

Interpretable Machine Learning in Natural Language Processing for Misinformation data

by

Yolanda Nkalashe

Submitted in partial fulfillment of the requirements for the degree
MIT Big Data Science
in the
Engineering, Built Environment and Information Technology
University of Pretoria, Pretoria

November 2022

Publication data:

Miss.Yolanda Nkalashe.Interpretable Natural Language Processing (Misinformation),University of Pretoria,(MIT Big data Science), Pretoria, November 2022.

Interpretable Machine Learning in Natural Language Processing

by

Yolanda Nkalashe

E-mail: u13193016@tuks.co.za

Abstract

The interpretability of models has been one of the focal research topics in the machine learning community due to a rise in the use of black box models and complex state-of-the-art models [6]. Most of these models are debugged through trial and error, based on end-to-end learning [7, 48]. This creates some uneasiness and distrust among the end-user consumers of the models, which has resulted in limited use of black box models in disciplines where explainability is required [33]. However, alternative models, "white-box models," come with a trade-off of accuracy and predictive power [7]. This research focuses on interpretability in natural language processing for misinformation data. First, we explore example-based techniques through prototype selection to determine if we can observe any key behavioural insights from a misinformation dataset. We use four prototype selection techniques: Clustering, Set Cover, MMD-critic, and Influential examples. We analyse the quality of each technique's prototype set and use two prototype sets that have the optimal quality to further process for word analysis, linguistic characteristics, and together with the LIME technique for interpretability. Secondly, we compare if there are any critical insights in the South African disinformation context.

Keywords: Disinformation, Interpretability, South African, Prototypes, Example-based.

Supervisors : Dr. Vukosi Marivate

Department : Engineering, Built Environment and Information Technology

Degree : Masters in Technology (Big Data Science)

“Interpret means to explain or to present in understandable terms. In the context of ML systems, we define interpretability as the ability to explain or to present in understandable terms to a human. A formal definition of explanation remains elusive; in the field of psychology, Lombrozo states “explanations are the currency in which we exchanged beliefs” and notes that questions such as what constitutes an explanation, what makes some explanations better than others, how explanations are generated and when explanations are sought are just beginning to be addressed.”

Doshi-Velez, F. and Kim, B. (2017) ‘Towards A Rigorous Science of Interpretable Machine Learning’.

Contents

List of Figures	iv
List of Algorithms	ix
List of Tables	x
1 Introduction	1
1.1 Motivation	2
1.2 Objectives	4
1.3 Contributions	5
1.4 Limitations	5
1.5 Dissertation Outline	5
2 Literature Review	7
2.1 Taxonomy of Interpretability	7
2.1.1 Evaluation of Interpretability	8
2.1.2 Classification of Interpretability	9
2.2 Intrinsically Interpretable models	11
2.3 Post-hoc Techniques	14
2.3.1 Gradient based techniques	14
2.3.2 Surrogate Models	17
2.3.3 Example based techniques	18
2.4 Application of NLP Interpretable models on misinformation models.	21
2.5 Summary	23

3	Data Collection	24
3.1	Datasets	24
3.1.1	ISOT DATA	24
3.1.2	COVID Dataset	25
3.1.3	South African Dataset	26
3.2	Data Pre-processing	27
3.3	Exploratory data analysis	28
3.3.1	Word/length analysis	28
3.3.2	Sentiment analysis	32
3.3.3	Unsupervised Topic Modelling	35
3.4	Summary	35
4	Measuring and Comparing Attribution Type Interpretable Models	36
4.1	Logistic Regression & Decision Trees	36
4.2	LIME (Local interpretable model-agnostic explainer)	40
4.3	SHAP Values	41
4.4	Attribution values: Integrated gradients	42
4.5	Quantitative results of trained models	42
4.6	Summary	43
5	Methodology	44
5.1	Experimental Design	44
5.2	Algorithms	45
5.2.1	K-nearest neighbour Clustering Algorithm	45
5.2.2	Set Cover Algorithm	47
5.2.3	Influential Instances	49
5.2.4	MMD-Critic Algorithm	50
5.3	Prototype Analysis	54
5.4	Summary	54
6	Results	55
6.1	Prototypes Selected	55

6.1.1	Clustering Sets	55
6.1.2	Set Cover Sets	61
6.1.3	Influential function sets	65
6.1.4	MMD-Critic Sets	69
6.2	Quality of prototypes	73
6.3	Further Processing of Selected Sets.	74
6.3.1	Analysis of Set Phrases	74
6.3.2	Sentiment and opinion detection	78
6.3.3	Word analysis	81
6.3.4	Linguistic Properties	85
6.3.5	Prototypical examples used with SP-Lime	87
6.3.6	Critics and harmful instances	88
6.4	Summary	90
7	Conclusions	92
7.1	Summary of Conclusions	92
7.2	Future Work	93
	Bibliography	95
A	Appendix 1	101
A.1	Word Cloud analysis	102
B	Appendix 2	103
B.1	Topic Modeling	104
C	Appendix 3	105
D	Appendix 4	107

List of Figures

2.1	High-level overview of a typical machine learning system. Which takes in an input of X data points, which is fed into a learned machine learning algorithm f with parameters and outputs a vector of decisions \hat{Y}	10
2.2	High-level overview of a typical post-hoc interpretable model. Which uses the output of machine learning and applies to an interpretable technique G to output an explanation.	11
2.3	High-level overview of a salient-based explainable model. Which uses the output of machine learning and calculates the gradient of output with respect to input to determine important features for making the decision.	15
2.4	Overview of a Layer-wise propagation technique as shown in [42]. Shows the decomposition process from model prediction layer to input feature layer.	16
2.5	Overview of counterfactual generator as shown in [38]. Shows how from the positive classified input text of a sentiment classifier "This movie is great," a % of the words are masked; in this case, the word 'great' is masked, and editor searches for appropriate counterfactual for the use case being "This movie is bad." Explaining that if the word "great" was not there, the sentiment decision would not have been positive	19
2.6	Overview of prototype and critics selection process; we start of with the full dataset X and from the data set add, x_i , data instances which minimise the distance between the distribution of prototypes and full dataset	21
3.1	Density plot of word count for ISOT dataset	29
3.2	Density plot of word count for Covid dataset	30

3.3	Density plot of word count for SA dataset	31
3.4	Density plot of Sentiment for ISOT Data	32
3.5	Density plot of Sentiment for Covid Data	33
3.6	Density plot of Sentiment for SA Data	34
4.1	Top positive and negative features for uni-gram logistic regression	37
4.2	Top positive and negative features for tri-gram logistic regression	37
4.3	Local explanation for Tfidf logistic regression model	38
4.4	Local explanation for word2vec logistic regression model	38
4.5	Local explanation for word2vec logistic regression model	39
4.6	Local explanation for word2vec LIME model	40
4.7	SHAP explainer on ISOT data	41
4.8	Attribution map using integrated gradients explanation for class label 0	42
5.1	High-level experimental design steps taken to achieve results	45
5.2	Image showing Set Cover example for a classification toy example	48
6.1	Word Cloud class 1(Fake)for ISOT	58
6.2	Word Cloud class 0(True) for ISOT.	58
6.3	Word Cloud class 1(Fake)for COVID Data	59
6.4	Word Cloud class 0(True) for COVID Data	59
6.5	Word Cloud class 1(Fake)for SA Data	60
6.6	Word Cloud class 0(True) for SA Data	60
6.7	Word Cloud ISOT prototype output for class 1(Fake) using Set Cover	62
6.8	Word Cloud ISOT prototype output for class 0(True) using Set Cover	62
6.9	Word Cloud Covid prototype output for class 1(Fake) using Set Cover	63
6.10	Word Cloud Covid prototype output for class 0(True) using Set Cover	63
6.11	Word Cloud SA prototype output for class 1(Fake) using Set Cover	64
6.12	Word Cloud SA prototype output for class 0(True) using Set Cover	64
6.13	Word Cloud ISOT prototype output for class 1(Fake) using Influential instances	66
6.14	Word Cloud ISOT prototype output for class 0(True) using Influential instances	66

6.15	Word Cloud Covid prototype output for class 1(Fake) using Influential instances	67
6.16	Word Cloud Covid prototype output for class 0(True) using Influential instances	67
6.17	Word Cloud SA prototype output for class 1(Fake) using Influential instances	68
6.18	Word Cloud SA prototype output for class 0(True) using Influential instances	68
6.19	Word Cloud ISOT prototype output for class 1(Fake) using MMD-Critic technique	70
6.20	Word Cloud ISOT prototype output for class 0(True) using MMD-Critic technique	70
6.21	Word Cloud Covid prototype output for class 1(Fake) using MMD-Critic technique	71
6.22	Word Cloud Covid prototype output for class 0(True) using MMD-Critic technique	71
6.23	Word Cloud SA prototype output for class 1(Fake) using MMD-Critic technique	72
6.24	Word Cloud SA prototype output for class 0(True) using MMD-Critic technique	72
6.25	Figure showing top 10 phrases extracted from ISOT MMD prototypical set examples. Green phrases are true news phrases and red fake news phrases	75
6.26	Figure showing top 10 phrases extracted from ISOT Influential prototypical set examples. Green phrases are true news phrases and red fake news phrases	75
6.27	Figure showing top 10 phrases extracted from SA MMD prototypical set examples. Green phrases are true news phrases and red fake news phrases	76
6.28	Figure showing top 10 phrases extracted from SA Influential prototypical set examples. Green phrases are true news phrases and red fake news phrases	76

6.29	Figure showing top 10 phrases extracted from Covid MMD prototypical set examples. Green phrases are true news phrases and red fake news phrases	77
6.30	Figure showing top 10 phrases extracted from Covid Influential prototypical set examples. Green phrases are true news phrases and red fake news	77
6.31	Figure showing opinion mining results based on the objective measure of textblob from all three datasets. It shows the percentage count of the text values that were subjective compared to objective	78
6.32	Figure showing opinion mining results based on subjectivity measure of textblob from all three datasets. It shows the percentage count of the text values that were subjective compared to objective	79
6.33	Figure showing sentiment results based on polarity measure of textblob from all three datasets. It shows the percentage count of the text values that were negative	79
6.34	Figure showing sentiment results based on polarity measure of textblob from all three datasets. It shows the percentage count of the text values that were neutral	80
6.35	Figure showing sentiment results based on polarity measure of textblob from all three datasets. It shows the percentage count of the text values that were positive	80
6.36	Figure showing word differences and similarities between the two class groups in SA prototypical examples	82
6.37	Figure showing word differences and similarities between the two class groups in ISOT prototypical examples	83
6.38	Figure showing word differences and similarities between the two class groups in COVID prototypical examples	84
6.39	Figure showing part of speech tag for Covid prototypes	85
6.40	Figure showing part of speech tag for ISOT prototypes	86
6.41	Figure showing part of speech tag for SA prototypes	86
6.42	Output of using SP-LIME with Prototypical examples to explain and indicate prevalent words for true news class	87

6.43	Output of using SP-LIME with Prototypical examples to explain and indicate prevalent words for Fake news class	88
6.44	Output showing comparison of phrases selected for harmful instance of fake news data to that of a true news instance	89
6.45	Output of using attribution map to determine key features for harmful instance indicated by influential and MMD-critic	89
A.1	Word Cloud class 1(Fake) word clod image.	101
A.2	Word Cloud class 0(True) word clod image.	101
A.3	Word Cloud class 1(Fake) word clod image.	102
A.4	Word Cloud class 0(True) word clod image.	102
A.5	Word Cloud class 1(Fake) word clod image.	102
A.6	Word Cloud class 0(True) word clod image.	102
B.1	Bi-gram topic analysis for ISOT Data	103
B.2	Bi-gram topic analysis for Covid Data	104
B.3	Bi-gram topic analysis for SA Data	104
C.1	SP-LIME results from a representative sample of fake news dataset	105
C.2	SP-LIME results from a representative sample of True news dataset	106
D.1	Extracted fake-Covid class prototypical examples	107
D.2	Extracted True-Covid class prototypical examples	108
D.3	Extracted fake-ISOT class prototypical examples	109
D.4	Extracted True-ISOT class prototypical examples	110
D.5	Extracted Fake-SA class prototypical examples	111
D.6	Extracted True-SA class prototypical examples	112

List of Algorithms

5.1	Clustering Prototype selection method	46
5.2	MMD-Critic Methodology: Prototype selection	53
5.3	MMD-Critic Methodology: Critic selection	53

List of Tables

2.1	Table of CoCoGen feature groups and descriptions adapted from the literature [36]	13
3.1	Table showing attributes ISOT dataset	25
3.2	Table showing number of news articles in each class	25
3.3	Table showing attributes COVID dataset	26
3.4	Table showing attributes of South African dataset	26
3.5	Table showing prominent words in each datasets as observed in the word cloud in Appendix1	28
4.1	A short table caption	42
6.1	Table showing the size of each prototype set using clustering technique for our dataset	56
6.2	Table showing the match rate of the prototype sets for ISOT data	56
6.3	Table showing the match rate of the prototype sets for Covid data	57
6.4	Table showing the match rate of the prototype sets for SA data	57
6.5	Table showing the size of each prototype set using Set Cover technique for our datasets	61
6.6	Table showing the size of each prototype set using Influential technique for our dataset	65
6.7	Table showing the size of each prototype set using MMD-Critic technique for our dataset	69
6.8	Table showing the quality of each prototype set by observing prediction quality	73

6.9 Average Words 81

Chapter 1

Introduction

Machine learning models have become less explainable in the last couple of decades, such that they can solve complex problems while not compromising on predictive accuracy [3, 25, 16]. However, the more complex the machine learning model is, the more difficult it can be to explain how it made its decisions. Therefore, explaining or interpreting how a model makes its decisions has become paramount in recent years. Resulting in the model's learning biases, models not performing as expected in "the real world," end-users wanting to understand why a model selected or did not select them for a particular class/category, etc. Interpretability has thus become a crucial research field, especially in high-risk areas like self-driving cars, medical diagnosis, and criminal detection systems. Interpretable models build trust and transparency between the model and end-users. Various explainable models have been developed; however, some of the explainable models do not produce humanly interpretable outputs. Human interpretable models are essential as this ensures lay users can understand how a model makes its decision without having to be the model/system developer.

Natural language processing models interact with society frequently, like translation systems, next-word prediction, sentiment, and fake news detection. Therefore explainability needs to be humanly understandable in this field to ensure safety and trust.

This research aims to explore a humanly explainable model in natural language focusing on misinformation. We leverage known humanly explainable machine learning techniques on misinformation data to determine if there are no key characteristics of misinformation data that can aid in safety online when it comes to fake news.

This chapter provides an introduction to the study; we first discuss in detail the motivation and background, followed by research objectives & contributions, and finally, discuss the outline of this paper.

1.1 Motivation

Explainable AI refers to techniques designed to assist in interpreting how models make their decisions. The need to be able to interpret how a model makes its decision has risen in the past decade; due to regulation and ethical concerns for end-users to understand how a model came to its decisions [3]. For example, a recent law that has been put in place is the General Data Protection Regulation (GDPR) which states the "right to explanation," meaning that consumers of a model have a right to be provided with an understandable reason on how the model reached its decision [16, 27].

Interpretability is not required by all machine learning tasks or use-cases, it is required where an incorrect outcome has significant consequences or the problem has not been studied sufficiently leading to untrust in systems output [16, 3].

Explainable machine learning in natural language processing (NLP) tasks has recently become a focal topic. Historically NLP systems were based on hand-written rules and decision trees based on hard if-then rules. Thus, in earlier years, NLP tasks were explainable to a certain extent [13, 42]. However, with the introduction and rise in deep learning models and modern NLP tasks, the systems and models used for NLP tasks are more complex with deep learning networks or state-of-the-art transformer models such as BERT (Bidirectional Encoder Representations from Transformers) being used, making them less transparent and humanly explainable [3, 16].

The need for interpretability in NLP tasks is because most of the tasks in NLP interact with the society at large and tend to learn human biases [42]; For example, a sentiment classifier that Luminoso trained showed that the trained classifier was representing racist behaviour; as a neutral sentence "Let's go get Italian food" was classified more positive than a functionally similar statement "Let's go get Mexican food." Another well-known example is Taytweets, the Microsoft tweet chatbot, which was supposed to resemble a 19-year-old girl but instead learned and tweeted racist and offensive tweets. This shows that interpretability in NLP is crucial to ensure the models can be trusted as some of the models have the potential to create instability in society. [13, 11].

In this research, we focus on the interpretability of NLP misinformation classification tasks. We have seen a rise in misinformation being distributed on various social media platforms. An example is the current COVID-19 pandemic; a lot of misinformation has been filtering through the media, causing significant panic in society. Governments took this spread of misinformation seriously and even tasked working groups to deal with the issue, as it may impact how effective a government can be in dealing with the pandemic. Thus, if we can explain misinformation and identify humanly interpretable characteristics, there would be more safety online.

Three main techniques can be used to explain models: model-agnostic, model-specific, and example-based techniques [17, 3, 16, 33]. Model agnostic are techniques that can be applied to any model, whilst model-specific are methods that can only be used for a specific family of models and lastly, example-based techniques focus on extracting examples from the data which is then used as an explanation for the models or summarisation of the data [17, 3, 16, 33].

In NLP tasks, extensive research has been done on model-agnostic and model-specific techniques, where the main aim is to determine which variables are the most influential to the decision made by the model. However, there is limited application of the use of example-based techniques where the objective is to explore a sparse sample of the dataset to achieve explainability. This paper explores using example-based techniques to achieve explainability in an NLP misinformation task.

Explainable machine learning literature has limited consensus on the difference between the term's interpretability and explainability. Some literature uses the terms interchangeably, while others have clear distinct differences between the two. In this paper, we will use the terms equally as done so in [29, 33] and define interpretability and explainability as "the degree to which a human can understand the cause of a decision" [16].

1.2 Objectives

The primary goal of the thesis is to use a human interpretable model for misinformation classification to identify and characterise misinformation effectively. We leverage example-based techniques, as psychological studies state humans understand or create reasoning through creating a mental model by using examples [30].

More specifically, our goal seeks to answer the following research questions:

1. What behavioural factors can distinguish a misinformation class?
 - a. Can we use prototypes with a combination of another model and feature engineering technique to extract such features?
 - b. Is there a contrast in these behavioural factors on different datasets from different countries or topics; what underlying commonalities exist?
2. How does prototype/example-based methodology compare to the most commonly used algorithms?
 - a. Is an example-based approach significant to an approach that outputs an attribution map?

The secondary goal of this research is to determine the quality of prototypes selected by training a model on the sparse dataset and analysing how well the model performs. Based on the results, we can evaluate the quality of the selected prototypes (assist in use cases where there is insufficient data).

1.3 Contributions

The main contribution that this research will bring to the explainable research community is the use of example-based techniques on natural language misinformation tasks compared to the extensively researched model-agnostic methods with a focus on extracting key features and characteristics. Instead of focusing on the feature space, we focus on representative data samples and extract key behavioural insights to achieve interpretability in misinformation data. We further compare whether there is a contrast between South African fake news to Global news data characteristics.

1.4 Limitations

The main limitation we experience is the limited data for using South African fake news data. The dataset is comparable small to the rest of the data with only a sample size of 763 Fake news data and more than 40000 real news.

1.5 Dissertation Outline

The rest of this thesis will be organised as follows.:

- In **Chapter 2** we look at and review the literature that has been researched on explainable machine learning.
- In **Chapter 3** we discuss the data that was used in this research and how we performed exploratory data analysis[EDA] on our datasets.
- In **Chapter 4** we provide a comparative application on the well known explainable techniques in natural language processing.
- In **Chapter 5** we discuss the methodology that was applied in this research to determine interpretability in misinformation.

⁰Work presented on the 28th October 2021 at the Ethics and Explainability for Responsible Data Science (EE-RDS) and the paper is under Review

- In **Chapter 6** we provide the results of the experiments we performed.
- In **Chapter 7** provided the concluding remarks and discussion on future work.

Chapter 2

Literature Review

The research in this thesis aims to understand interpretability for specifically natural language processing tasks with a focus on misinformation tasks. We aim to identify characteristics of misinformation with the help of using interpretability techniques. Such that we can produce humanly explainable examples for misinformation tasks. This chapter starts by summarising the taxonomy of interpretability in section 2.1. We then discuss and review the literature on intrinsically interpretable models in section 2.3, followed by section 2.3, which focuses on post-hoc model interpretability, and finally, section 2.5 summarizes key takeaways from the literature reviewed.

2.1 Taxonomy of Interpretability

Interpretability is defined as the degree to which a human can understand an explanation and is required because of the incompleteness of the problem being solved. However, further to the issue of incompleteness various pieces of literature like [33] and [16] state that the importance and the why for having an explainable model is for the following reasons:

- Trust & Verification:

The models must be trusted by the end-users so that when they interact with the model, they can fully trust the decisions made by it.

- Safety:

Refers to understanding how a model works, which is valuable in high stake decisions. This allows for constant improvement of the system such that it has a near-to-zero error rate as we constantly learn from it.

- Debug & audit:

Explainability allows users and developers to debug models when the model fails.

- Detect bias:

Assists in picking and detecting if a model learns any biases from the data.

The survey by [14]: Opportunities and Challenges in Explainable Artificial Intelligence (XAI) summarises the above-stated need for interpretability into three groups being:

1. Accountability; to create a sense of accountability when a model fails.
2. Fairness; to ensure that models are fair and safe from bias and lastly
3. Transparency; to ensure trust and verification.

It is argued that accountability is the most important reason that we should aim to achieve with interpretability.

2.1.1 Evaluation of Interpretability

There has yet to be a consensus on how to validate interpretable models. However, there are three main suggested evaluation techniques that were first presented in the paper "Towards A Rigorous Science of Interpretable Machine Learning" by [16]:

1. Application grounded method: The explainable model to be evaluated by the end-user in the environment in which it will be deployed.
2. Human grounded: How simply can a human evaluate the explanation.
3. Functionally grounded: How well does the explanation faithfully represent the model.

Further to the proposed evaluation methods [33] and [13] state that an explainable technique should have one or more of the following properties:

- **Translucent:** Relates to how reliant the model is to model parameters. The more translucent an explainable model is the better.
- **Portability:** Model can be used for different types of models.
- **Accuracy:** How well the explainable model predicts unseen data.
- **Fidelity:** How well the model mimics the model's prediction. The lower the fidelity, the more unreliable the explanations are.
- **Consistency:** Explanation must be consistent on different models for the same data.
- **Stability:** high stability is desired for explainable models; similar to consistency an explainable model is stable if there are no significant changes in data instances similar to the data point being explained.
- **Comprehensibility:** Refers to how well humans can understand the explanation.

2.1.2 Classification of Interpretability

Traditionally interpretable models are classified as either intrinsic or post-hoc which is determined based on when interpretability is achieved [11, 42]. For example, suppose interpretability is achieved with the model's output or during the prediction phase. In that case, the model is classified as intrinsic, using figure 2.1 as a visual explanation, the model would end at the output phase and no additional processing would be required. Conversely, post-hoc interpretability is achieved through further processing after the model has predicted an outcome; this is seen in the adapted figure 2.2.

An explainable model $G(\cdot)$ can either be 1. model-specific indicating that the technique can only be used for a specific class of machine learning models, or 2. Model-agnostic indicates that the technique can be used for all model types [16, 3, 27]. The output, g , of an explanation can be a:

- Feature summary: This indicates features that are the most relevant in making a decision for a particular class.
- Feature visualization: Output of an image highlighting the important features or pixels.
- Model internals: Internal weights or structure of models; usually the output of inherently interpretable models like decision trees, linear models, etc. and
- Data points: Returns sparse data points that best represent the full dataset X .

The scope of an explanation output, g , can either be local, explaining a single data point, or global, explaining the high-level overview of the model of class [14, 29, 17].

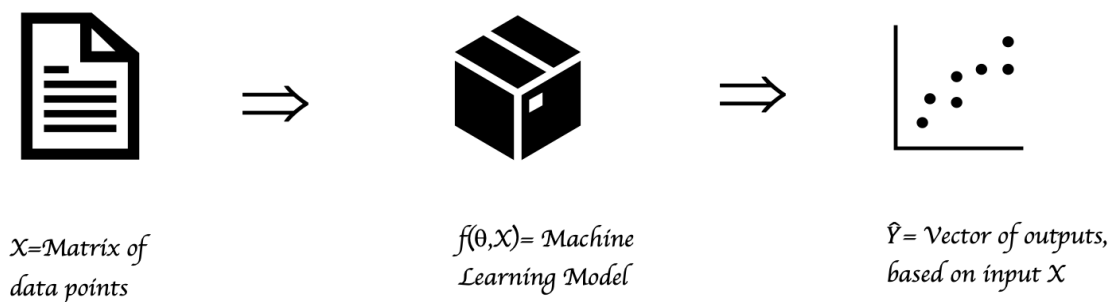


Figure 2.1: High-level overview of a typical machine learning system. Which takes in an input of X data points, which is fed into a learned machine learning algorithm f with parameters and outputs a vector of decisions \hat{Y} .

