- **Ethnography** — in situ where the participants are located, akin to natural environment, idea being to set participants at ease and receive the most accurate answers
- **In-person tests** — in my work location, would be face to face in same physical location but not as comfortable for the participants

Further steps to take are having participants use their own equipment. It is better to explicitly state that I did not design the graphs so as to avoid the risk of the participants wanting to please you with their answers. Frame the test in a believable scenario that is familiar to the participants. Ask a colleague to take the notes – so there are no distractions in testing and no valuable replies missed. Use a consistent script and question list. Group size limited to five participants — using the Nielsen Norman Group size formula which is enough to spot trends while avoids the group becoming unwieldy.

8 Ethical Considerations of the Research

The data are publicly available under Apache 2.0 license. There are no test nor data subjects in the paper's research.

9 Project Plan

Planned research is to follow the Gantt chart shown in Figure (Fig 6) below

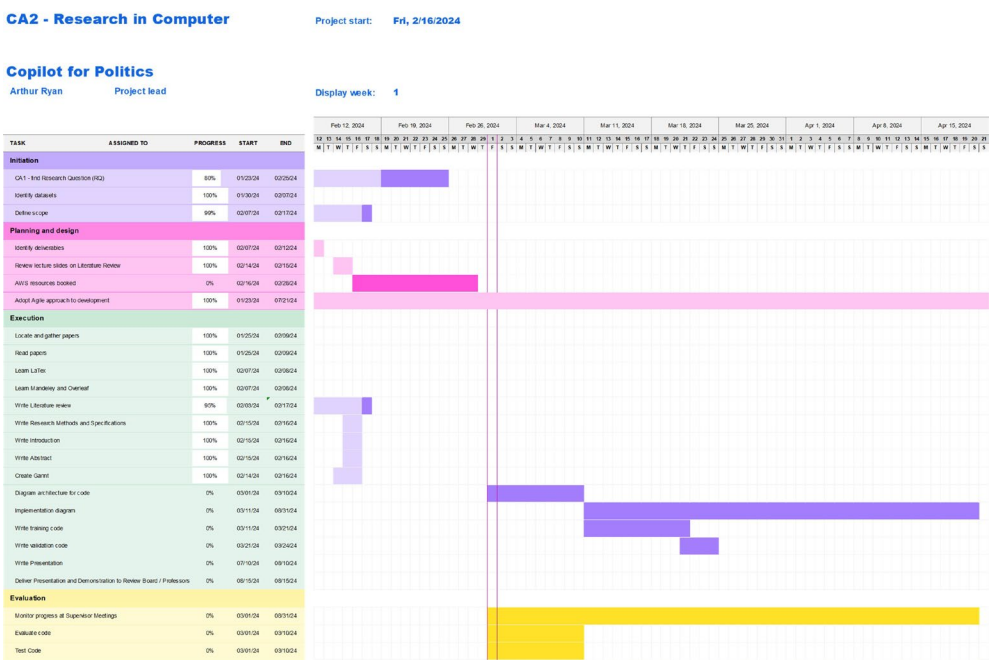

Figure 6: Project Plan for Research Work

References

Anshul Saini (2024). Guide on Support Vector Machine (SVM) Algorithm.

Arifuddin, N., Indrabayu, and Areni, I. (2019). Comparison of feature extraction for sarcasm on twitter in bahasa. cited By 5.

Bharti, S. K., Gupta, R. K., Pathik, N., and Mishra, A. (2022). Sarcasm detection in news headlines using voted classification. page 208 212. Cited by: 2.

BTD (2023). Understanding Neural Network Architecture: Neurons in a General Neural

Network. Cantarella, M., Fraccaroli, N., and Volpe, R. (2023). Does fake news affect voting

behaviour? *Research Policy*, 52(1).

Chebolu, S. U. S., Dernoncourt, F., Lipka, N., and Solorio, T. (2022). Survey of Aspect-based Sentiment Analysis Datasets.

Chy, M. S. R., Chy, M. S. R., Mahin, M. R. H., Rahman, M. M., Hossain, M. S., and Rasel, A. A. (2023). Sarcasm detection in news headlines using evidential deep learning-based lstm and gru. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 14406 LNCS:194 202. Cited by: 0.

Damaraju, A. J. and Rao, D. (2023). Beyond face value: Identifying irony and sarcasm in short texts through multiclass classification techniques. page 74 79. Cited by: 0.

Enes Zvornicanin (2023). Differences Between Bidirectional and Unidirectional LSTM.

Gelles-Watnick Risa (2024). Pew Institiute - Mobile Phone Ownership - PI 2024.01.31 Home-Broadband- Mobile-Use FINAL - 3-2-2024. *Pew Institute*.

Gupta, R., Kumar, J., Agrawal, H., and Kunal (2020). A statistical approach for sarcasm

detection using twitter data. pages 633–638. cited By 29.

Hardalov, M., Arora, A., Nakov, P., and Augenstein, I. (2021). A Survey on Stance Detection for Mis- and Disinformation Identification.

Kowsari, K., Meimandi, K. J., Heidarysafa, M., Mendu, S., Barnes, L., and Brown, D. (2019). Text classification algorithms: A survey.

Lillie, A. E. and Middelboe, E. R. (2019). *Fake News Detection using Stance Classification: A Survey*. PhD thesis.

Manav Mandal (2024). Introduction to Convolutional Neural Networks (CNN).

Niklas Lang (2022). Understanding LSTM: Long Short-Term Memory Networks for Natural Language Processing.

Pawar, N. and Bhingarkar, S. (2020). Machine learning based sarcasm detection on twitter data. pages 957–961. cited By 35.

Rohith, H. P., Sooda, K., Karunakara Rai, B., and Srinivas, D. B. (2023). A Natural Language Processing System for Truth Detection and Text Summarization. In *Proceedings - 7th International Conference on Computing Methodologies and Communication, ICCMC 2023*. Institute of Electrical and Electronics Engineers Inc.

Sharma, S. and Joshi, N. (2024). An optimized approach for sarcasm detection using machine learning classifier. In Nanda, S. J., Yadav, R. P., Gandomi, A. H., and Saraswat, M., editors, *Data Science and Applications*, pages 73–86, Singapore. Springer Nature Singapore.

Sinha, S. and Yadav, V. K. (2023). Sarcasm detection in news headlines using deep learning. Cited by: 0.

Sundararajan, K. (2021). Textual feature ensemble-based sarcasm detection in twitter data. *Intelli- gence in Big Data TechnologiesâBeyond the Hype, Vol. 1167 of Advances in Intelligent Systems and Computing*, page 2021. cited By 1.

Vecchi, E. M., Falk, N., Jundi, I., and Lapesa, G. (2021). Towards Argument Mining for Social Good: A Survey. Technical report.

Zhan, T. (2021). Classification Models of Text: A Comparative Study. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference, CCWC 2021*, pages 1221–1225. Institute of Electrical and Electronics Engineers Inc.