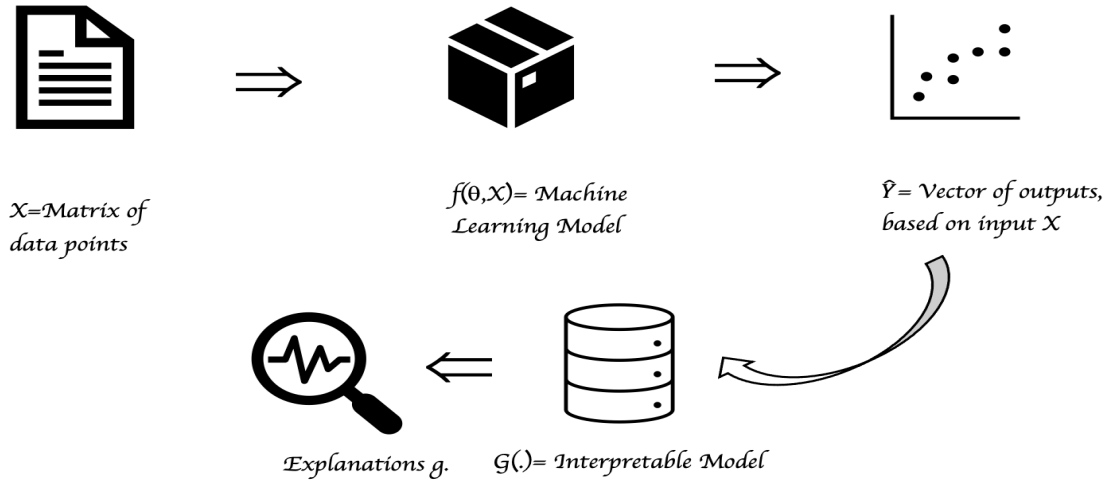


Figure 2.2: High-level overview of a typical post-hoc interpretable model. Which uses the output of machine learning and applies to an interpretable technique G to output an explanation.

2.2 Intrinsically Interpretable models

Most of the traditional intrinsic methods, like decision trees or logistic regression, do not offer the complex learning that comes with dealing with language data resulting in a significant trade-off of in accuracy. This motivated researchers to explore the ability to achieve intrinsic explainability in complex deep learning models by leveraging the internal structure and knowledge learned by the model. The prominently used architecture to achieve intrinsic interpretability using deep learning models is the attention mechanism that can be added to the architecture of a deep learning model [24].

The attention mechanism was proposed to improve the performance of language models. Attention uses a contextual vector, which is the output of the encoder layer; attention allows the input to focus on the previous output or, similarly, for the input features to interact with each other, known as self-attention [24, 44]. Because the attention mechanism produces weights, the weights can be used to output an attention map that highlights the most prevalent features that were used in each layer [12, 6]. Although attention may seem to provide a sense of transparency for deep learning models; [37] and

[6] states that the attention mechanism should be used with caution as it is a mixture of various representations and can be misleading.

[44] proposes to iterate through the attention query using an iterative, recursive attention model which attempts to trace back the contextualized word representation from the encoder with the aim of learning better representations and attention weights that can be used as an explanation of the model rather than normal attention.

Alternative to attention as an intrinsic interpretable model [10, 36, 19, 1] leverage off the convolution neural network (CNN) structure with adaptations in the last layers of the network to achieve explainability for the model's decisions. [49]'s approach uses the CNN with a ReLu(rectified linear unit) in the last two hidden layers, such that they can leverage the local linear models (LLM) produced by ReLu. The explanations are obtained by determining the importance of the text inputs by using the weights of LLM. Similarly, [19] uses a TC-CNN which, instead of ReLu, in the last hidden layer global average pool is applied to the spatial feature vectors of the CNN to obtain a single feature vector F . The obtained feature vector is used on class activation map extraction to determine the contribution of each feature.

Uniquely [36] uses latent features which are explainable to train a deep learning model. The proposed latent variables fall into six categories of the complexity contour generator (CoCoGen), which are part of the Stanford CoreNLP toolkit. The latent variable list and description can be seen below in table 2.1. The approach first analyses text using CoCoGen to obtain an output sequence of 154 feature vectors that relate to the features described in table 2.1. The feature vector obtained can then be fed into a network to determine output and explanation based on the relative importance of the language feature groups. The relative importance of a feature group is determined through delta scores which are standard scores obtained by performing normal standardization on all indicators (feature groups); with the scores, one can distinguish interesting patterns between classes.

Feature group	Size of Sub-types	Example/Description
Syntactic complexity	18	mean-length, clauses per sentences, Coordinate phrases per clause and Particular structures
Lexical Richness	12	contents words, Lexical diversity, type token ratio, Lexical sophistication
Register-based n-gram frequency	25	measures of frequencies, n-gram frequency of order 1-5 from five language registers
LIWC-style	60	function, grammar perceptual, cognitive and biological processes, personal concerns, affect, social, basic drives.
Word-Prevalence	36	information on word frequency, contextual diversity and semantic distinctiveness.

Table 2.1: Table of CoCoGen feature groups and descriptions adapted from the literature [36]

2.3 Post-hoc Techniques

The most common techniques for explainability for deep learning models is post-hoc methods. The techniques aim to achieve interpretability after the model has been trained, by using the information and output of the trained model to explain why a model made its decision as shown visually in figure 2.2. The techniques can be split into gradient/saliency based, surrogate based, probing and example based techniques.

2.3.1 Gradient based techniques

Gradient-based interpretable methods use gradient signals to assign relevancy of feature inputs with the goal of creating an attribution map[35, 43]. The attribution map captures the importance of each feature. Attribution maps have to be sensitive to changes in the input space to ensure they do not focus on unimportant features and are invariant such that they are consistent in the output result when applied to different models [43, 29]. Further, gradient methods commonly achieve local interpretability for a single instance x_i looking back at figure 2.2; figure 2.3 shows an adaptation of the explanation function G and output to show what the methodology outputs and the baseline explanation function.

One of the well-known attribution techniques is saliency, termed the vanilla gradient in some literature. Explainability is achieved by taking the gradient of the output with respect to the input. However, saliency can be very noisy. To improve the noise [29, 2] proposed grad * input to smoothen the attribution maps by multiplying the gradient with the instance value, see equation 2.1 and 2.2 respectively.

$$AttributionMap = A^C = \left| \frac{\partial \hat{Y}}{\partial x_i} \right| \quad (2.1)$$

$$A^C = \left| \frac{\partial \hat{Y}}{\partial x_i} \right| \circ x_i \quad (2.2)$$

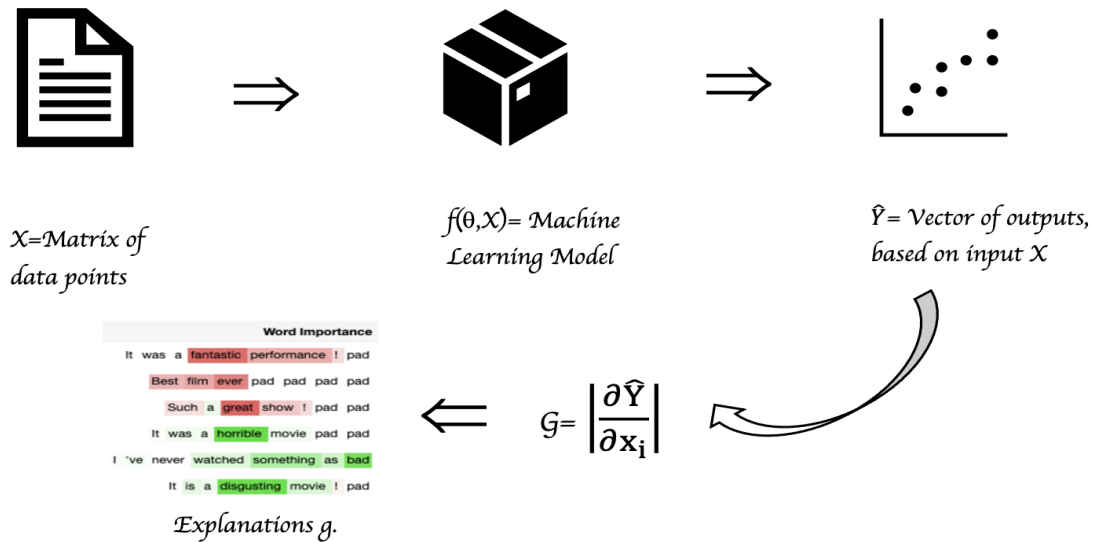


Figure 2.3: High-level overview of a salient-based explainable model. Which uses the output of machine learning and calculates the gradient of output with respect to input to determine important features for making the decision.

Vanilla gradient explanations are usually best suited in models where a linear relationship exists between output and input; given that deep learning models are complex and linearity does not exist, the technique may not be as faithful to deep learning models [34, 35]. [43] proposed integrated gradients, a modification of the gradient approach. Integrated gradients averages all gradients along a straight line between a baseline value \bar{x} and the input of interest x_i . The baseline is usually chosen to be an empty sequence.

Attribution methods only focus on the change in the output layer with respect to the input layer. They do not consider the intermediate layers and the changes of the feature gradient [43, 35]. Attribution propagation is proposed, which looks at each layer from the output to the input. Propagation methods decompose the network work layer by layer assigning relevancy scores on each neuron [35, 34]. One of the most popular attribution propagation methods applied to NLP downstream tasks is layer-wise relevancy propagation (LRP). LRP starts from the outer layer \hat{Y} and works backward, assigning relevancy score, R_k , to each neuron based on its sensitivity it is to its successor layer [43]. The score is calculated by decomposing the activation function in each neuron rather than

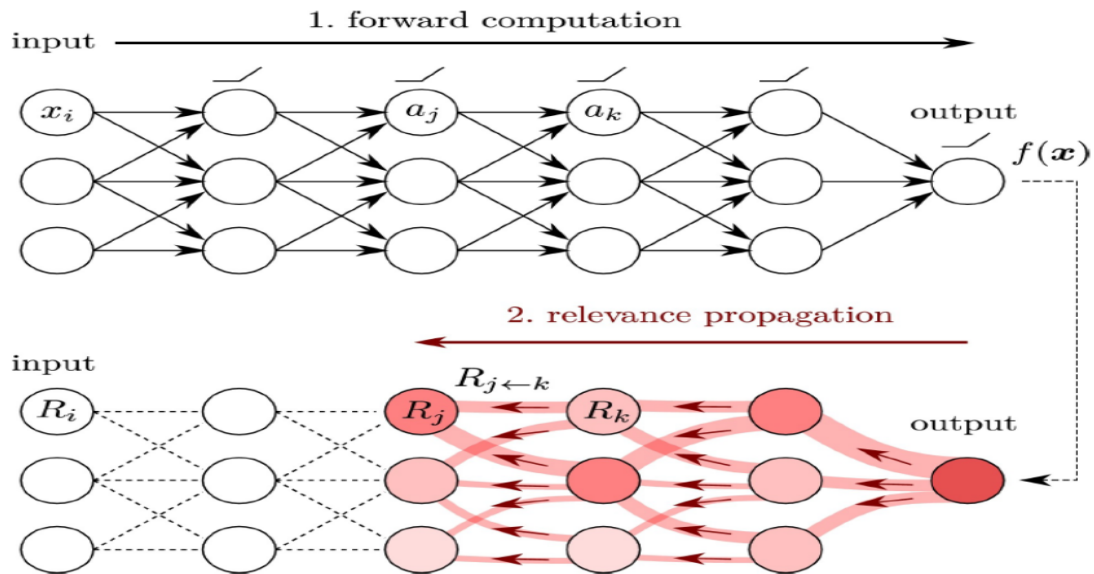


Figure 2.4: Overview of a Layer-wise propagation technique as shown in [42]. Shows the decomposition process from model prediction layer to input feature layer.

calculating gradients [42]. Figure 2.4 shows a visual example of the decomposition.

Alternative to LRP, [39] proposes a more improved propagation technique, DeepLift. Instead of directly redistribution the score from the upper layer to the lower layer, DeepLift uses a baseline to calculate the relevance score by looking at the effect of the activation neuron from the original data, x_i , compared to the baseline data \hat{x}_i . Contrary to attribution-based methods, [27, 3] Proposes the use of surrogate models over attribution methods as attribution methods can be fragile and highly unreliable for an explanation of complex-based models.

2.3.2 Surrogate Models

Surrogate models attempts to mimic the model and how it achieved its output by using the information from the model to train a more intrinsic model with the information captured [37, 23]. Surrogate models were first proposed by [46] to model latent variables with the aim of making the process more explainable. The benefit of using surrogate models for explainability is that they are portable and can be applied to any model for any dataset [33, 16]. The commonly used surrogate model in machine learning is LIME (local interpretable model-agnostic explainer). LIME was first proposed by [37], in the paper "why should I trust you? Explaining the predictions of any classifier". The technique selects an instance x_i for interpretation that has been predicted using a model F ; LIME selects perturbed points close to the instance to create a new data set. The new data is trained using the model F and an intrinsic model like logistic regression to gain insight and explanation for the model's decision. LIME outputs the most significant variables for the model when making a classification decision.

LIME only produces local explanations; as such, there have been various adaptations of LIME to achieve global interpretability. [23] proposes K-LIME, which uses partitioning to partition the space of the predictors and fits an intrinsic model in each partition. Similarly, [37] proposed SP-LIME, which rather than partitioning, picks a set of instances that best represent the data that are used to fit the intrinsic model to gain a global understanding of the model.

A major limitation of LIME is that the weights produced do not necessarily indicate the importance of a feature and can create a false sense of relevance in the features. [49] and [33] propose to compute Shapley values for explanation compared to LIME. Shapley values use the ideology of game theory to create a linear combination or model for every permutation of features. The shapely value shows the average contribution of a feature across all possible permutations. For example given data which has two features $[x_1, x_2]$ shapely will average across the weights obtained from the permutation $[x_1], [x_2], [x_1, x_2]$. The computational time of obtaining Shapley values can exponentially increase with the more features one has.

Contrary to determining which features are relevant as explanation [22], proposes to use feature importance techniques with probing techniques that can inform us what

linguistic properties are captured in the model. There are various techniques which probe for various properties such as syntactical structure, context, semantics, sentence length, word order, verb tense. Probes are shallow networks that are placed on the intermediate layers of a network and can investigate in each layer what is captured [22, 48, 18]. If the task being probed for performs poorly, in accuracy terms, then the information is not encoded or is not needed to make a decision. The main issue with probes is they can be complex and maybe be unfaithful; instead of learning the representation being probed it learns the task [22]. Given that linguistics are essential for NLP, probes are important in giving a holistic explanation and should instead be used as a combination with an alternate explanation technique.

2.3.3 Example based techniques

[45] argues that using importance value for interpretation may be abstract for a lay-user to gain a full understating for NLP task as they may not grasp the significance of the importance of the word “help” for instance. They thus propose using example-based techniques, given that humans work with mental models and contrasts for explanations; therefore, example-based techniques will help humans build mental models of the machine learning model [30]. The three most prominent interpretation example-based methods are:

Counterfactual examples

Counterfactual examples aim to achieve interpretability by producing a contradictory reality; It aims to describe a causal situation by stating that if X had not occurred, Y could not have occurred. In figure 2.5 from [38], we see an example of a counterfactual example generator using text-to-text transfer learning; for an instance where the explanation is a contrasting scenario selected from a sample of candidates. Counterfactual is interested in what changes in the input of an instance will cause a prediction to flip [33, 29]. In earlier years, to obtain counterfactuals, it was either through trial and error, where features were flipped until the desired outcome was obtained, or manual annotations were required by experts of the data [38]. Recently, there have been tool-kits created to obtain counterfactual explanations from a model by masking a percentage

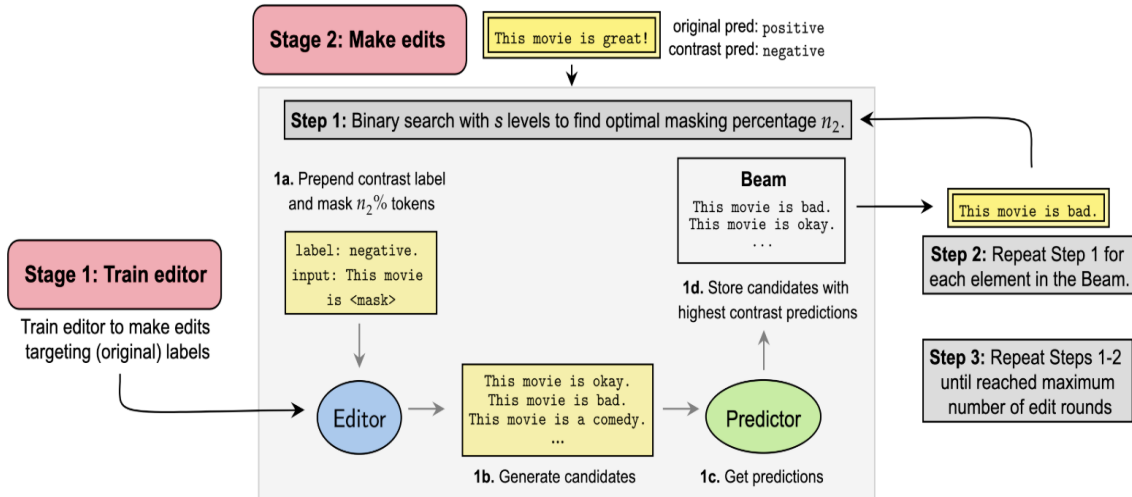


Figure 2.5: Overview of counterfactual generator as shown in [38]. Shows how from the positive classified input text of a sentiment classifier "This movie is great," a % of the words are masked; in this case, the word 'great' is masked, and editor searches for appropriate counterfactual for the use case being "This movie is bad." Explaining that if the word "great" was not there, the sentiment decision would not have been positive

of data through text-to-text transfer learning and conducting a binary search to generate edits, and once the desired outcome is obtained, the process stops. Similarly, [45] proposes Polyjuice, which uses a library of annotated data to generate real-world counterfactuals. Counterfactuals may run into the issue of having too many explanations that can contradict each other [3]. Moreover, they need to be minimal in the sense that a minimal change to the input is inserted into the original data point, further, they must be grammatically, and semantically meaningful [45, 37, 16].

Adversarial Examples

Adversarial examples are very similar to counterfactual examples, but more with the aim of identifying the model's vulnerable points [33, 29]. Adversarial examples are looking for input points that would make the model make an incorrect classification; for instance, in an NLP misclassification case, the input example would be text that would classify the text as genuine when in fact, it is fake. Adversarial examples are commonly used for debugging a model more than interpretability [29].

Prototype Selection

The prototype algorithm goes back to 1968 when it was termed instance-based learning. The original aim of selecting prototypes was for data editing purposes with the aim of improving accuracy and efficiency in their data, rather than wanting to understand the data set [8]. Historically the techniques to select prototypes were using clustering techniques (K-medoids or K-means) and determining how far instances are from one another [8, 33, 25]. A recent approach, with the aim of explainability, is the set cover theorem proposed by [8]. This approach aims to find the smallest subset of data so that every point around the data is in close proximity.

All the techniques above are aimed at just identifying the dataset which best represents the full data; however, what about the outliers of the data? [7] states that if we just rely on prototypes, we may miss important information about our data and consequently over-generalize. They propose an approach that leverages a bayesian framework and aims to extract prototypes as well as critics which are data points that do not best represent the data to gain holistic explainability and understanding of complex data [7, 33]. The algorithm is commonly known as MMD-critic. It has been explored extensively in image classification tasks where the prototypes are visual examples of images that best represent the data. From the small set of prototypes and critics, one can understand the data and different attributes and features; a visual example of the method can be seen in figure 2.6. The outputs of the prototypes can be used to explain a particular class in a classification task and why the model made its decision. Further, prototypes can be used to train the nearest neighbour classifier for intrinsic explainability [3].

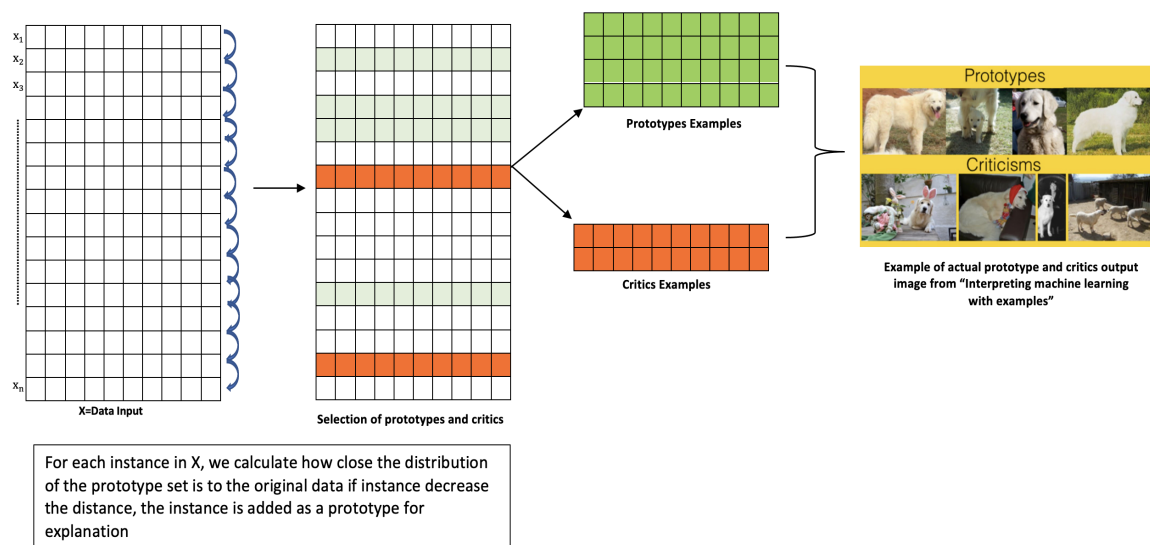


Figure 2.6: Overview of prototype and critics selection process; we start of with the full dataset X and from the data set add, x_i , data instances which minimise the distance between the distribution of prototypes and full dataset

2.4 Application of NLP Interpretable models on misinformation models.

In this research, we focus on one of the applications of NLP, misinformation. The rise and detrimental effects of fake news have attracted the use of deep learning models for detecting fake news. However, the model's performance alone is not sufficient [5]. Therefore in this section, we zoom into the literature related specifically to misinformation explainability. [31] in the paper "A Survey on Explainable Fake News Detection" categories the explainability of misinformation models into:

1. Explanation by social features:

Techniques that use social media platform data to achieve interpretability in fake news detection. These methods often use comments and user profile history of who shared the news article to explain why an article is classified as fake. For example, [40] proposes the dEFEND framework, which uses the co-attention mechanism by using both comments

and news content to achieve interpretability. Similarly, [28] proposed using Graph Neural Networks, which looks at more attributes compared to the dEFEND framework. Over and above the comments and news content, the methodology also uses profile history features to explain why a text was classified as fake. However, these techniques are often limited to articles disseminated through social media platforms.

2. Explanation by feature importance:

The most common techniques used to achieve interpretability in misinformation models aim to extract key features. [5, 20, 4] suggest the use of surrogate models or a combination of surrogate models to extract key features. They propose to use techniques such as LIME, anchors, SHAP, and attention mechanism with misinformation models to achieve interpretability. Given the gaps identified with surrogate methods in 2.3.2; [5] further suggests using human-in-the-loop together with the techniques. They selected a sample of people; one group was provided the output of the model's prediction, while another group was provided with both the output of the model and feature importance outcome from SHAP. The observations noted were that when humans are included to achieve explanation, it is easier for the users to trust a model and build mental models of the different categories in the future. Uniquely, [36, 32, 41] suggest using latent variables and hierarchal structures to achieve explainability in misinformation use cases. [36] used latent variables as discussed in 2.2; the latent variables are generated using the complexity contour generator and fed into a misinformation model. The relative importance of the feature group is then used for an explanation by observing any interesting patterns in the classes. On the other hand, [32, 41] use hierarchal structures to achieve interpretability in misinformation use cases. [32] uses a hierarchal attention network; the attention network looks at salient sentences to determine the most important sentences in each article. The highlighted sentences are outputs and are used to explain the model's prediction. Alternatively, [41] builds a hierarchical propagation network for each class to extract structural, temporal, and linguistic features. The features extracted from each class are compared to note if any differences exist between fake news data and true news data.

3. Explanation by news content:

These methods output the most important words based on the linguistic and semantic features from the content of the article [31]. [47] developed the xFake system to achieve this; the system is designed to analyze news items from different perspectives. The system uses PERT, which studies the linguistic properties in the news context, and ATTN, which studies the semantic properties. As a result, the system highlights important words or features based on linguistic and semantic properties to achieve explainability in misinformation articles.

In most literature, when it comes to misinformation models and interpretability, the aim is to provide a reasonable explanation to the user to increase user trust and confidence. Given that the users are not experts, it is thus paramount for the explanation to be humanly understandable [5, 31, 4].

2.5 Summary

In the sample literature reviewed, we see that there has been significant progress in the research of interpretable black box models for natural language tasks. Most approaches offer post-hoc interpretability meaning explainability is achieved once the model is trained. We determined two categories of techniques of explainability into two model-agnostic and model-specific. We find that the approaches applied to NLP models aim to understand how features interact, determine which words are the most relevant, and what linguistic properties are captured in the model. We found that example-based techniques are applied for natural language tasks but with a primary focus on counterfactual and adversarial examples. We focus on example-based methodologies, which select a subset of the data or instance of the data for interpretability in an NLP misinformation classification task. We further apply the methodology to a South African fake news data set to determine if there are any significant insights between global vs. South African fake news.

Chapter 3

Data Collection

This chapter describes the data we used for this research and how it was collected. The data used is categorised into true and fake news classes. In section 3.2, we describe the pre-processing steps that we took on our datasets, and in section 3.3, we discuss the exploratory data analysis.

3.1 Datasets

3.1.1 ISOT DATA

We use the commonly known misinformation dataset, the ISOT dataset, as the baseline dataset collected by the University of Victoria’s Information Security and Object Technology lab. The dataset consists mainly of a combination of world news and political news from the year 2015 to 2018. The data was collected from reuters.com, Politifact, and Wikipedia. In table 3.1 and 3.2 we show how the data is structured and the size of the data.

Data Attributes	Description
ID	Unique identification of document
Title	Title of document
Author	Arthur of document
Text	Full text of document
label	Class specification (1-misinformation, 0-true)
Date	Date the news article was posted(ranges between the year 2015-2018)

Table 3.1: Table showing attributes ISOT dataset

Misinformation	Size (number of documents)
Fake News	23481
True News	21417

Table 3.2: Table showing number of news articles in each class

3.1.2 COVID Dataset

The second dataset we use is a COVID dataset; we chose this dataset given the recent increase in misinformation given the COVID pandemic; the data is named COVID-FNIR and extracted from IEEE DataPort. The dataset contains news articles related to COVID news during the year 2020. It consisted of India, USA, and Europe data scraped from different sources; fake news was collected from Poynter, and true news from verified Twitter handles and news publishers. The fake and true news datasets were stored in different files as they had slightly different attributes in table 3.3. We show the attributes and also include the size of the data.

Data Attributes	Description	FakeNews	TrueNews
Date	Date the article was created(Period of 2020)	Yes	Yes
Link	Link to the article	Yes	Yes
Text	Full text of the news	Yes	Yes
Region	Region the article originated from	Yes	Yes
Country	Country the article originated from	Yes	Yes
Explanation	Explanation of why it was labeled fake news	Yes	No
Origin URL	Original link to article	Yes	No
Poynter Label	The class label provided by poynter	Yes	No
Label	Classification provided by the data collectors	0(Fake)	1(True)
size	Size of Article	3794	

Table 3.3: Table showing attributes COVID dataset

3.1.3 South African Dataset

South African dataset, which was curated with fake and real news data; the news articles were posted between the years 2017-2019. The fake news data was obtained from Zenodo Data Repository, which was data scrapped from disinformation websites in South Africa. The true news data was collected from various news media across South Africa news24, eyewitness, etc. The limitation of South African disinformation data is that it is minimal in size; we had 763 Fake news data and more than 40000 real news datasets as shown in table 3.4.

Data Attributes	Description
Date	Date article was released (ranges between the year 2017-2019)
Title	Title of document
Text	Full text of document
URL	URL to the news article
Medium	Medium news was shared

Table 3.4: Table showing attributes of South African dataset

3.2 Data Pre-processing

We perform pre-processing such that we can have quality data when applying explanation techniques. First, we reduce the number of attributes in each dataset. We want all the data sets to be standard and comprise just two columns, text and label. For the Covid-19 dataset, we had to change the class labels to align to ISOT where 0=True and 1= Fake news; we added an additional label column to the South African data to have similar class labels.

To ensure our data was ready for processing, the following steps were taken.

- Removed all empty text instances and instances with less than five words as these instances may not provide us with the necessary insight required.
- We detect non-English texts in the text data through language detection.
- We perform regular expression processing by removing HTML tags, digits, punctuation, stop-words, remove duplicate texts, and lemmatizing the text data.
- For each data set we perform further cleaning; for the ISOT data we had to remove the publication words in the text such that the models do not pick up biases; for the Covid dataset we ensure one consistent wording for the word covid and lastly for the South African dataset provided that it has imbalanced classes we performed under-sampling.

3.3 Exploratory data analysis

To analyse the dataset further, we use some exploratory text analysis techniques. We analyze prominent words and word length, perform sentiment analysis, and lastly, unsupervised topic modelling.

3.3.1 Word/length analysis

We analyse the length distribution of words and characters in each dataset by determining the average word length and vocab in our text. We compare the word count of class 1 (fake) and class 0 (True). We find that in both classes across the datasets, the average length of words of fake and real text is similar, with fake news having more outlying values as seen in figure 3.1 to figure 3.3.

To further analyse the words in each datasets we observe the top bi- and tri-gram words and word cloud to observe the most frequent words in each dataset. As observed in appendix 1, we find the most frequent words in the true dataset similar to that of fake news, which is expected as fake news text tries to disguise itself to be true and is usually around a common topic; we summarise the frequent words in table 3.5.

Data	ISOT	Covid	SA
Class 1(Fake)	President,Coun-try,Trump	COVID19,Peo-ple,lockdown	ANC,country,Police,govern-ment,Ramaphosa
Class 0(True)	President,Coun-try,Trump,Republi-can	COVID19,Test, lock-down,hospital	ANC,country,newcase,gov-ernment,ramaphosa

Table 3.5: Table showing prominent words in each datasets as observed in the word cloud in Appendix1

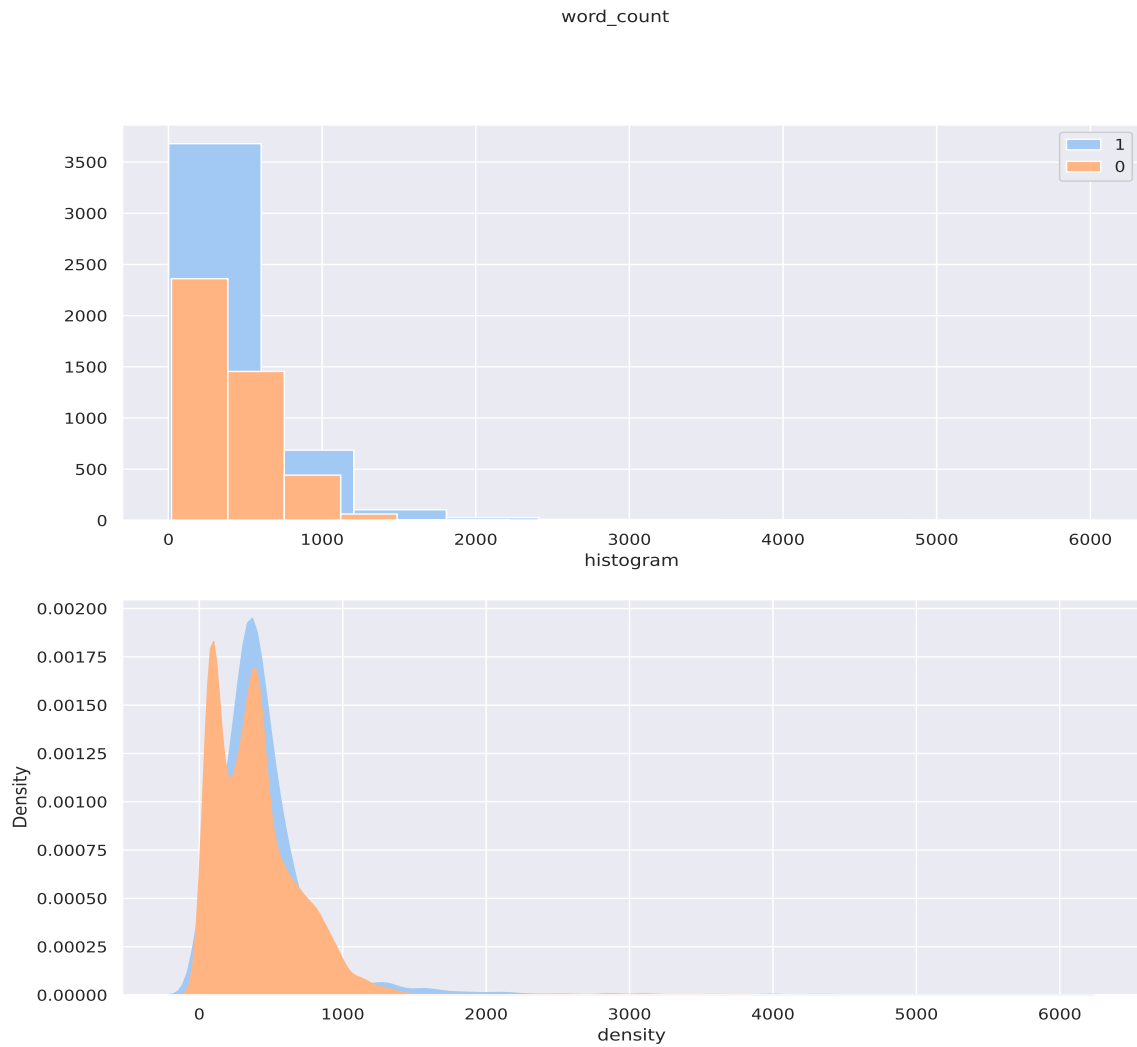


Figure 3.1: Density plot of word count for ISOT dataset

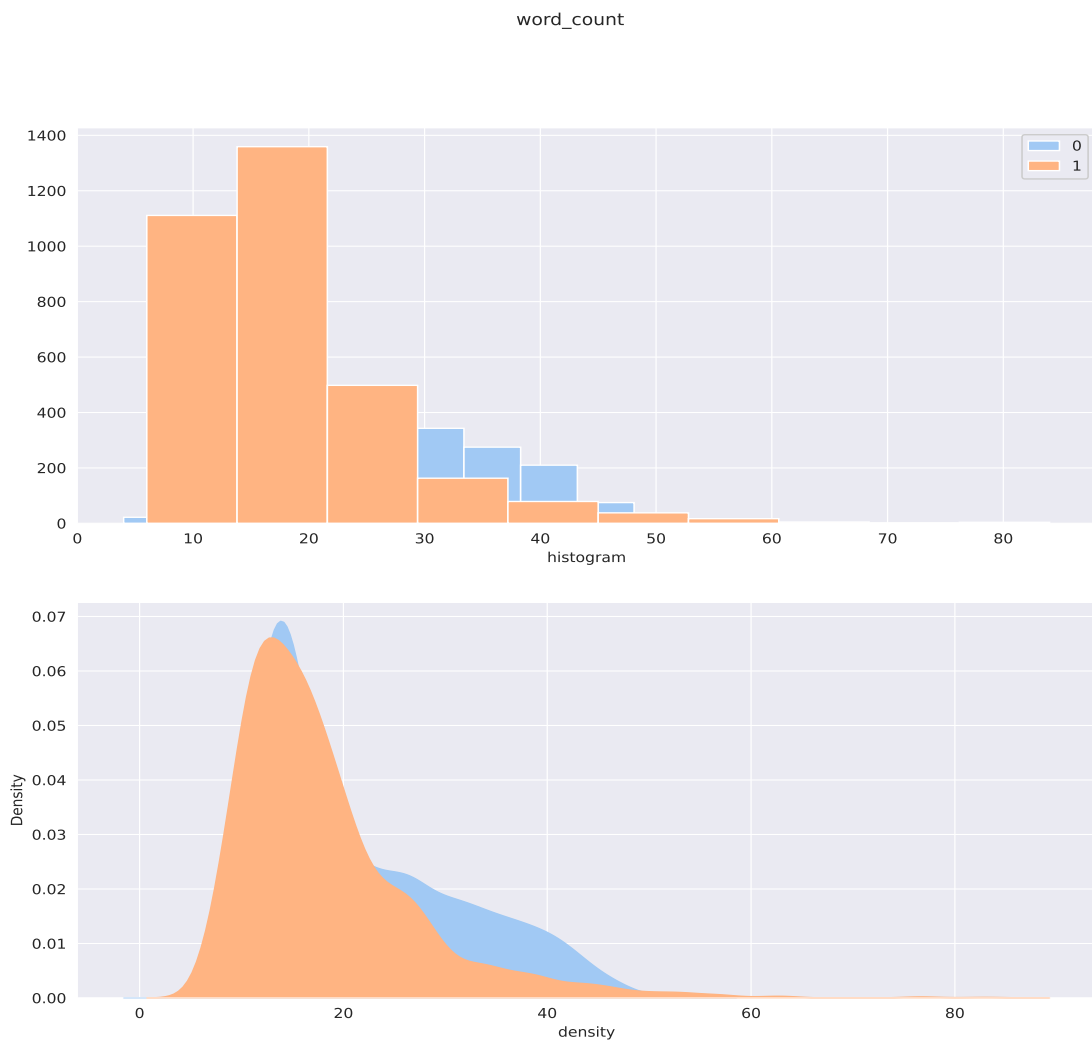


Figure 3.2: Density plot of word count for Covid dataset

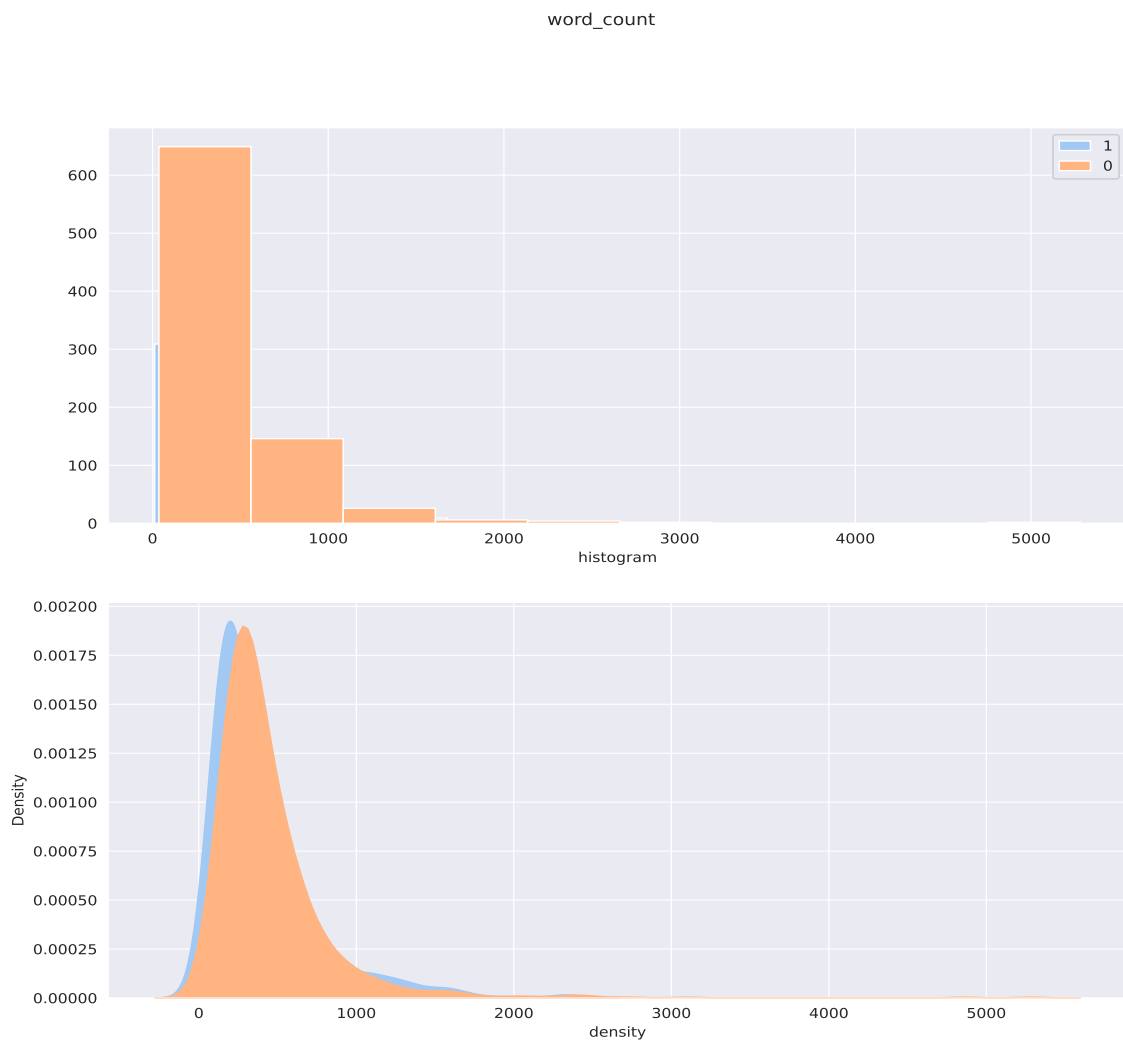


Figure 3.3: Density plot of word count for SA dataset

3.3.2 Sentiment analysis

Textblob is used to determine the sentiment of the text between the classes. Overall the sentiment of each class follows a similar distribution and is neutral. We further observe, as can be seen in figure 3.4 - 3.6 that the South African and 'Covid data has more positive outlook compared to that of the ISOT data, which may also be influenced by the vocabulary and culture in the different regions the data was collected.

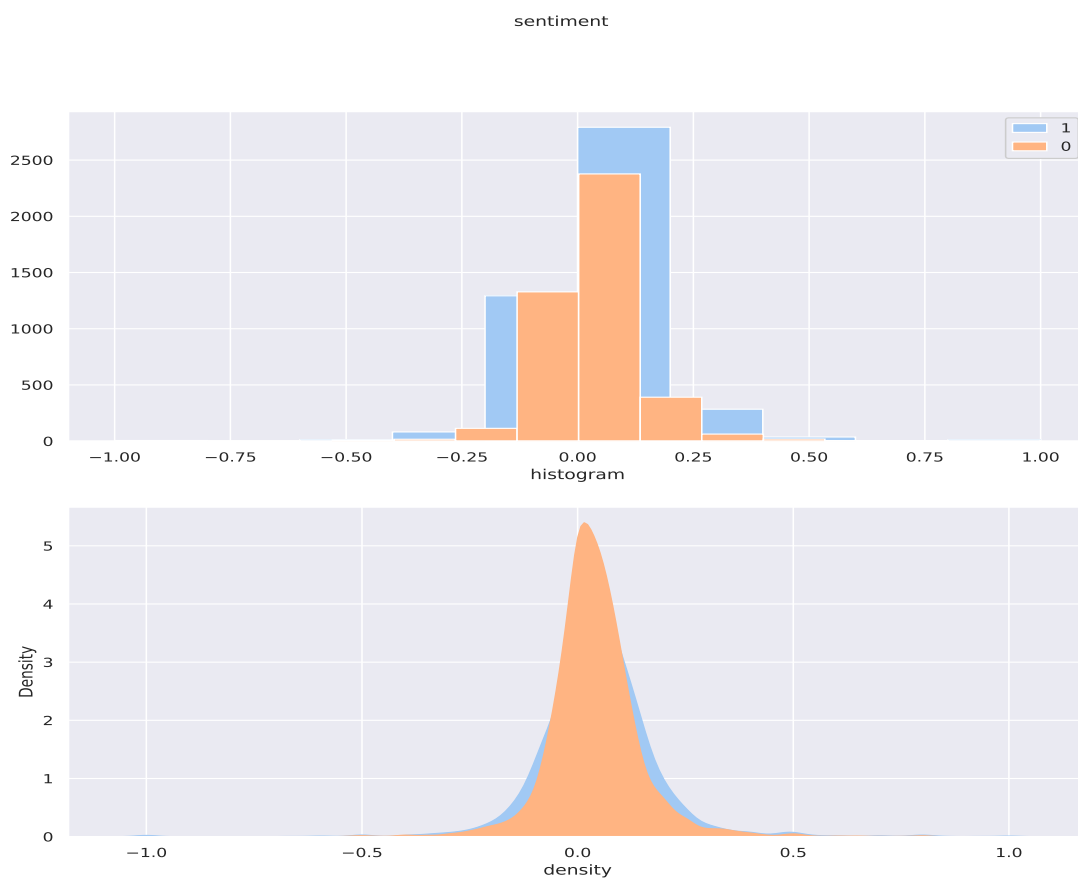


Figure 3.4: Density plot of Sentiment for ISOT Data

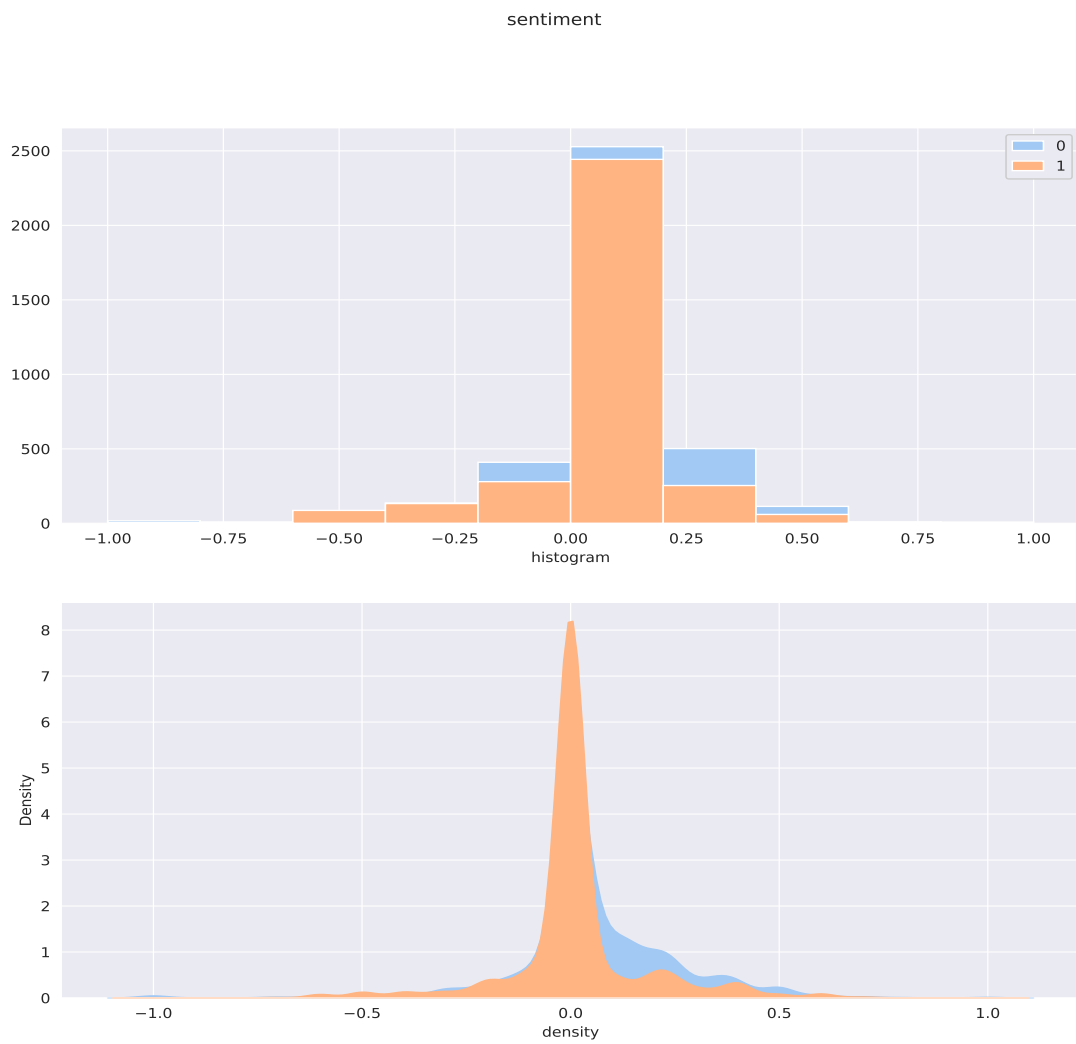


Figure 3.5: Density plot of Sentiment for Covid Data

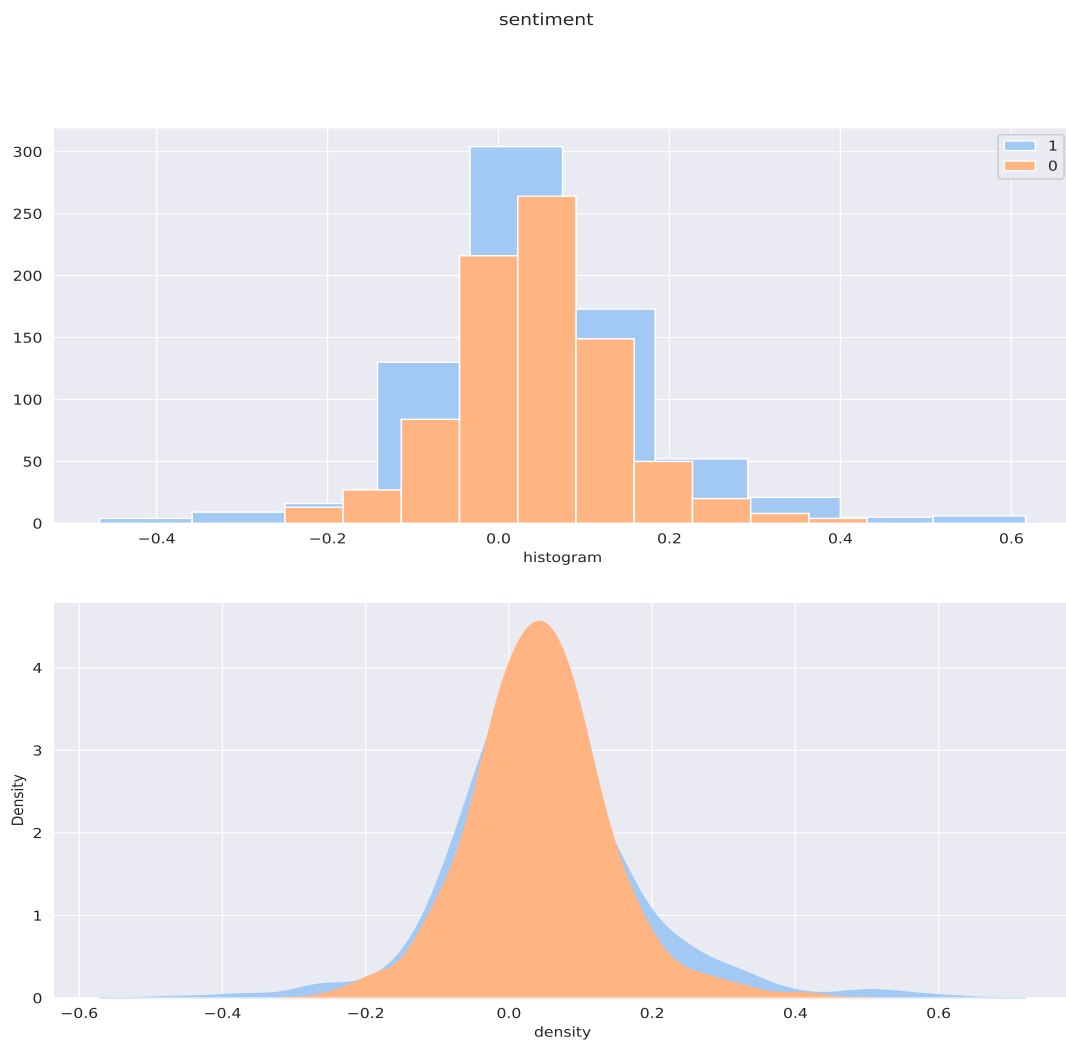


Figure 3.6: Density plot of Sentiment for SA Data

3.3.3 Unsupervised Topic Modelling

Lastly, we used the genism model to perform unsupervised bi-gram topic modelling on our dataset to determine the common themes in our data. The output can be observed in Appendix 2. We observe the themes in ISOT are centered around Presidency, American politics, and Ministers; the Covid data is centered around Covid cases, death rate, and lockdowns.

3.4 Summary

We observe that the prominent words, sentiments, and topics in fake news are similar to that of true news. Showing it is challenging to separate fake and true news by just using prominent words. We will explore this further in the coming chapters.

Chapter 4

Measuring and Comparing Attribution Type Interpretable Models

In this chapter, we explore some of the popular explainable techniques applied to NLP misinformation classification tasks. We use the baseline dataset ISOT to obtain results from some of the most popular techniques for interpretable machine learning. We discuss each technique and compare the output results.

4.1 Logistic Regression & Decision Trees

We first explore intrinsic approaches, logistic regression, and decision trees that were traditionally used on NLP classification tasks to achieve interpretability [27]. We first tokenize our datasets before we train them using the models. We explore two tokenizing approaches, TfidfVectorizer, and word2vec embeddings. We further explore uni-gram and tri-gram tokenization to be able to extract phrases rather than single words.

Logistic regression

Once we have tokenized our text data, we use a logistic regression pipeline to train both uni-gram and tri-gram tokenized data. Given that once the model has been trained and tested, there is no further pre-processing required, we interpret the model's decision by extracting the learned model coefficients.

From the coefficients, we can determine which words and phrases were the most influential in making a model decision. In figure 4.1 and 4.2 we show the most positive and negative words for the model in the uni-gram and bi-gram case.

Weight?	Feature
+0.661	century
+0.592	america
+0.579	hillary
+0.533	press
+0.419	fox
+0.418	video
+0.416	isis
+0.411	watch
+0.407	american
+0.399	breitbart
...	65000 more positive ...
...	28587 more negative ...
-0.400	presidential
-0.416	minister
-0.418	ministry
-0.419	reporter
-0.423	clintons
-0.426	representatives
-0.431	spokesman
-0.461	obamas
-0.503	statement
-1.191	trumps

Figure 4.1: Top positive and negative features for uni-gram logistic regression

Weight?	Feature
+1.373	video screen capture
+1.355	donald trump realdonaldtrump
+1.126	black lives matter
+1.105	president united states
...	1049497 more positive ...
...	921118 more negative ...
-0.808	us secretary state
-0.888	chancellor angela merkel
-0.920	respond request comment
-0.944	us presidentelect donald
-0.973	president vladimir putin
-0.988	islamic state militant
-0.990	minister theresa may
-1.000	prime minister theresa
-1.015	president tayyip erdogan
-1.020	president robert mugabe
-1.190	presidentelect donald trump
-1.264	president barack obamas
-1.480	us house representatives
-1.674	us president donald
-1.833	president donald trump
-1.999	president donald trumps

Figure 4.2: Top positive and negative features for tri-gram logistic regression

We observe that words and phrases with president name's and ones that mention American politics negatively contribute to the model's decision.

We further perform local explanation for a single instance and compare the explanation between the tfidf and word2vec. In figure 4.3 and 4.4 we observe that the explanation for the instance is identical in both cases indicating that the logistic regression as an explainer is faithful; there is a slight difference in the probability and score and some words are highlighted darker in word2vec compared to Tfidf meaning they assigned more importance in the explanation.

y=1 (probability 0.999, score 7.575) top features

Contribution?	Feature
+2.879	Highlighted in text (sum)
+1.425	<BIAS>
	... 67 more positive ...
	... 32 more negative ...

no one happier see sarah palin back stephen colbert following former alaska governor endorsement donald trump republican nomination colbert plenty material make fun woman called original material girl god missed magical eagle made wish flag pin came life colbert palin reappearance which great trump admittedly given much material make joke nobody compare original material girl colbert played palin socalled greatest hit referring bizarre statement palin made community organizer obama drill baby drill gonna kick isis as colbert sarah palin guaranteed donald trump evangelical vote think speaking tongue while donald trump frequent target colbert one come close original material girl palin primary target colbert year run colbert report comedy centralthere truly lot make fun seems palin gone deep end lately ever she nerve blame president son delinquency rambling speech give clear impression someone seem thing togetherin case trump apparently excited palin endorsement position administration possibly vice president sound quite frightening but let laugh twofeatured youtube screenshot

Figure 4.3: Local explanation for Tfidf logistic regression model

y=1 (probability 0.996, score 5.574) top features

Contribution?	Feature
+2.111	Highlighted in text (sum)
	... 73 more positive ...
	... 26 more negative ...

no one happier see sarah palin back stephen colbert following former alaska governor endorsement donald trump republican nomination colbert plenty material make fun woman called original material girl god missed magical eagle made wish flag pin came life colbert palin reappearance which great trump admittedly given much material make joke nobody compare original material girl colbert played palin socalled greatest hit referring bizarre statement palin made community organizer obama drill baby drill gonna kick isis as colbert sarah palin guaranteed donald trump evangelical vote think speaking tongue while donald trump frequent target colbert one come close original material girl palin primary target colbert year run colbert report comedy centralthere truly lot make fun seems palin gone deep end lately ever she nerve blame president son delinquency rambling speech give clear impression someone seem thing togetherin case trump apparently excited palin endorsement position administration possibly vice president sound quite frightening but let laugh twofeatured youtube screenshot

Figure 4.4: Local explanation for word2vec logistic regression model

Decision Tree

With the decision tree, the model’s output is a rule-based global explanation. During the training phase, we limit our trees to a depth of 5. The results of the decision tree can be shown in figure 4.5.

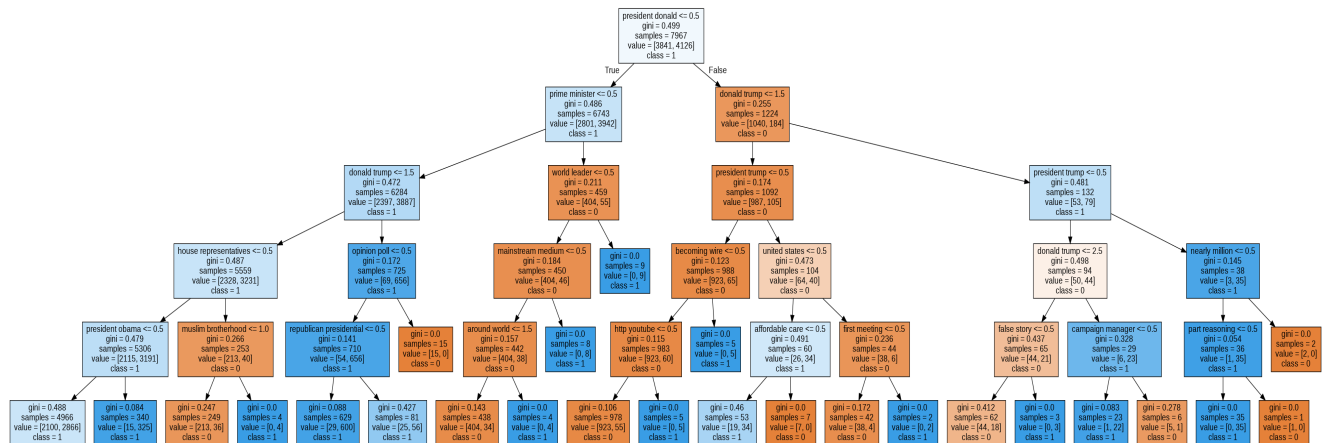


Figure 4.5: Local explanation for word2vec logistic regression model

The phrase used to make decision rules in the model is also related to the presidency and politics from the USA, similar to the logistic regression. However, we also note that the decision rules for fake and true news classes are very similar and making it difficult to distinguish a clear decision rule between the classes..

4.2 LIME (Local interpretable model-agnostic explainer)

We apply the LIME surrogate model as explained in chapter 2. We first train an LSTM (long short-term memory) network on our ISOT dataset to classify misinformation. Once our model has been trained and tested, we apply the LIME methodology to derive a local explanation for a single instance. The steps we followed are:

- Use word2vec embeddings on the cleaned dataset.
- Train LSTM
- We select an instance of interest x_{10} from our test dataset for an explanation, similar datapoint used across the techniques.
- Select perturbed points that are in the proximity of x_{10}
- We use LIME to generate explanations.

The output of the explanation can be seen in figure 4.6. The words with the highest weights for fake and real news are highlighted, with orange indicating importance for fake and blue real news. The darker the color, the more significant the feature is in explaining the outcome.

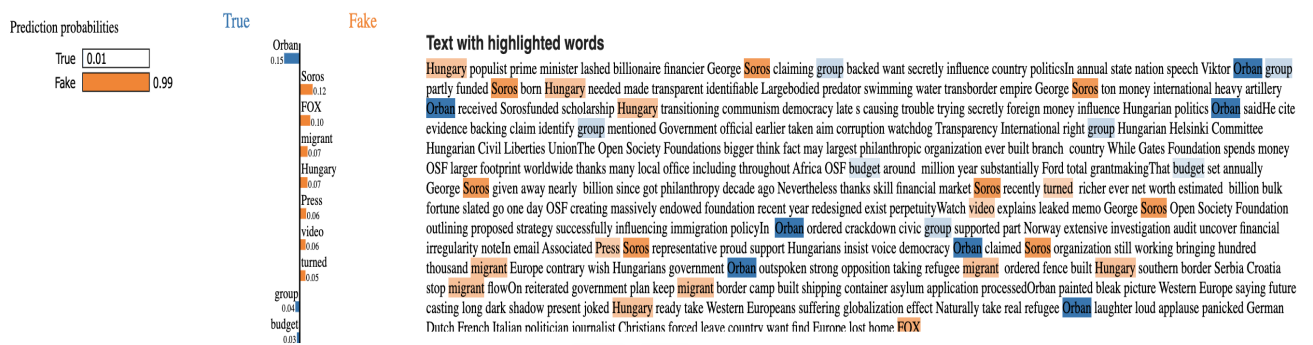


Figure 4.6: Local explanation for word2vec LIME model

SP-LIME

We further use SP-LIME to derive a global explanation. We pick a representative sample in each class that applies LIME to each instance in the chosen samples to determine the most common explanation for each category. We observe that true news explanations are words with dates, facts, and neutral words; Fake news explanations are usually related to the government and are politically connected. The results of SP-LIME can be seen in Appendix C.

4.3 SHAP Values

Given that LIME weights cannot be trustworthy in determining the importance of a feature [33], we apply SHAP values as described in chapter 2. We observe the outcome in figure 4.7, which shows the top 20 features based on their importance. The weight of each variable based on the average of its combinations across the feature space is the x-axis and shows whether the variable positively or negatively affects the predicted outcome.

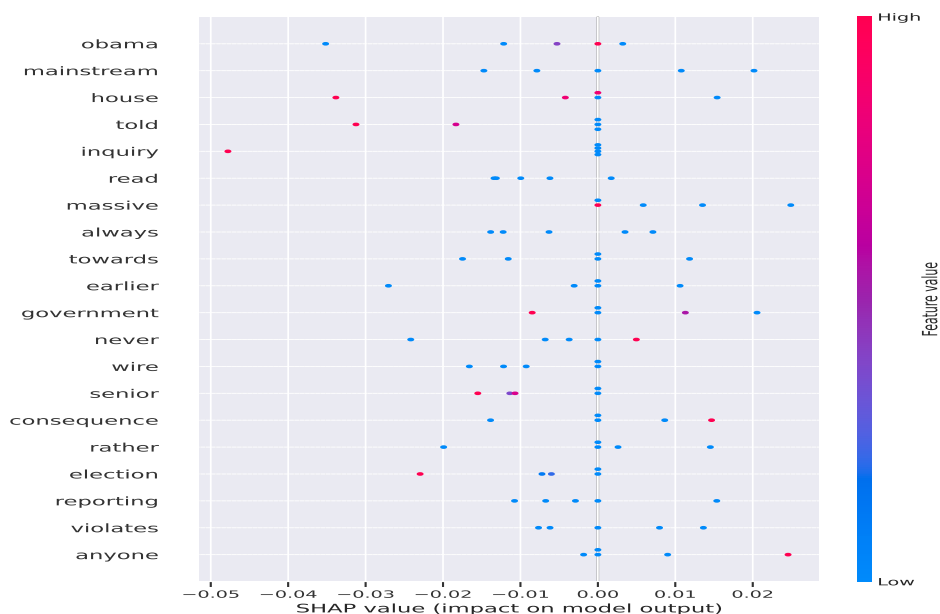


Figure 4.7: SHAP explainer on ISOT data

4.4 Attribution values: Integrated gradients

Lastly, we applied attribution methods to observe the outcome of the explanation technique. We used the same LSTM model applied in LIME; however, for integrated gradients, we create a baseline, empty embedding, and calculate the gradient between the baseline and the input. In figure 4.8 we observe the results of an explanation for class label 1 (fake news). It shows the words or features with the most substantial attribution score, with darker colors indicating more significant attribution scores compared to the lighter color.

us senate flag pelosi lead week company many credit regarding us senate replace nothing light direction UNK senate us full shooting finance officer calling money iran unbelievable saudi used campaign us senate meeting ahead thousand but turkey opposition crisis conspiring complain legislation fight at used state october called security house number top game conspiring complain used oil recommend last officer used thousand but victory organization congress cambodia at edt engineering engineering used oil donald us senate used disappointed penalty counterpart citing iowa iowa pundit citing attack year money attack true attack panel UNK democratic voter war used called house times

Figure 4.8: Attribution map using integrated gradients explanation for class label 0

4.5 Quantitative results of trained models

In table 4.1 we show the accuracy score of the models trained, indicating how well the models can predict unseen fake news data. As expected, we observe low accuracy scores for the Logistic regression and Decision tree model, models which are intrinsically interpretable as described in 2.2. When interpretability is achieved in black-box models through surrogate and model-agnostic techniques (LIME, SHAP, and Integrated gradients), we see the model's performance is not compromised. This shows the trade-off between accuracy and intrinsic explainable models.

Table 4.1: Table showing the accuracy values of the trained models to achieved interpretability in section 4.

Logistic Regression	Decision Tree	LIME LSTM	SHAP LSTM	Integrated gradient LSTM
89%	62%	96%	96%	96%

4.6 Summary

Majority of the results of the explanations discussed in this section output attribution/saliency maps highlighting the most important words(features) that are important in making the decision. However, although we found some explanation techniques to be faithful to a lay-user, the words may not be sufficient to distinguish and interpret between the two classes as we find fake news class to have similar words to true news. We thus try to use an example based technique in conjunction with one of the techniques to try and identify any key characteristics between the two classes.

We further observe the trade-off in accuracy and interpretability in table 4.1 as the accuracy values of intrinsic interpretable models are far less desirable than that of model-agnostic techniques.

Chapter 5

Methodology

In chapter 4 we discovered that most interpretable model outputs are attribution maps; we aim to use example-based techniques such that the explanations are more humanly understandable. We use four techniques that are commonly used on image data. First, we scale the techniques to our misinformation data to determine if we cannot extract any characteristics that could be used to interpret between the two classes. Further, we choose the best technique by analysing the quality of the examples produced. In this section, we discuss the methodology we used.

5.1 Experimental Design

The four techniques explored are Clustering, Set-Cover, Influential instances, and MMD-Critic. The techniques can be summarised as prototypical techniques. Prototype methodologies output a subset of the dataset. The subset selected is considered the best representation of the dataset. These subsets are then used to explain the data class before model prediction or after; explanation can be achieved by extracting the commonality and characteristics of each prototype class. Further, they can also be used to understand and investigate complex data [21]. For image data, prototypical examples are easy to digest and distinguish the characteristics that make up a class; however, with text data, it may not be and may be overwhelming to explain. Because of this, we further process the text prototypes extracted using text analytics techniques and employ one of the

model-agnostic techniques described earlier. The design steps we follow are described high-level in figure 5.1.

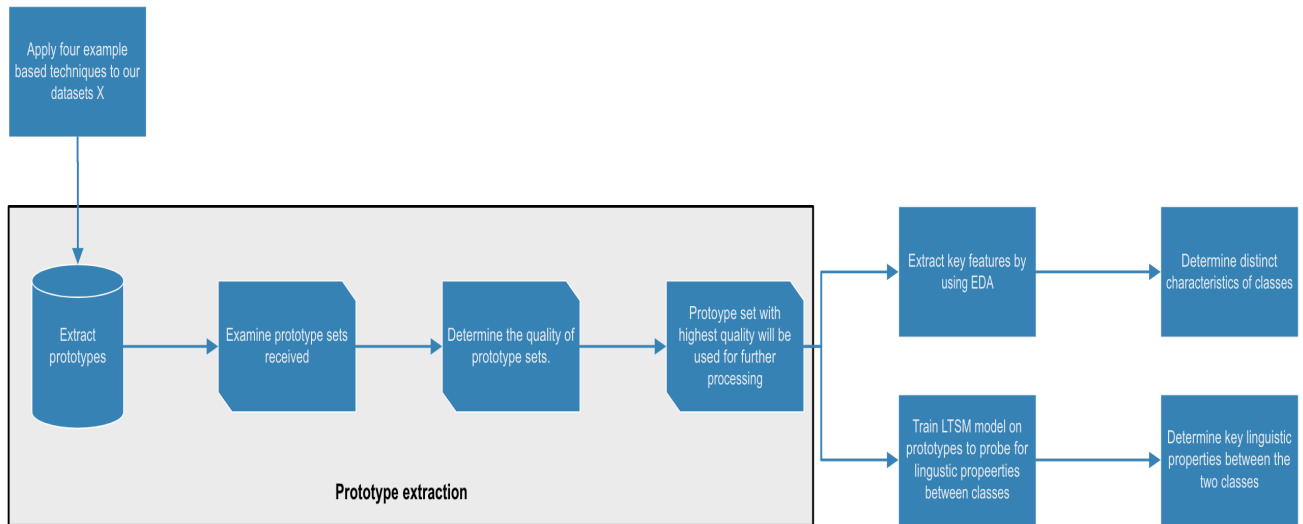


Figure 5.1: High-level experimental design steps taken to achieve results

5.2 Algorithms

5.2.1 K-nearest neighbour Clustering Algorithm

Clustering is one of the earliest techniques, for example-based methods. The approach splits the data into clusters or classes based on their proximity to each other. A new instance x_i is assigned to the cluster to which it has the closest proximity based on K centroids. The advantage of the clustering method is that it can be used both as a method to extract prototypes and as an intrinsic interpreter in its own right; however, the disadvantage is that it can not achieve interpretability on a modular level (global explainability) just on a local level.

For text data, the similarity score between documents is used to determine the clusters of the dataset. We utilise the K-nearest neighbour clustering technique to determine the set of prototypes for each class. We show that running the algorithm on each class respectively, as stated in [9], yields prototype sets with the K instances chosen for an instance prediction being the set. Given that interpretability cannot be achieved on a modular level, the prototype sets are acquired through the prediction phase of test instances.

Given a dataset X with class label Y , where $y_i \in \{1, 0\}$ we split the data X into X^{train} and X^{test} . For each $x \in X^{test}$ we determine which K documents in X^{train} are closest to the instance x ; the K selected documents are the prototype sets for the selected instance. We repeat for all instances in X^{test} and obtain a list of documents K , which we make our set of prototypes. The algorithm can be seen in algorithm 5.1 below.

```
Start with an empty set  $K = \emptyset$ , given dataset  $X^{train}$  and instance  $x \in X^{test}$   
while  $K \leq 10$ :  
    Find  $x_i^{train}$  which is the most similar to  $x$   
     $K \cup x_i^{train}$   
end while
```

Algorithm 5.1: Clustering Prototype selection method

5.2.2 Set Cover Algorithm

[9] proposed the set cover approach to select prototypes as an improvement to the K-clustering techniques. As the name suggests, the methodology leverages the set cover theorem, which states that given a set X what are the optimal subsets S which cover the universal set X by minimising the cost. Compared to the above clustering technique, the set cover approach automatically selects the number of prototypes for each class and can achieve both modular and local interpretability

The selection of prototypes using the technique can be made via two approaches the integer program and the greedy approach. We make use of the greedy approach.

An instant $x_i \in X$ is selected as a prototype if it encompasses the following characteristics:

- In its sphere, it should cover more training points that have the same class as itself.
- It should cover fewer instances of the opposite class.
- It must be sparse

To determine the prototype set P_l where P_l is the prototype set of class l . We require a sphere size dependent on a radius value; the sphere and radius are used to determine each instance's coverage. The size of the sphere is calculated by using a pairwise distance formula; we want to minimise the cost of covering a new instance.

Below in figure 5.2 we show an image of the selected prototypes adapted from [15] which shows the greedy algorithm for a 2-class toy problem using a radius of 1.5. The prototypes chosen for each class are bolded as their coverings include the most instances which belong to the same class and less of the other class.

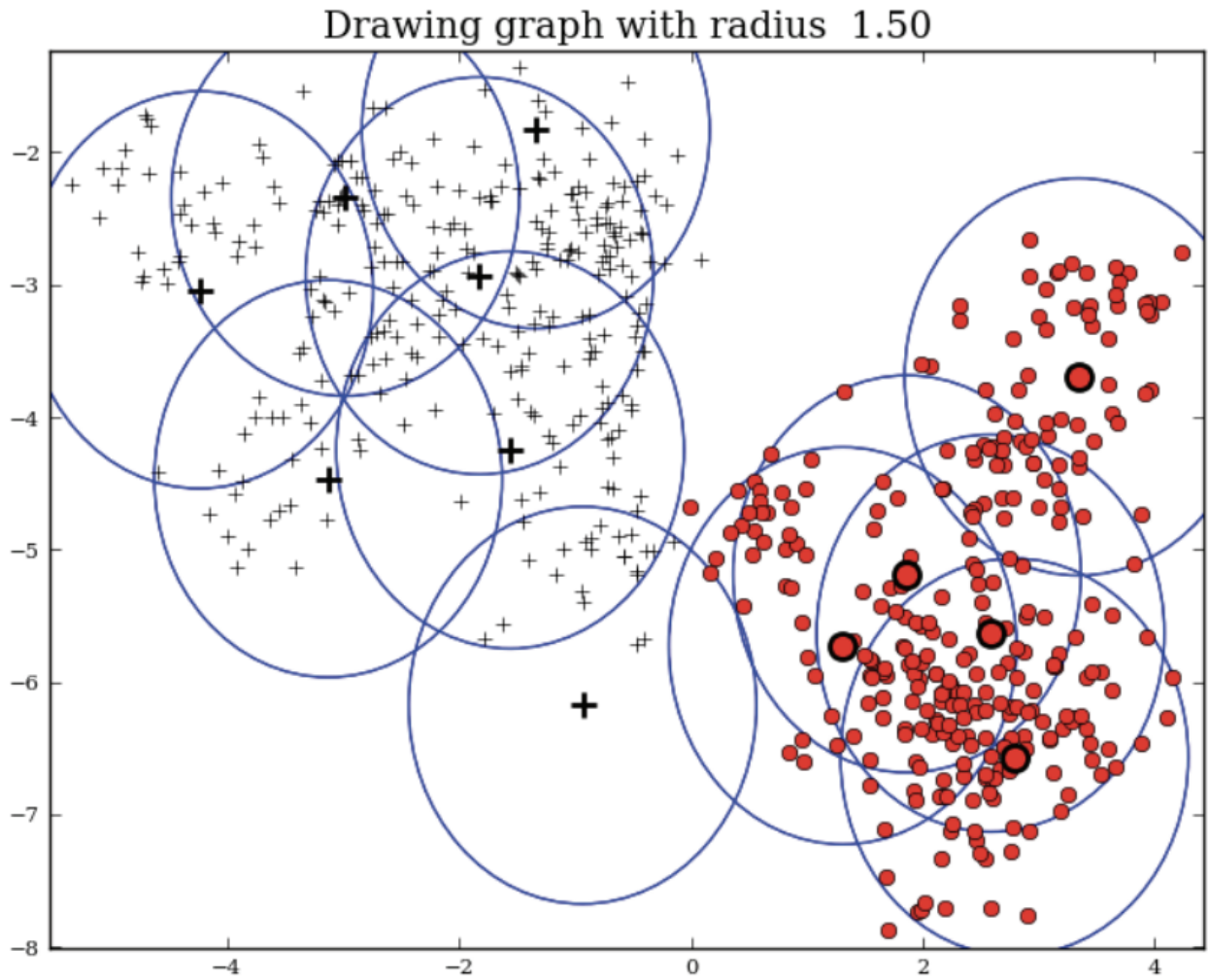


Figure 5.2: Image showing Set Cover example for a classification toy example

5.2.3 Influential Instances

Influential instances is an example-based technique that does not necessarily output a set of instances but calculates what influence each instance has on the model parameters. An influential instance changes the model parameters and decision when deleted; [26] states that the counterfactual question that is asked is: "how would the model's predictions change if we did not have this training point?". The method is used mainly for debugging and to explain behaviour of the model.

There are two approaches one can take to determine influential instances:

- Deletion: In this approach, one needs to iterate through the training data and delete each instance; once deleted, analyse how the model parameters change by retraining the model.
- Calculation of influence function: The influence function determines which instances are the most influential to the model by up-weighting the instance by a small amount and determines the change in the model parameters.

We use the calculation of influence function as it is known to be more robust [3, 26]; further, the approach also has the benefit of highlighting which instances are the most harmful to the prediction of a particular test instance. To use the influential function, one needs to train a model with a loss function that is twice differentiable and a loss gradient that is accessible with respect to model parameters. The influence score calculation is based on the loss function.

Given a dataset X , which maps to a label set Y , we split the dataset into training and test points. With $x_1, x_2, \dots, x_n \in X^{train}$ and $y_1, y_2, \dots, y_n \in Y^{train}$ we state that $z_i = (x_i, y_i)$ as being the training point for instance i . The loss function is then defined as $\frac{1}{n} \sum_{i=1}^n L(z_i, \theta)$, where θ is the model parameters. By up-weighting an instance z_i by a small amount ϵ we get a new loss function: $\frac{1}{n} \sum_{i=1}^n L(z_i, \theta) + \epsilon L(z, \theta)$.

To determine the influence measure, which measures the change of up-weighting a training instance, z_i by ϵ , the gradient of the loss and inverse of hessian is used as shown in the below equation 5.1:

$$I(z) = -H_{\hat{\theta}}^{-1} \nabla L(z, \hat{\theta}) \quad (5.1)$$

Where:

$H_{\theta} = \sum_{i=1}^n \nabla_{\theta}^2 L(z_i, \hat{\theta})$ is the Hessian of the loss function and twice differentiable.

We use the CNN network to train our data with cross-entropy loss. Then for each instant in the training data, we up-weight the loss function by a small amount ($\frac{1}{n}$) and determine their influence score as described in equation 5.1. We then use the top 100 most influential instances for each class as our prototype sets.

5.2.4 MMD-Critic Algorithm

MMD-critic is a quantitative approach that uses maximum mean discrepancy, a greedy approach in selecting prototypes. It aims to select a subset of prototypes Z that best represent our dataset X such that the probability distribution of the data points in Z approximates that of X . This approach is one of the preferred approaches for selecting prototypes [33, 25] given that together with prototypes, it outputs critics, which are data points that least represent a class and has a high probability of being misclassified.

To get the prototype and critic sets, the following attributes are required:

1. **m* and n* number of prototypes and critics:**

The number of prototypes and critics can be selected manually or alternatively use ϵ an error term, such that the algorithm stops searching once the MMD is above the threshold of the error.

2. **Kernel function:**

We use the kernel function in the MMD formula to determine the distance between two points. The kernel ranges between 0 and 1. Values close to 0 indicate the points

are infinitely apart, while a value closer to 1 indicates similar points. The Euclidean distance is used in our kernel, and we use the radial basis kernel, a popular kernel used across the literature.

3. MMD:

The maximum mean discrepancy function is generally used to determine the "closeness" of two distributions. We use it to determine how well our prototype distribution approximates the data i.e., we will add data points that provide the lowest MMD compared to other samples in the data. The equation of the MMD is shown in equation 5.2:

$$MMD^2 = \frac{1}{m^2} \sum_{i,j=1}^m k(z_i, z_j) - \frac{2}{mn} \sum_{i,j=1}^{m,n} k(z_i, x_j) + \frac{1}{n^2} \sum_{i,j}^n k(x_i, x_j) \quad (5.2)$$

Where:

z_i is the i^{th} data point in prototype sample set.

x_i is the i^{th} data point in data set

k =Kernal measure

m = number of prototypes and

n = number of data points in the dataset

The goal is to optimize the function so that the function's middle term approximates to zero.

4. Witness Function:

The witness function is only used once all prototypes are selected. The purpose of the function is to search for a subset of data points n , which are not represented well by the data (outliers). It measures the proximity of each remaining data point to the dataset as well as its proximity to the chosen prototype set. If the witness function value is close to zero, it means the instance is not a critic, but instances with a large positive or negative witness value are a critic. Having a large positive and negative witness function is an indication that the instance either underestimates or overestimates the data distribution; the equation is shown in equation 5.3

$$W = \frac{1}{n} \sum_i^n k(x, x_i) - \frac{1}{m} \sum_{j=1}^m k(x, z_j) \quad (5.3)$$

where:

z_j is the j^{th} data point in prototype set.

x_i is the i^{th} data point in data set

The algorithm of using the MMD-critic approach for selecting prototypes and critics is described in algorithm 5.2 and 5.3.

Start with an empty set $Z = \emptyset$, given dataset X

while $Z \leq m$:

 Find x_i which reduces the MMD^2

for all x_i in the X **do**

 Calculate the MMD^2

 Select x_i which optimises MMD^2

$Z \cup x_i$

end while

Algorithm 5.2: MMD^2 Prototype selection algorithm

Start with an empty set $C = \emptyset$, given dataset $X \setminus Z$

while $C \leq n$:

 Find x_i with largest $|W|$

for all x_i in the $X \setminus Z$ **do**

 Calculate the W

 Select x_i with largest value

$C \cup x_i$

end while

Algorithm 5.3: Critic selection algorithm

5.3 Prototype Analysis

In this section, we describe how we analyse the prototypes extracted from the data to determine characteristics that can be used to associate and explain the classes.

Prototype Quality

To determine the quality of the prototypes, we use the trained LSTM model on unseen data which was not selected as part of the prototypes. The accuracy score of how well the model predicted will help us understand the quality of chosen prototypes.

Determining class characteristics from prototypes

To determine class characteristics from prototypes, we apply various techniques. First, we extract top key phrases from each class using Tfidf and Word2Vec. We also determine the sentiment and perform opinion detection on the extracted prototypes. Further, we perform a comparative analysis on the part of speech used, word length, and special characters to determine any significant character differences between the two classes.

Using SP LIME in conjunction with prototypes

We employ the global explainer SP-LIME to determine how the model-agnostic technique would explain prototype classes, Such that we can have an explainer that is truthful, portable, and can be understood by a human to a degree.

5.4 Summary

This chapter discusses the approaches taken to extract, analyse and explain the prototypical examples. We highlighted the algorithm used and the steps that were taken to extract the necessary results, which we discuss in chapter 6.

Chapter 6

Results

In this chapter we discuss the results that we retrieve from the data and analysis performed as described in chapter 5.

6.1 Prototypes Selected

We apply the algorithms described in our methodology section to each of our datasets; ISOT, COVID, and SA data. We analyse the prototypes selected for each data using the algorithms and determine how common the sets are to each other.

6.1.1 Clustering Sets

We use our test data to acquire our prototype set for clustering. For each instance in the test data, we select the top 20 documents it is similar to; we then remove any duplicate indices chosen for more than one test indices to get a unique subset. After applying the approach to our datasets, we find that for all datasets, the fake news set is larger than the true news set by at least a margin of 12, as shown in table 6.1. This may indicate that fake news has more variety compared to true news data. In table 6.2 - 6.4, we also observe that the prototype sets chosen for ISOT and COVID match the Set Cover sets highly 43% and 47%, while SA has a match rate of 51% with MMD-critic.

When looking at the phrases extracted and word cloud for each set figure 6.1 - 6.6, we observe that the sets are closely related to each other and may be difficult to distinguish between the two classes. This is especially the case for ISOT classes; however, for Covid and SA, we can see that fake news phrases are centered around controversial topics and show subjectivity.

Data Set	Size of True Set	Size of Fake Set
ISOT	29	43
COVID	24	33
SA	20	32

Table 6.1: Table showing the size of each prototype set using clustering technique for our dataset

	Clustering Set	Set Cover	Influential	MMD-Critic
Clustering Set	100%	43%	21%	19%
Set Cover	43%	100%	41%	31%
Influential	21%	41%	100%	51%
MMD-Critic	19%	31%	51%	100%

Table 6.2: Table showing the match rate of the prototype sets for ISOT data

	Clustering Set	Set Cover	Influential	MMD-Critic
Clustering Set	100%	47%	33%	22%
Set Cover	47%	100%	44%	29%
Influential	33%	44%	100%	56%
MMD-Critic	22%	29%	56%	100%

Table 6.3: Table showing the match rate of the prototype sets for Covid data

	Clustering Set	Set Cover	Influential	MMD-Critic
Clustering Set	100%	42%	41%	51%
Set Cover	42%	100%	49%	33%
Influential	41%	49%	100%	52%
MMD-Critic	51%	33%	52%	100%

Table 6.4: Table showing the match rate of the prototype sets for SA data



Figure 6.1: Word Cloud class 1(Fake)for ISOT

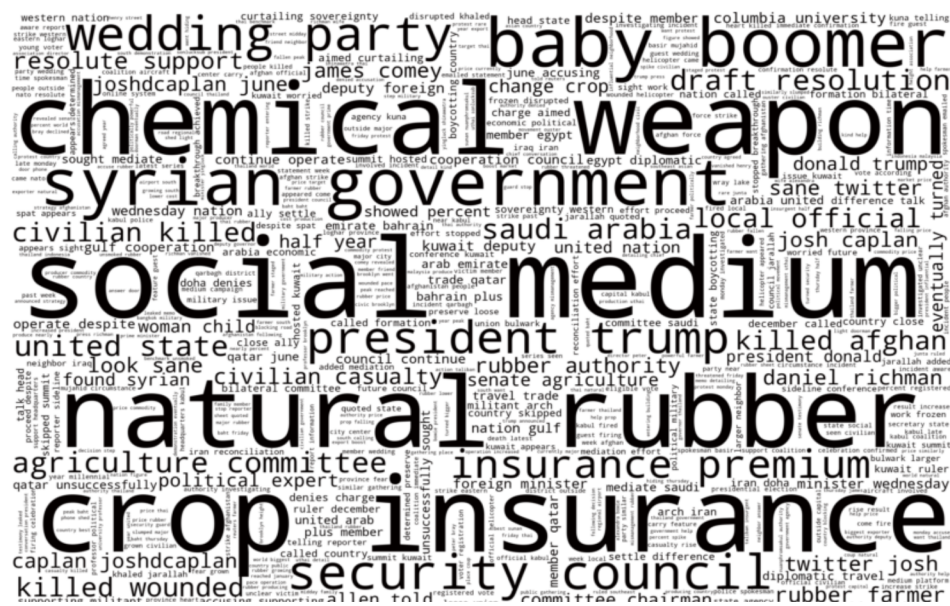


Figure 6.2: Word Cloud class 0(True) for ISOT.

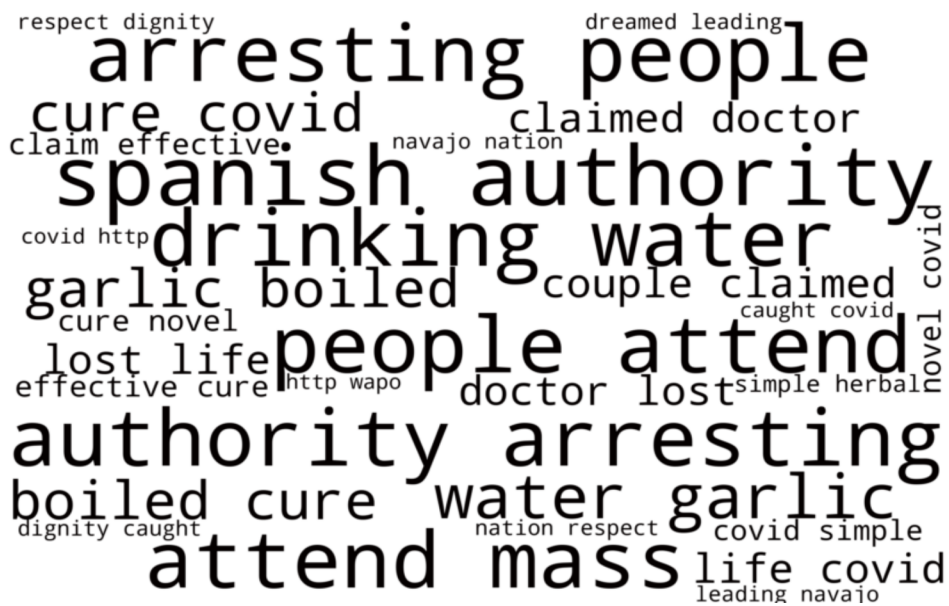


Figure 6.3: Word Cloud class 1(Fake)for COVID Data

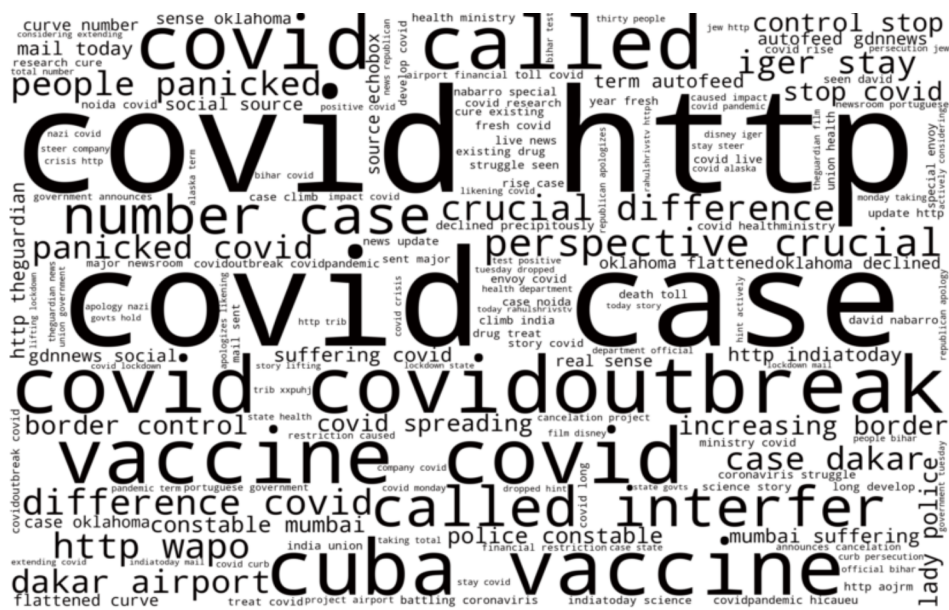


Figure 6.4: Word Cloud class 0(True) for COVID Data



Figure 6.5: Word Cloud class 1(Fake)for SA Data

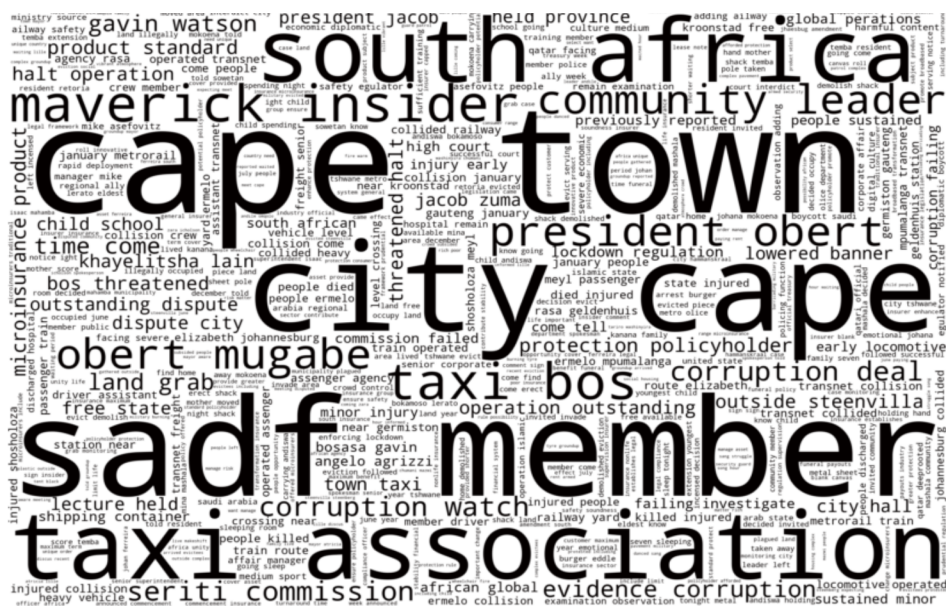


Figure 6.6: Word Cloud class 0(True) for SA Data

6.1.2 Set Cover Sets

Set Cover has a larger set of prototypes selected compared to clustering, as shown in table 6.5. Unlike clustering, we observe that the prototype sets for Set Cover are slightly larger for true news compared to fake, but given that the variance is smaller between the classes, it indicates similar variation.

When observing the matching rate, table 6.2-6.4, of each set we observe that ISOT and COVID sets have a higher match rate with clustering sets whilst SA set has higher rate with Influential instances set. Looking at the word cloud phrases in figure 6.7-6.12, similar to clustering, the top phrases can be difficult to distinguish between classes; however, for SA and COVID fake news word cloud we notice phrases such as farm attack and farm murderer and around covid cures which are controversial and subjective compared to true news phrases.

Data Set	Size of True Set	Size of Fake Set
ISOT	230	225
COVID	174	171
SA	123	119

Table 6.5: Table showing the size of each prototype set using Set Cover technique for our datasets

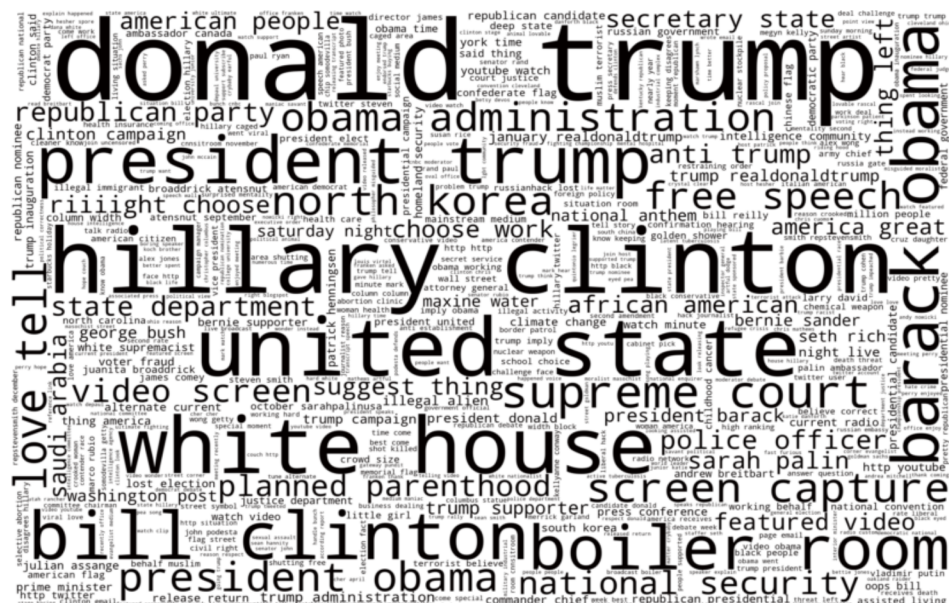


Figure 6.7: Word Cloud ISOT prototype output for class 1(Fake) using Set Cover

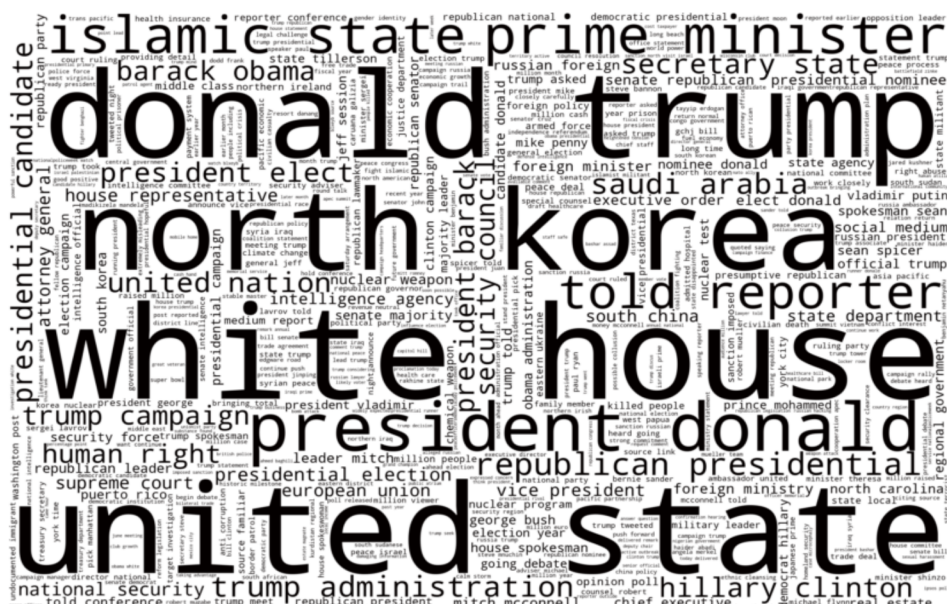


Figure 6.8: Word Cloud ISOT prototype output for class 0(True) using Set Cover

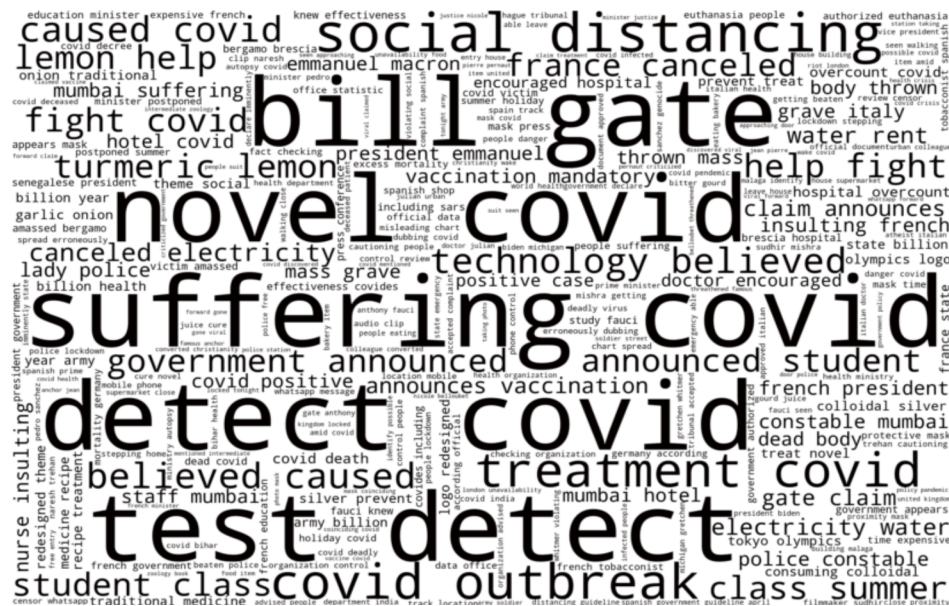


Figure 6.9: Word Cloud Covid prototype output for class 1(Fake) using Set Cover



Figure 6.10: Word Cloud Covid prototype output for class 0(True) using Set Cover



Figure 6.11: Word Cloud SA prototype output for class 1(Fake) using Set Cover



Figure 6.12: Word Cloud SA prototype output for class 0(True) using Set Cover

6.1.3 Influential function sets

For influential instances, the prototype sets are not automatically chosen. The algorithm will tell us which instances are the most influential and harmful. We then choose the top 100 instances, which are the most influential, and the top 20 harmful instances. From the top 100 influential examples, we show the split per class in table 6.6 and observe that the true news class makes up 47% of the set and fake news 53% for ISOT and SA data, similarly for COVID set we find true news is 44% and fake news 56%. We observe that the top 100 influential instances have a higher match rate with MMD-critic sets.

Data Set	Size of True Set	Size of Fake Set
ISOT	47	53
COVID	44	56
SA	47	53

Table 6.6: Table showing the size of each prototype set using Influential technique for our dataset

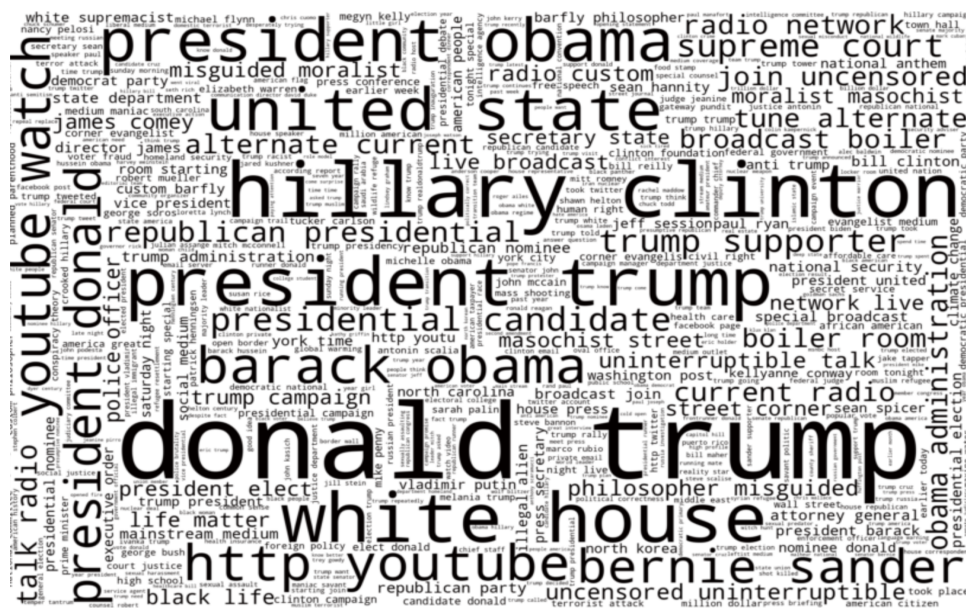


Figure 6.13: Word Cloud ISOT prototype output for class 1(Fake) using Influential instances

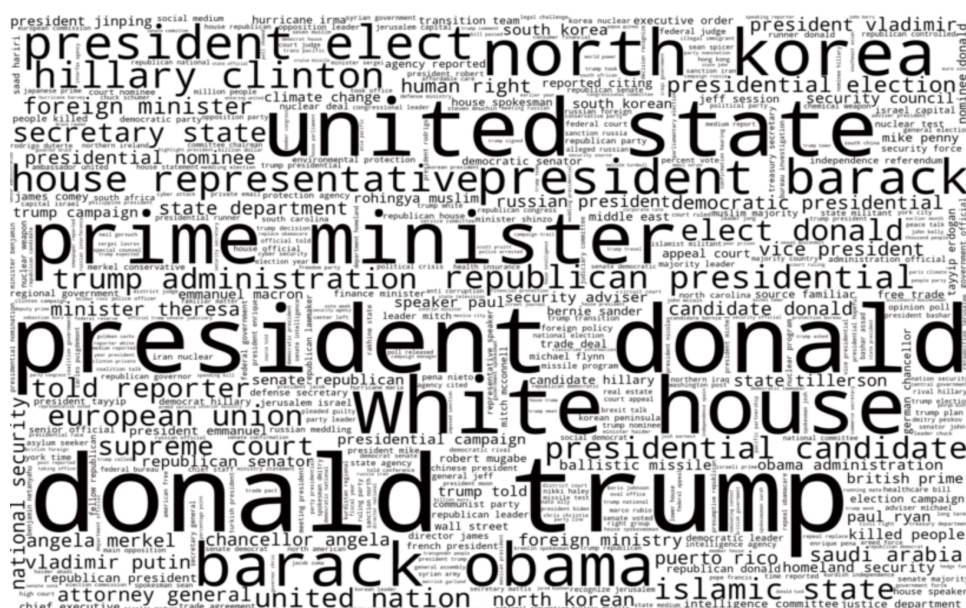


Figure 6.14: Word Cloud ISOT prototype output for class 0(True) using Influential instances

6.1.4 MMD-Critic Sets

The algorithm automatically selects MMD-Critic sets; we find that the size of the sets is larger for fake news compared to true news, shown in size split table 6.7. This indicates there is more variation in fake news compared to true news. We further select the top 10 data instances with a high witness score. Finally, we compare the critics to the harmful instances selected in influential instances and find that the chosen set is 57% match. It is observed in figure 6.19 to 6.24 that true phrases are more neutral and have a less aggressive tone to fake phrases as we can note that there are words such as farm attack, deliberately negligence, people dead in the fake news phrases.

Data Set	Size of True Set	Size of Fake Set
ISOT	54	113
COVID	49	92
SA	42	87

Table 6.7: Table showing the size of each prototype set using MMD-Critic technique for our dataset

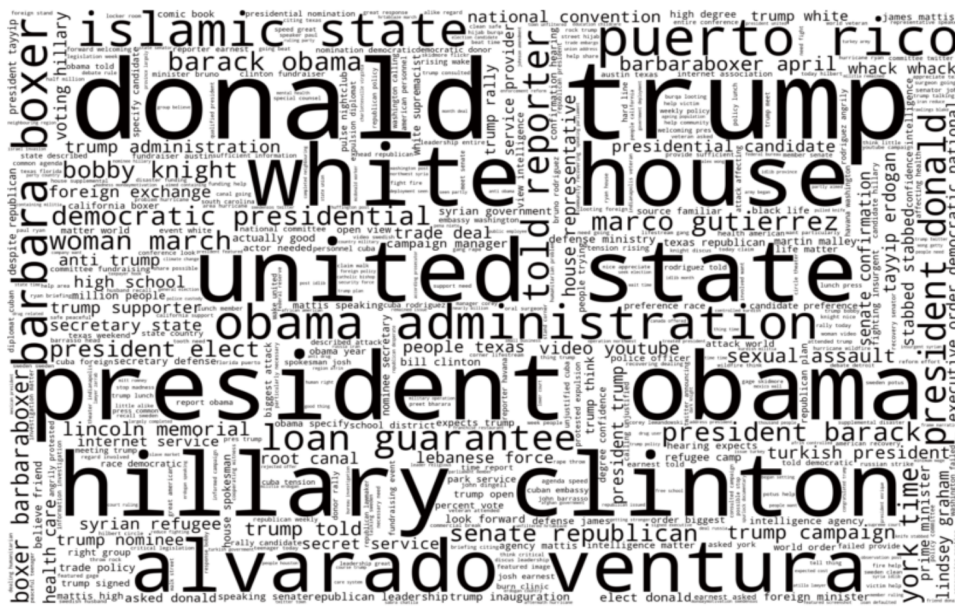


Figure 6.19: Word Cloud ISOT prototype output for class 1(Fake) using MMD-Critic technique

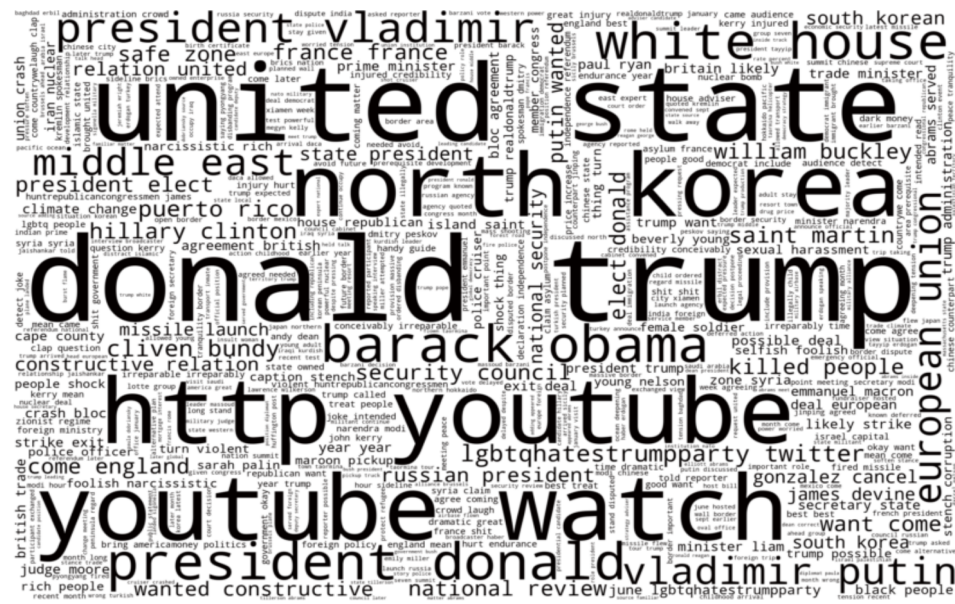


Figure 6.20: Word Cloud ISOT prototype output for class 0(True) using MMD-Critic technique



Figure 6.21: Word Cloud Covid prototype output for class 1(Fake) using MMD-Critic technique



Figure 6.22: Word Cloud Covid prototype output for class 0(True) using MMD-Critic technique

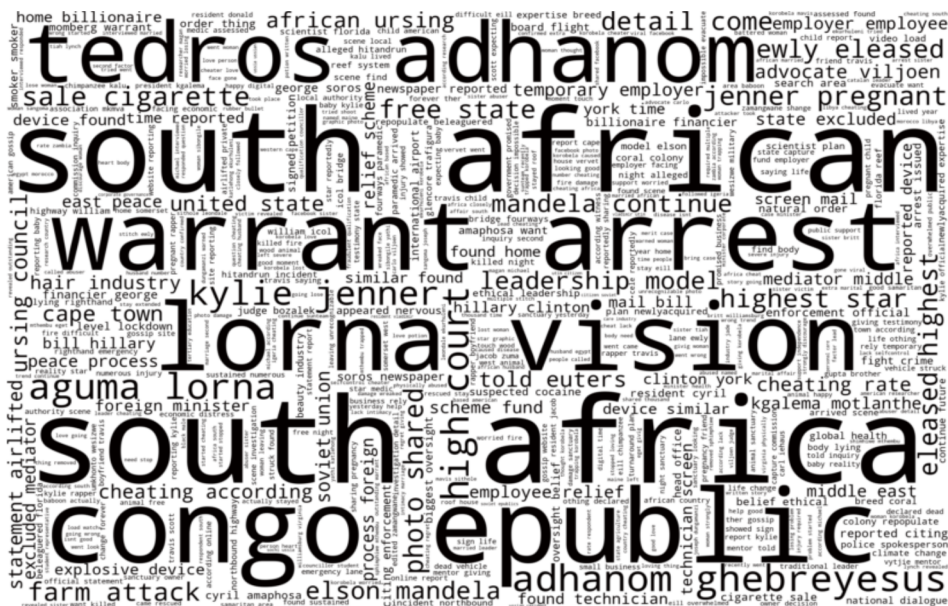


Figure 6.23: Word Cloud SA prototype output for class 1(Fake) using MMD-Critic technique



Figure 6.24: Word Cloud SA prototype output for class 0(True) using MMD-Critic technique

6.2 Quality of prototypes

We need to determine the quality of the prototypes to ensure that the information we extracted is truthful and portable to any use case related to misinformation classification. As described in [7, 8] to determine the quality of prototypes, we train a model with the prototype sets and determine the accuracy percentage when predicting unseen data. The prototype set with the highest accuracy has the highest quality. We train our prototype sets using an LSTM and observe the prediction accuracy in table 6.8.

Across the four example-based methods, we observe that the prototype sets with the highest accuracy are Influential and MMD-Critic; although the two methods have an accuracy of less than 70% we believe that if we manually select the prototype sets to be larger, the accuracy value would increase. Therefore, in the next section, we further process both the Influential and MMD-critic to determine key characteristics we can extract.

Data Set	Clustering	Set Cover	Influential	MMD-Critic
ISOT	36%	41%	67%	71%
COVID	33%	39%	56%	59%
SA	31%	34%	47%	53%

Table 6.8: Table showing the quality of each prototype set by observing prediction quality

6.3 Further Processing of Selected Sets.

In this section, we further analyse the two prototype sets with the highest quality measured by the prediction accuracy. First, we look at the top phrases of each set; we use them to perform sentiment, opinion detection, and word analysis. We further use a model agnostic explainer on the sets and, lastly, use a probing technique to determine linguistic properties.

6.3.1 Analysis of Set Phrases

As seen in Appendix D, the prototypical examples from text data are just a list of sentences where one can not distinguish the key characteristics between the classes. We take advantage of the SBERT functionality, keyBERT, which uses the embedding of the model to generate keywords and phrases from a document by calculating the cosine similarity of each generated key phrase to the document. We select the top 10 key phrases with the highest similarity score.

It is observed in figure 6.25 to 6.30 that true phrases are more neutral and have a less aggressive tone to fake phrases as we can note that there are words such as farm attack, deliberately negligence, people dead in the fake news phrases.

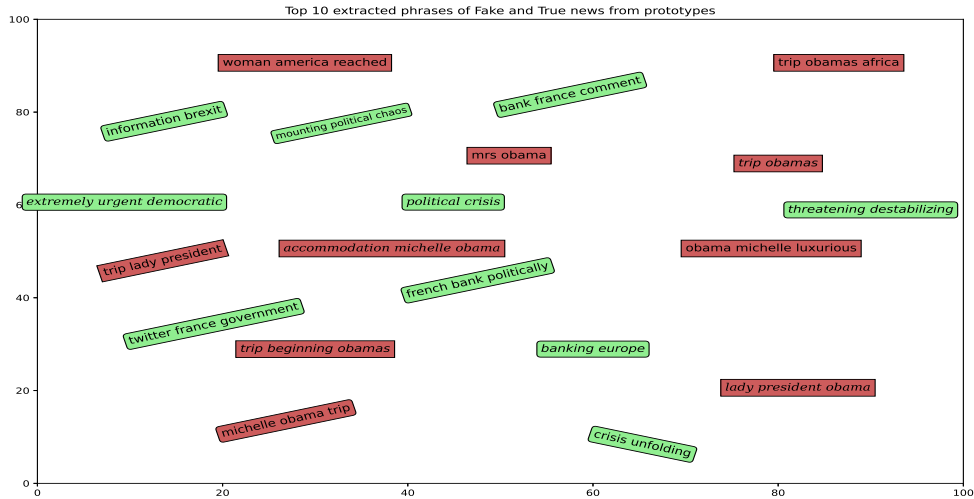


Figure 6.25: Figure showing top 10 phrases extracted from ISOT MMD prototypical set examples. Green phrases are true news phrases and red fake news phrases

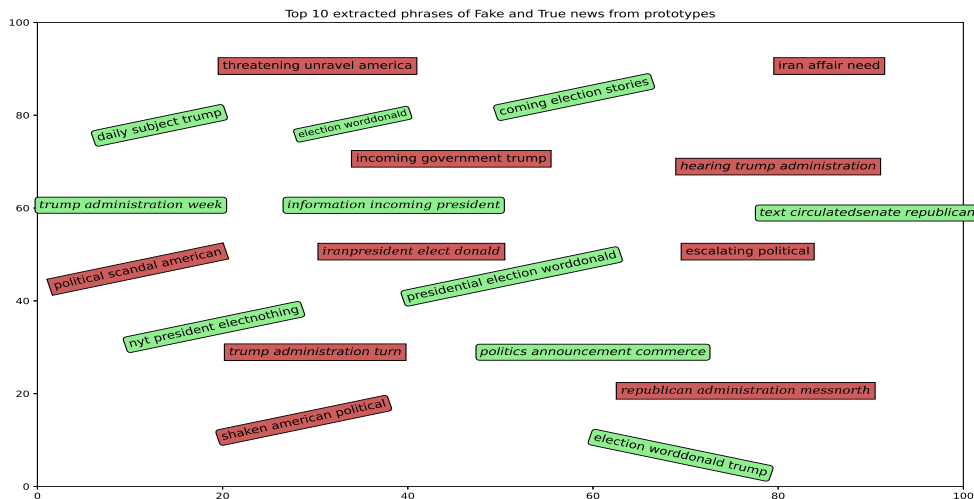


Figure 6.26: Figure showing top 10 phrases extracted from ISOT Influential prototypical set examples. Green phrases are true news phrases and red fake news phrases

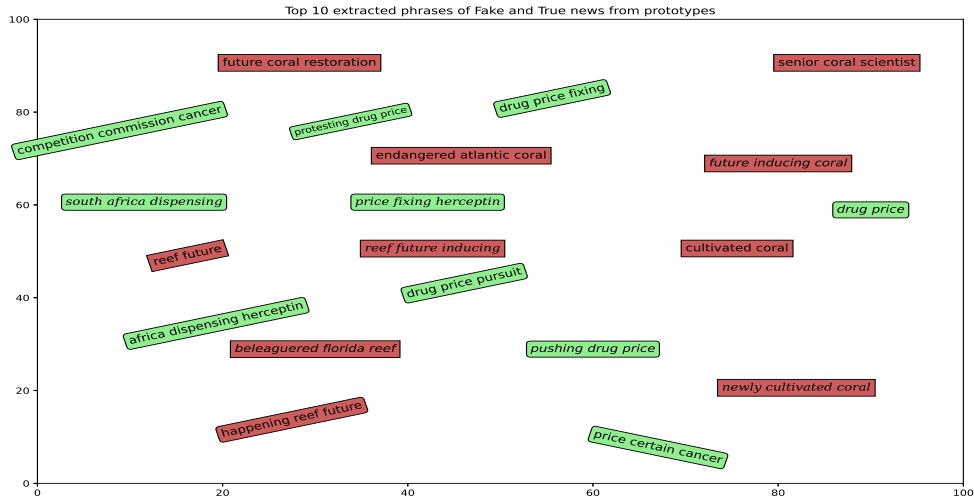


Figure 6.27: Figure showing top 10 phrases extracted from SA MMD prototypical set examples. Green phrases are true news phrases and red fake news phrases

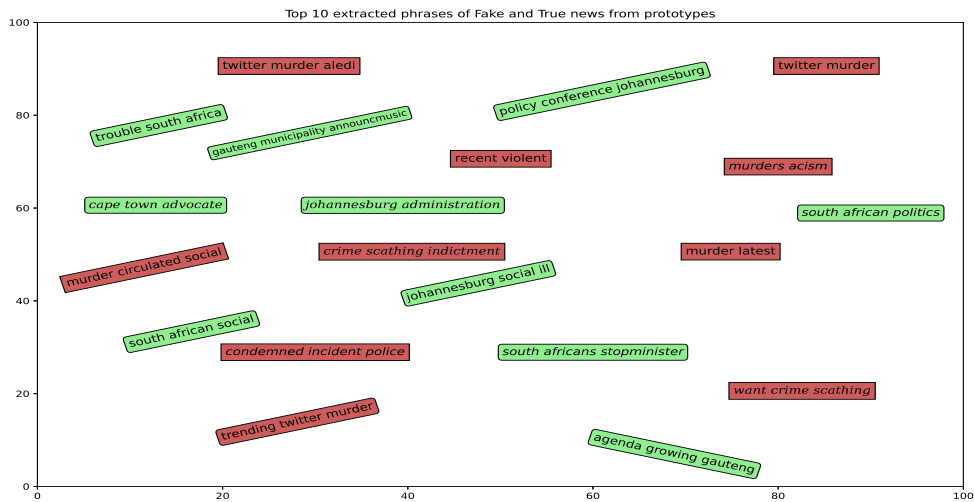


Figure 6.28: Figure showing top 10 phrases extracted from SA Influential prototypical set examples. Green phrases are true news phrases and red fake news phrases

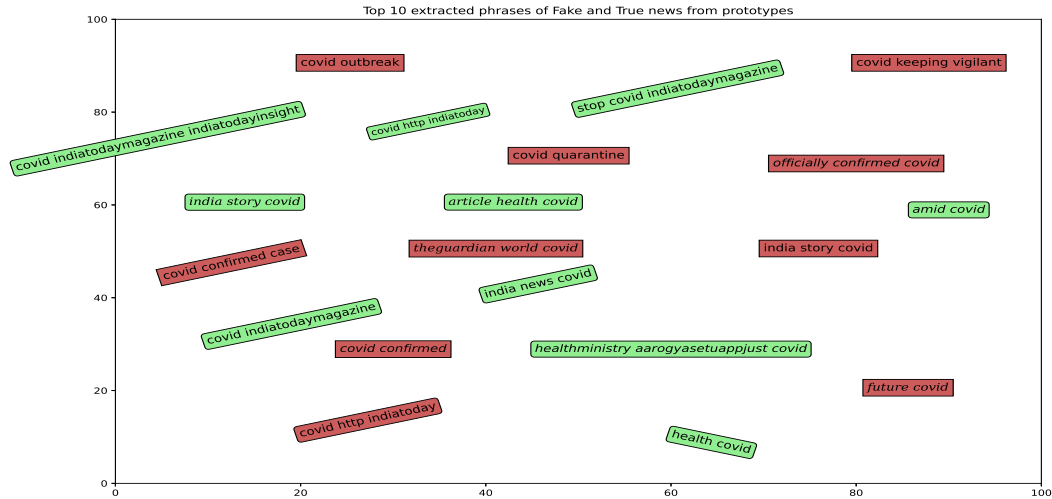


Figure 6.29: Figure showing top 10 phrases extracted from Covid MMD prototypical set examples. Green phrases are true news phrases and red fake news phrases

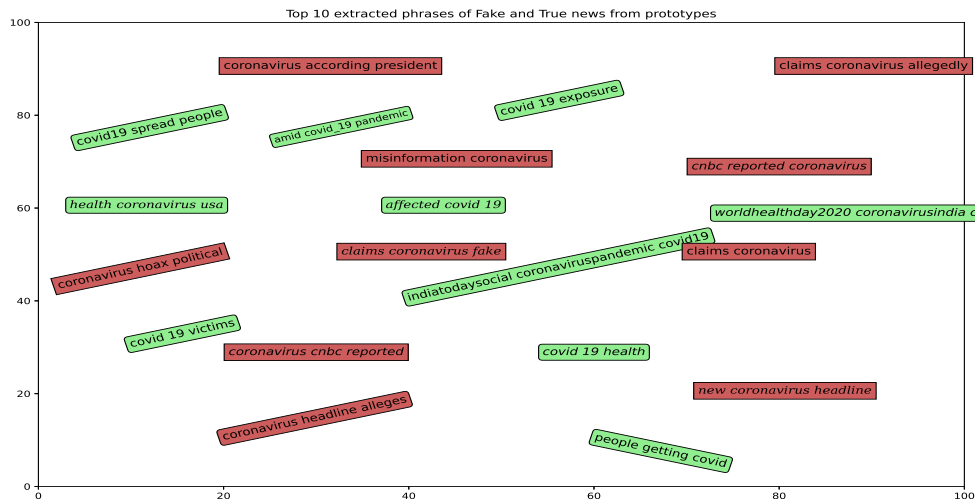


Figure 6.30: Figure showing top 10 phrases extracted from Covid Influential prototypical set examples. Green phrases are true news phrases and red fake news phrases

6.3.2 Sentiment and opinion detection

We examined both sentiment and subjectivity from the prototypical examples. We used the textblob measure of subjectivity and polarity to determine whether the text classes are more positive, negative, neutral, objective, or subjective. As can be observed in figure 6.31-6.35, we find that overall, fake news displayed in the orange bar is more subjective compared to true news displayed in the blue bar; however, when it comes to South African text, we find that true news is more subjective and fake is more objective. We further observe in figure 6.33-6.35 that true news is more positive than fake news. Another unique observation of the South African dataset is that it does not have neutral text when analysed with the MMD set.

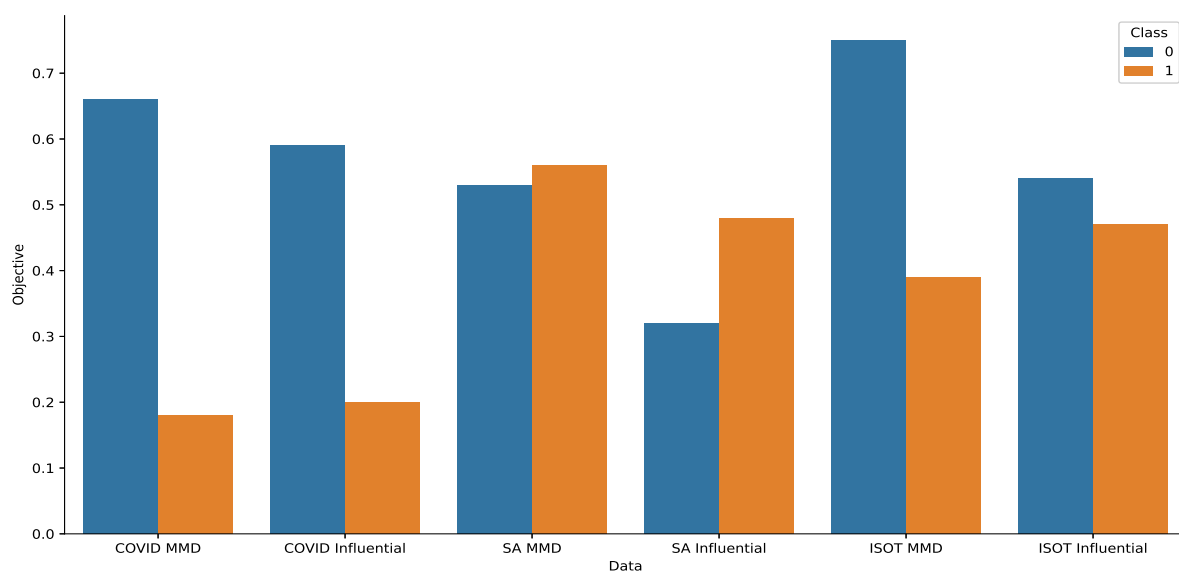


Figure 6.31: Figure showing opinion mining results based on the objective measure of textblob from all three datasets. It shows the percentage count of the text values that were subjective compared to objective

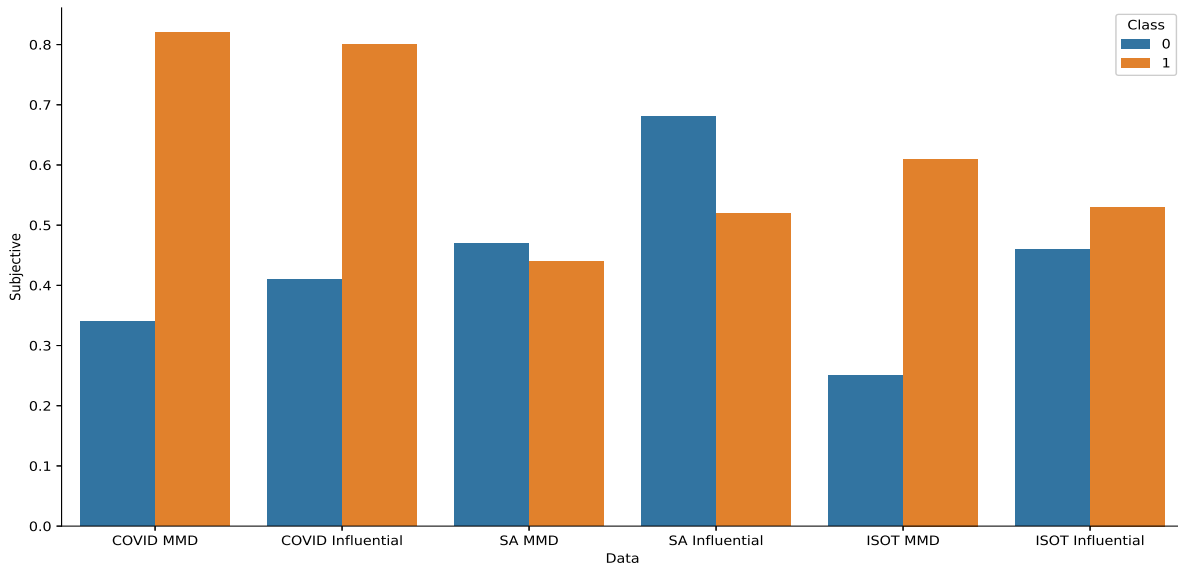


Figure 6.32: Figure showing opinion mining results based on subjectivity measure of textblob from all three datasets. It shows the percentage count of the text values that were subjective compared to objective

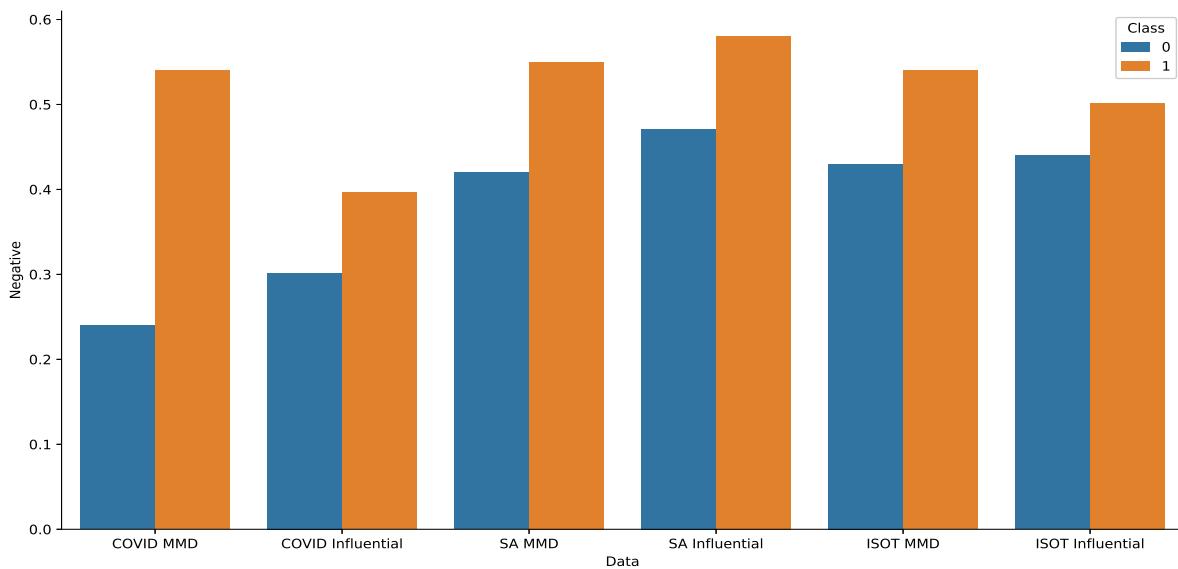


Figure 6.33: Figure showing sentiment results based on polarity measure of textblob from all three datasets. It shows the percentage count of the text values that were negative

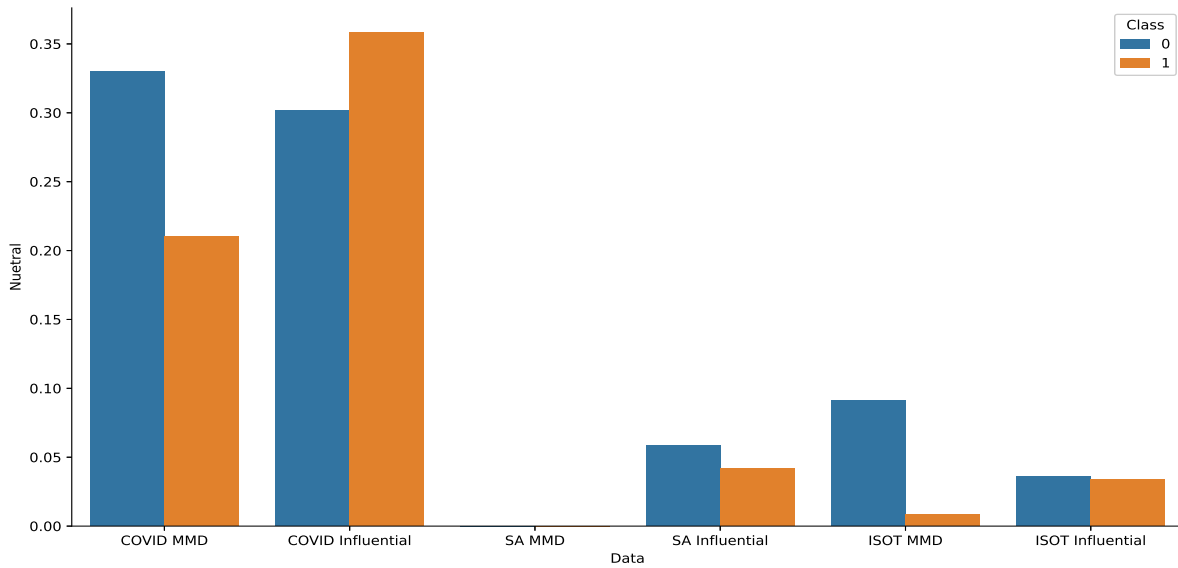


Figure 6.34: Figure showing sentiment results based on polarity measure of textblob from all three datasets. It shows the percentage count of the text values that were neutral

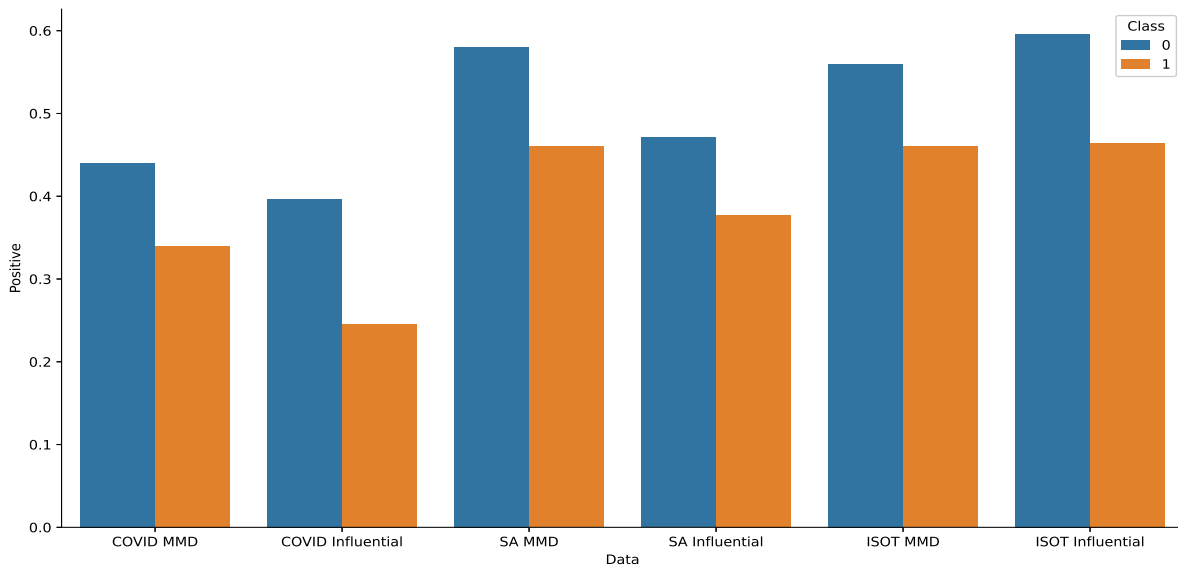


Figure 6.35: Figure showing sentiment results based on polarity measure of textblob from all three datasets. It shows the percentage count of the text values that were positive

6.3.3 Word analysis

Word length

Next, we observed if there were any significant differences in the average words between fake and true news data. We find that, on average, between the three datasets, fake news is about 18% less compared to true news, and we further observe that fake news has more words in capital, showing a more aggressive tone. Finally, we show the average words and capital word counts in table 6.9.

Data	Average Words	Average Upper
Covid True	35	3
Covid Fake	14	7
SA True	336	2
SA Fake	274	3.5
ISOT True	463	5
ISOT Fake	328	7.5

Table 6.9: Table showing the average words and upper case words in the prototypical example dataset

Word comparison

We use the Scattertext library to plot how fake and true news words differ in words and the intersection points between the groups. For example, in figure 6.36-6.38, we note that true words for covid data that are not used in fake news are words such as http, story, covidoutbreak, trending, and social; which are very neutral words that are not persuasive or negative. While fake news has words such as viral, country names, image, and kill, which are commonly associated with controversial words or topics.

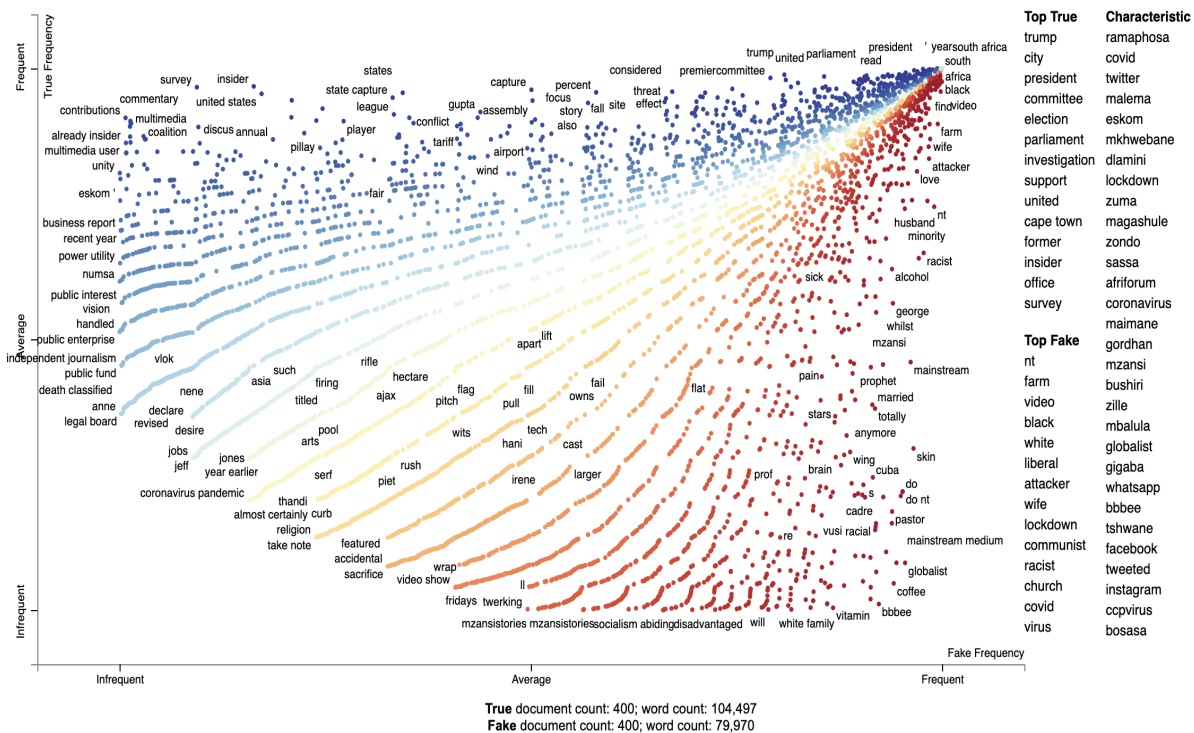


Figure 6.36: Figure showing word differences and similarities between the two class groups in SA prototypical examples

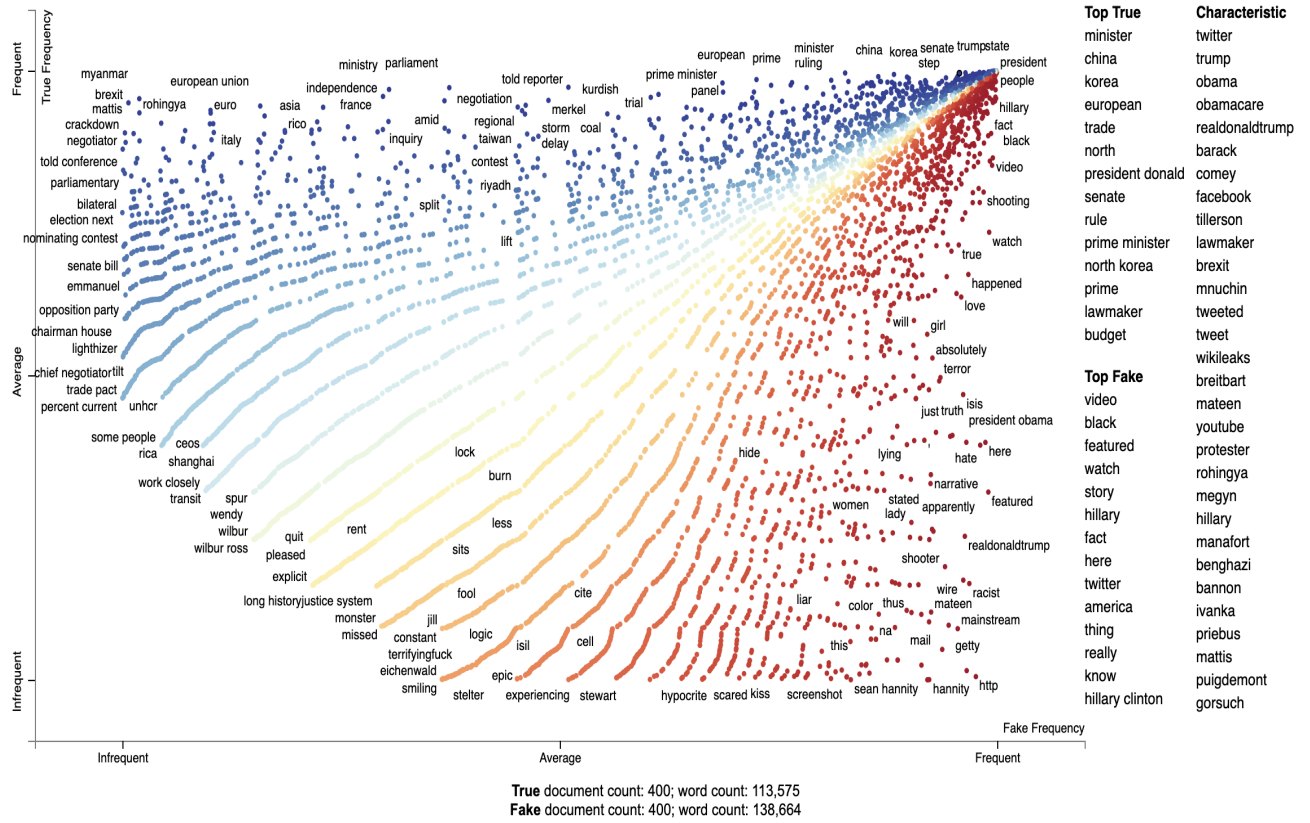


Figure 6.37: Figure showing word differences and similarities between the two class groups in ISOT prototypical examples

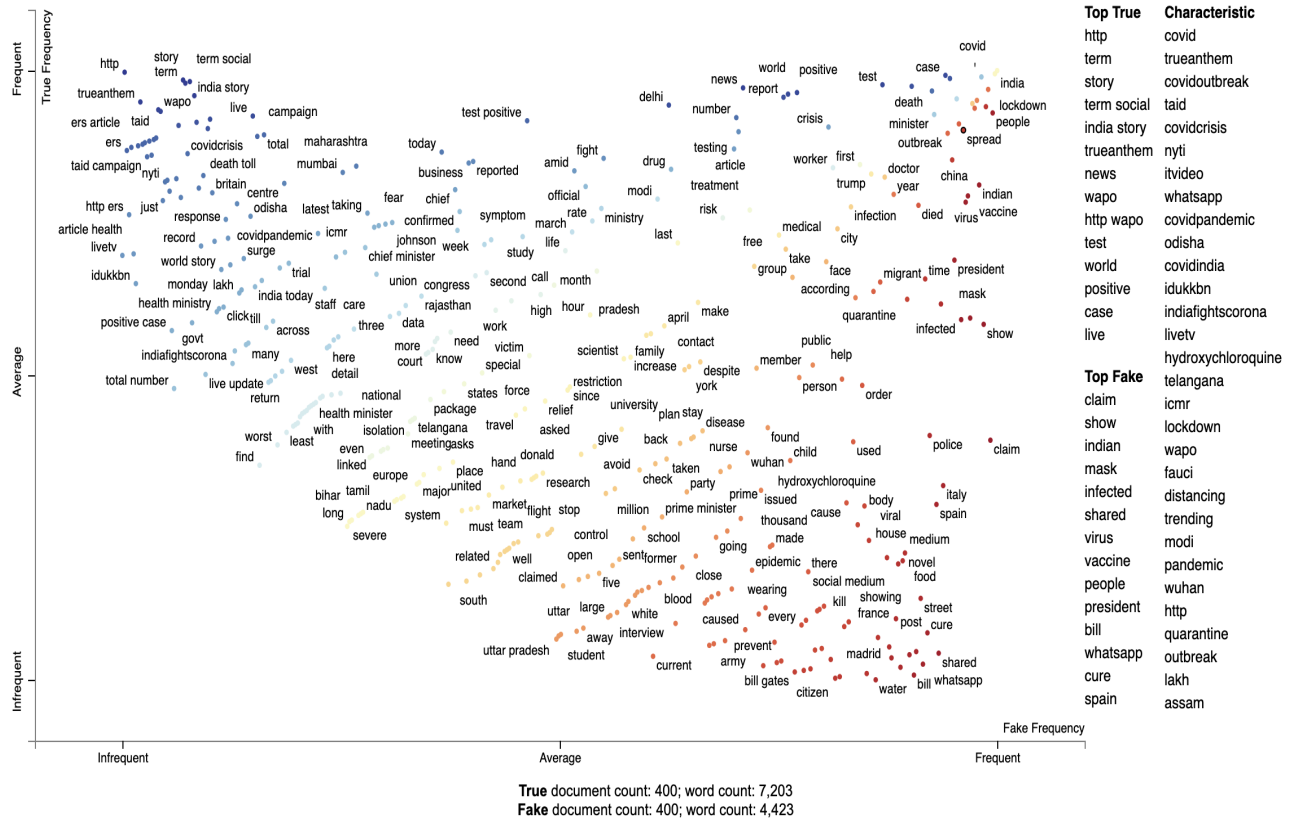


Figure 6.38: Figure showing word differences and similarities between the two class groups in COVID prototypical examples

6.3.4 Linguistic Properties

Given that we are dealing with language data, we also need to observe characteristic differences in linguistic properties. We, thus, observe the part of speech from each class in the prototypes shown in figure 6.39 to 6.41. The most significant differences are noted regarding noun usage in the ISOT prototype examples, where true news uses more nouns than fake news. Overall across the datasets, true news text use past tense verbs(VBD) and prepositions(IN) more compared to fake news, while fake news has more adjective (JJ), verb base form(VB), and verb 3rd person(VBZ) compared to true news. It is further noted that ISOT fake news text has more pronouns than true text compared to the rest of the datasets. South African text data has more foreign words(FW) than the rest of the dataset's text; this may be because South African text is usually used with a combination of colloquial words.

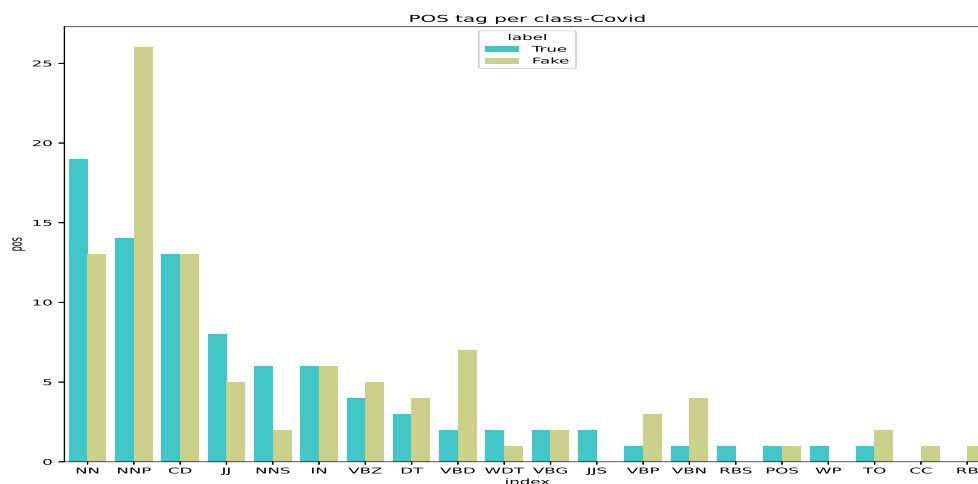


Figure 6.39: Figure showing part of speech tag for Covid prototypes

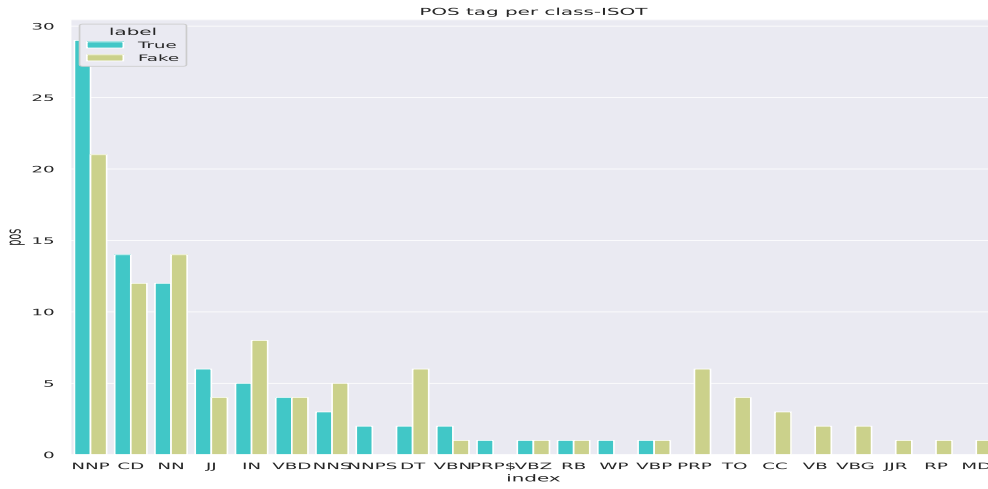


Figure 6.40: Figure showing part of speech tag for ISOT prototypes

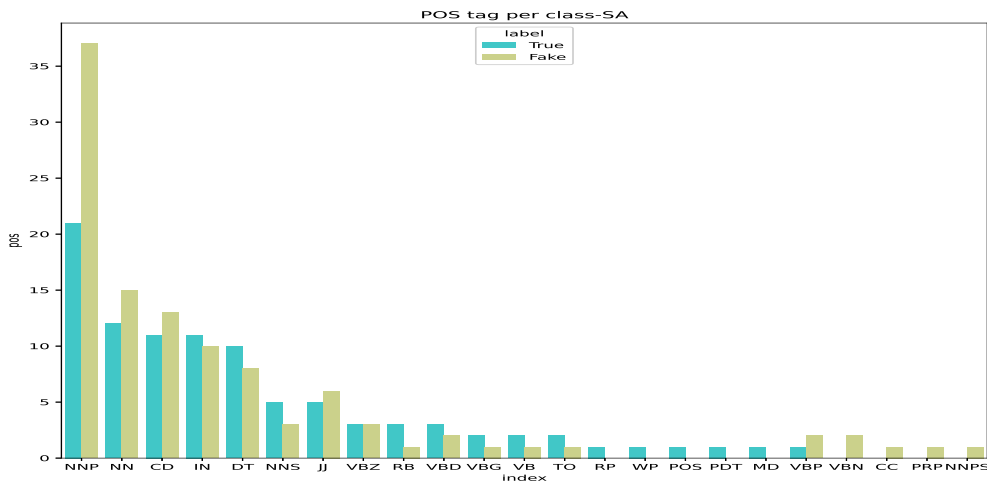


Figure 6.41: Figure showing part of speech tag for SA prototypes

6.3.5 Prototypical examples used with SP-Lime

We use the LSTM trained to perform model agnostic global explainer SP-Lime to determine any additional information that can be extracted using both example-based explainer and model agnostic explainer. The results of the explainer can be seen in figure 6.42-6.43. We find that the most relevant words indicated by the explainer are similar to what we observed in the scattertext output, where text classified as fake news has words that evoke emotion and are centered around a controversial topic whilst true news is more neutral and objective.

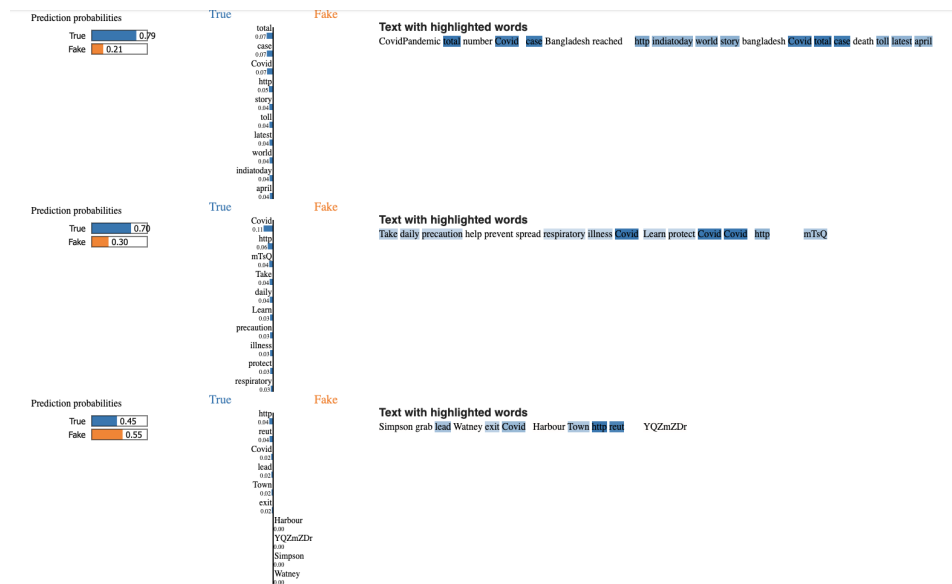


Figure 6.42: Output of using SP-LIME with Prototypical examples to explain and indicate prevalent words for true news class



Figure 6.43: Output of using SP-LIME with Prototypical examples to explain and indicate prevalent words for Fake news class

6.3.6 Critics and harmful instances

The advantage of both MMD-Critic and Influential instances is that they indicate which instances are harmful or are outliers to a class. MMD automatically selects the critics for each class, and 10% of the selected critic set belonged to true news, whilst 90% of the instances belonged to fake news. This indicates that models are more likely to misclassify fake news instances as true news.

We analyse phrases and the output of an attribution map for the top harmful influential instance and MMD-critic instance for fake news class (1). In figure 6.44, we can see that the phrases selected for the harmful instances are very similar to that of true news, making it difficult to distinguish between the classes.

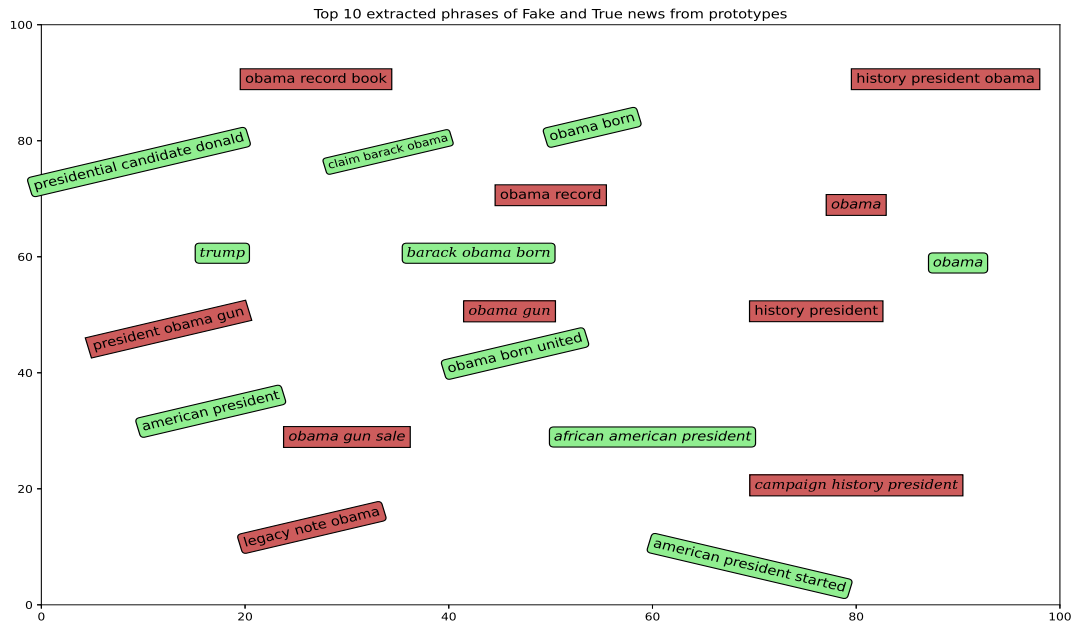


Figure 6.44: Output showing comparison of phrases selected for harmful instance of fake news data to that of a true news instance

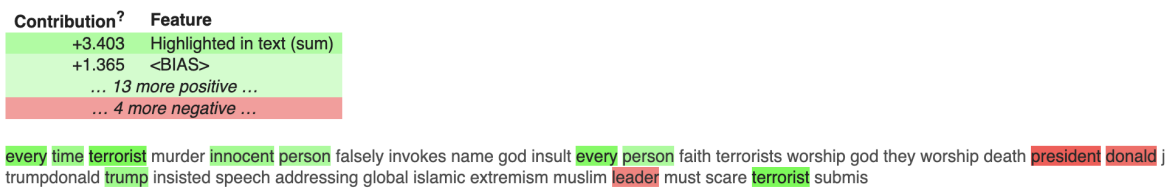


Figure 6.45: Output of using attribution map to determine key features for harmful instance indicated by influential and MMD-critic

6.4 Summary

We used four approaches to select prototypes on three of our datasets; once we had our chosen sets based on the algorithms, we measured for each set the quality of the prototypes. Influential instances and MMD-Critic produced prototype sets with the highest quality; for these two sets, we explore if there are any character differences between the two classes. This is applied across the datasets to determine any differences between the datasets. Below we list some of the significant differences which we observe:

- In section [6.3.3](#), we performed word analysis, and the result across the datasets led to the conclusion that Fake news is shorter in length and contains more capital words.
- In section [6.3.2](#), we measured the subjectivity of the prototypes and observed that Fake news is more subjective and often opinionated.
- Using POS tagging in [6.3.4](#), we were able to conclude that Fake news uses more adjectives, verb in 3rd person, and pronouns in linguistic properties.
- By observing the output of the word cloud images, word comparison's from scattertext, and phrase extraction, we note that Fake news phrases are more aggressive and are always around controversial topics.
- In figure [6.40](#) POS tag results for South African data and [6.36](#) the scattertext output; We determine that South African fake data has more foreign words as more colloquial words are used.
- Further, from the POS tag in [6.3.4](#), we can also say South African text is either positive or negative, with hardly any neutral text.
- Lastly, South African fake news is less subjective and more objective compared to the rest of the datasets as per the output of subjectivity analysis performed in [6.3.2](#).

We also observed from harmful instances that fake news text could be very difficult to detect by a layperson. However, by using linguistic properties, a linguistic expert can detect the slight differences that exist. Consumers of news can use the above characteristics to determine the probability of fakeness from text data.

Chapter 7

Conclusions

This chapter summarizes the research work presented in this study. We revisit our research questions and discuss the main findings. We also discuss proposed future work.

7.1 Summary of Conclusions

In this research, we discuss the problem of interpretability in natural language processing, specifically with respect to misinformation classification tasks. We want to determine what behavioural factors can be used to distinguish a misinformation class and which interpretability model is best suited for interpreting misinformation data.

In chapter 4 we used the commonly used interpretation methods for natural language processing, methods which output attribution maps by highlighting the most important words and phrases in a prediction to explain the prediction outcome. We found that the attribution methods were faithful in their interpretation. However, given that they explained a single instance, it was challenging to determine common characteristics of a class and clearly determine the characteristics, if any, of misinformation data.

In chapter 5 we used example-based techniques for model interpretation to determine if any key characteristics could be used to explain the prediction between the two classes. We found that with natural language; prototype selection methods, unlike images, one cannot easily determine the differences and explanations of the predictions of the two classes. We, therefore, further probed the prototypes of each class extracted using vari-

ous techniques and tools to determine the character and linguistic differences. We found some significant differences in word length, tone, subjectivity, part of speech, and sentiment as described in chapter 6. We further found that more fake news instances are critics compared to true news, showing that fake news curated tries to mimic the properties and authenticity of the news piece.

From the results obtained in chapter 5 and 4 we were able to compare prototype/example-based methodology with the most commonly used algorithms. We note that with example-based methods, one can extract a sample of the data which best represents a particular class. Then from the sample, easily extract characteristics that make up the class. The key difference's between example-based techniques and common techniques is that with example-based techniques, one could make characteristic differences before having to train a model. In contrast, most techniques require a model to be trained to extract patterns. Further, examples extracted may help build mental models using the prototypes and critics to achieve better human interpretability. However, when it comes to natural language processing, using both techniques to achieve interpretability in misinformation is best suited to truly understand the data and extract key information between the classes. Lastly, given that most of the characteristic differences between the two classes observed require a language expert, the method may be extended to include human-in-the-loop to determine the prototypes' robustness, faithfulness, and quality.

7.2 Future Work

There are more insights that we can achieve from the prototype methodology. We list below some potential avenues for future research, specifically in disinformation:

- The quality of prototypes and critic sets selected can change based on the kernel used in MMD-critic and how we represent the data. Using MMD-critic as the algorithm, we can explore how the prototype and critic sets differ if a string kernel or sequence embeddings is used to capture the sequence of the documents.
- Determining the quality of the prototypes is crucial and needs further exploring. Current literature proposes, together with training, a model to have humans in

the loops to assess the prototypes to determine their quality and characteristics. In the future, we can extend our research by adding a human-in-the-loop measure, more specifically, linguistic experts, to examine the quality of prototypes.

- Lastly, Given the distinct key patterns observed, we can explore building an intrinsic interpretable pipeline. Which will have two phases, with the first phase extracting key feature attributes from the text and then using the feature attributes as feature inputs in a logistic regression model or any white-box model. Such that we can distinguish if, over time, we can have key features that can be used in white-box models to classifier misinformation.

Bibliography

- [1] A Convolutional Approach for Misinformation Identification. *IJCAI International Joint Conference on Artificial Intelligence*, 0:3901–3907, 2017.
- [2] SmoothGrad: removing noise by adding noise. 2017.
- [3] Interpretable Machine Learning. A Guide for Making Black Box Models Explainable. *Book*, page 247, 2019.
- [4] Raed Alharbi, Minh N. Vu, and My T. Thai. Evaluating Fake News Detection Models from Explainable Machine Learning Perspectives. *IEEE International Conference on Communications*, 2021.
- [5] Jackie Ayoub, X. Jessie Yang, and Feng Zhou. Combat COVID-19 infodemic using explainable natural language processing models. *Information Processing and Management*, 58(4), 2021.
- [6] Jasmijn Bastings and Katja Filippova. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? pages 149–155, 2020.
- [7] Kim Been, Khanna Rajiv, and Oluwasanmi Koyejo. Examples are not Enough, Learn to Criticize! Criticism for Interpretability Been. *NIPS*, 2016.
- [8] Jacob Bien and Robert Tibshirani. PROTOTYPE SELECTION FOR INTERPRETABLE CLASSIFICATION. *Institute of Mathematical Statistics*, 5(4):2403–2424, 2011.

- [9] Jacob Bien and Robert Tibshirani. Prototype selection for interpretable classification. *Annals of Applied Statistics - ANN APPL STAT*, 5, 02 2012.
- [10] Peng Ce and Bao Tie. An analysis method for interpretability of CNN text classification model. *Future Internet*, 12(12):1–14, 2020.
- [11] Akif Cinar. Overview of existing approaches for the interpretation of machine learning models. pages 1–11.
- [12] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What Does BERT Look at? An Analysis of BERT’s Attention. pages 276–286, 2019.
- [13] Marina Danilevsky, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. A Survey of the State of Explainable AI for Natural Language Processing. (Section 5), 2020.
- [14] Arun Das and Paul Rad. Opportunities and Challenges in Explainable Artificial Intelligence (XAI): A Survey. pages 1–24, 2020.
- [15] Hwang Doosung and Kimt Daewon. Nearest neighbor based prototype classification preserving class regions. *Journal of Information Processing Systems*, 5:1345–1357, 02 2017.
- [16] Finale Doshi-Velez and Been Kim. Towards A Rigorous Science of Interpretable Machine Learning. feb 2017.
- [17] Mengnan Du, Ninghao Liu, and Xia Hu. Techniques for interpretable machine learning. *Communications of the ACM*, 63(1):68–77, 2020.
- [18] Keyur Faldu and Amit Sheth. Discovering the Encoded Linguistic Knowledge in NLP models. *Towards Data Science*, pages 7–8, 2020.
- [19] Guillaume Gadek and Paul Guélorget. An interpretable model to measure fakeness and emotion in news. *Procedia Computer Science*, 176:78–87, 2020.
- [20] Moyank Giri, Tarun Aditya, Prasad B Honnavalli, and Sivaraman Eswaran. Automated and Interpretable Fake News Detection with Explainable Artificial Intelligence. *SSRN Electronic Journal*, 2022.

- [21] Karthik S. Gurumoorthy, Amit Dhurandhar, Guillermo Cecchi, and Charu Aggarwal. Efficient data representation by selecting prototypes with importance weights. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 260–269, 2019.
- [22] John Hewitt and Percy Liang. Designing and interpreting probes with control tasks, 2019.
- [23] Linwei Hu, Jie Chen, Vijayan N Nair, and Agus Sudjianto. Surrogate Locally-Interpretable Models with Supervised Machine Learning Algorithms. *arXiv:2007.14528 [cs, stat]*, pages 1–24, 2020.
- [24] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. On the Validity of Self-Attention as Explanation in Transformer Models. *arXiv*, 2017.
- [25] Been Kim, Cynthia Rudin, and Julie Shah. The Bayesian case model: A generative approach for case-based reasoning and prototype classification. *Advances in Neural Information Processing Systems*, 3(January):1952–1960, 2014.
- [26] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions, 2017.
- [27] Hima Lakkaraju, Julius Adebayo, and Sameer Singh. Explaining Machine Learning Predictions: State-of-the-art, Challenges, Opportunities Slides and Video: explainml-tutorial.github.io Motivation. 2020.
- [28] Qing Liao, Heyan Chai, Hao Han, Xiang Zhang, Xuan Wang, Wen Xia, and Ye Ding. An integrated multi-task model for fake news detection. *IEEE Transactions on Knowledge and Data Engineering*, 34(11):5154–5165, 2022.
- [29] Andreas Madsen, Siva Reddy, and Sarath Chandar. Post-hoc Interpretability for Neural NLP: A Survey. 2021.
- [30] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

- [31] Ken Mishima and Hayato Yamana. A Survey on Explainable Fake News Detection. *IEICE Transactions on Information and Systems*, E105D(7):1249–1257, 2022.
- [32] Sina Mohseni, Fan Yang, Shiva Pentyala, Mengnan Du, Yi Liu, Nic Lupfer, Xia Hu, Shuiwang Ji, and Eric Ragan. Machine Learning Explanations to Prevent Overtrust in Fake News Detection. *Proceedings of the International AAAI Conference on Web and Social Media*, 15:421–431, 2021.
- [33] Christoph Molnar, Giuseppe Casalicchio, and Bernd Bischl. Interpretable Machine Learning – A Brief History, State-of-the-Art and Challenges. *Communications in Computer and Information Science*, 1323(01):417–431, 2020.
- [34] Grégoire Montavon, Wojciech Samek, and Klaus Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing: A Review Journal*, 73:1–15, 2018.
- [35] Ian E. Nielsen, Ghulam Rasool, Dimah Dera, Nidhal Bouaynaya, and Ravi P. Ramachandran. Robust Explainability: A Tutorial on Gradient-Based Attribution Methods for Deep Neural Networks. pages 1–21, 2021.
- [36] Yu Qiao, Daniel Wiechmann, and Elma Kerz. A Language-Based Approach to Fake News Detection Through Interpretable Features and BRNN. *Proceedings of the 3rd International Workshop on Rumours and Deception in Social Media (RDSM)*, pages 14–31, 2020.
- [37] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Model-Agnostic Interpretability of Machine Learning. (Whi), 2016.
- [38] Alexis Ross, Ana Marasović, and Matthew Peters. Explaining NLP Models via Minimal Contrastive Editing (MiCE). pages 3840–3852, 2021.
- [39] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences, 2019.

- [40] Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. Defend: Explainable fake news detection. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 395–405, 2019.
- [41] Kai Shu, Deepak Mahudeswaran, Suhang Wang, and Huan Liu. Hierarchical propagation networks for fake news detection: Investigation and exploitation. *Proceedings of the 14th International AAAI Conference on Web and Social Media, ICWSM 2020*, pages 626–637, 2020.
- [42] Xiaofei Sun, Diyi Yang, Xiaoya Li, Tianwei Zhang, Yuxian Meng, Han Qiu, Guoyin Wang, Eduard Hovy, and Jiwei Li. Interpreting Deep Learning Models in Natural Language Processing: A Review. pages 1–36, 2021.
- [43] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Integrated Gradient Axiomatic attribution for deep networks. *34th International Conference on Machine Learning, ICML 2017*, 7:5109–5118, 2017.
- [44] Martin Tutek and Jan Šnajder. Iterative Recursive Attention Model for Interpretable Sequence Classification. pages 249–257, 2019.
- [45] Tongshuang Wu, Marco Tulio Ribeiro, Jeffrey Heer, and Daniel Weld. Polyjuice: Generating Counterfactuals for Explaining, Evaluating, and Improving Models. pages 6707–6723, 2021.
- [46] Pengtao Xie, Yuntian Deng, and Eric Xing. Latent Variable Modeling with Diversity-Inducing Mutual Angular Regularization. 2015.
- [47] Fan Yang, Mengnan Du, Eric D. Ragan, Shiva K. Pentyala, Hao Yuan, Shuiwang Ji, Sina Mohseni, Rhema Linder, and Xia Hu. XFake: Explainable fake news detector with visualizations. *The Web Conference 2019 - Proceedings of the World Wide Web Conference, WWW 2019*, 2:3600–3604, 2019.
- [48] Die Zhang, Huilin Zhou, Hao Zhang, Xiaoyi Bao, Da Huo, Ruizhao Chen, Xu Cheng, Mengyue Wu, and Quanshi Zhang. Building Interpretable Interaction Trees for Deep NLP Models. 2020.

- [49] Wei Zhao, Rahul Singh, Joshi Tarun, Sudjianto Agus, and Nair Vijayan N. Self-interpretable Convolutional Neural Networks for Text Classification. *Lecture Notes in Electrical Engineering*, 715:309–316, 2021.

Appendix A

Appendix 1

The first Appendix shows the word analysis images discussed in the exploratory data analysis section in chapter 3. We show visually word cloud and tri-gram output of our three datasets.



Figure A.1: Word Cloud class 1(Fake) word clod image.



Figure A.2: Word Cloud class 0(True) word clod image.

Appendix B

Appendix 2

In this Appendix we show the results of unsupervised topic modelling with the gensim model as described in Chapter 3.

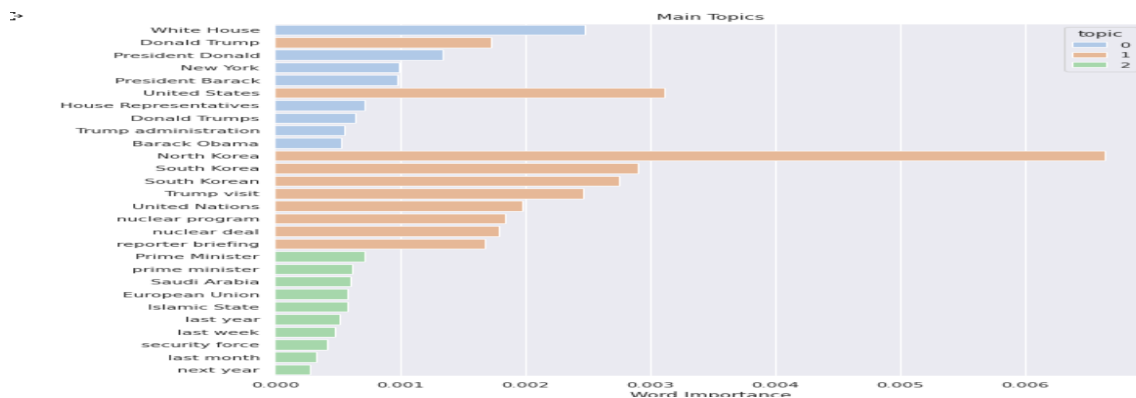


Figure B.1: Bi-gram topic analysis for ISOT Data

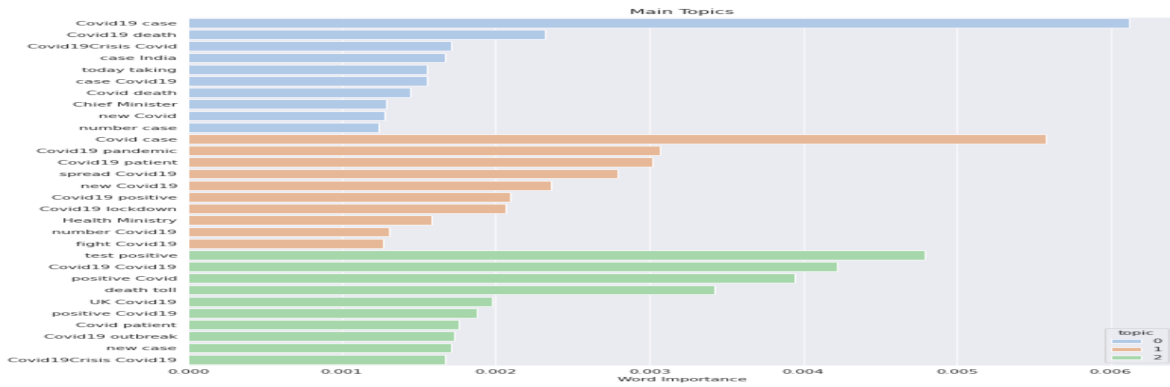


Figure B.2: Bi-gram topic analysis for Covid Data

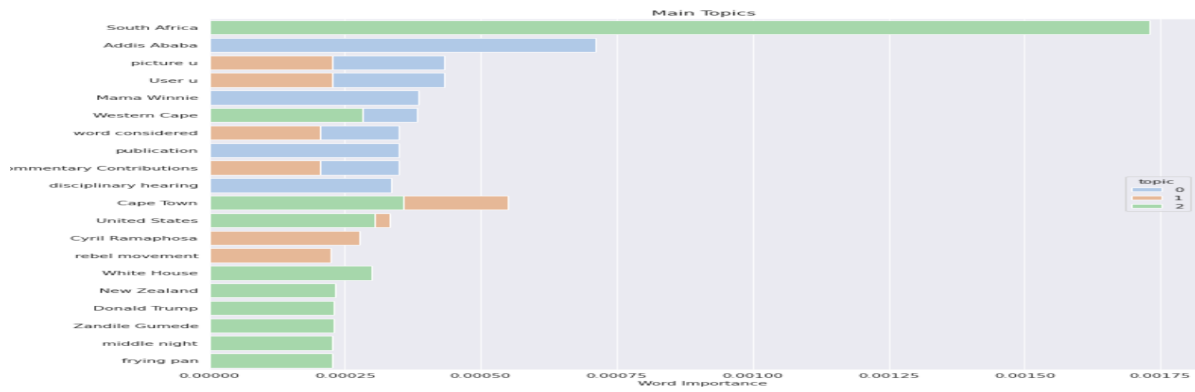


Figure B.3: Bi-gram topic analysis for SA Data

B.1 Topic Modeling

We show the three main topic themes across each dataset.

Appendix C

Appendix 3

In this Appendix we show the results of unsupervised topic modelling with the gensim model as described in Chapter 4.



Figure C.1: SP-LIME results from a representative sample of fake news dataset



Figure C.2: SP-LIME results from a representative sample of True news dataset

Appendix D

Appendix 4

In this Appendix we show the results of using MMD-critic approach to extract prototypes as discussed in chapter 6. We extracted prototypes for each dataset.

Fake Class Prototypes:

shared multiple post alongside claim show body people died Italy infected novel Covid Covid
 Bill Gates Covid vaccine kill people
 Government India announced National Lockdown country
 Garbling water cure Covid doctor Mumbai Covid hospital
 Claim Indian Prime minister Modi crore million Covid positive patient treated free
 President Donald Trump test positive Covid
 show police killing Covid affected people China
 Message claiming city Mumbai military lockdown nest contain spread Covid virus silk medicine available time
 Amid Covid lockdown food distributed Indian State Tamil Nadu
 Italian officer killed prevent police taking quarantine suffering Covid infection
 covid pandemic caused death world population
 show Covid patient state Covid isolation ward Pakistan
 Image word foot sole migrant worker walking street India asia Covid Lockdown
 paramedic claim risk Covid increase wear face mask
 hospital left without Covid care child young people Madrid
 alleged alert issued infectious outbreak India virus called Nipah deadly Covid published Spanish medium
 show muslim woman spitting plastic throwing house spread Covid
 Every Indian citizen Covid relief clicking link given viral Whatsapp message
 German government people stay home
 Ministry Health Spain issue advice Covid outbreak
 news report prof Covid made Chinese laboratory
 fear Covid Muslims China offering Namaz open street
 Covid case Uttar Pradesh district India
 vaccine help cure Covid three hour
 Image year doctor died Covid
 medical facility shared claim Indian Army hospital Rajasthan Covid pandemic
 Whatsapp message suggests holding breath second drinking water every minute fight Covid
 carrying mother back show migrant travelling home India amidst lockdown
 People Italy threw away money lost family member Covid
 viral apparently showing Muslim devotee sneezing circulated claim Misamuddin mosque Delhi deliberately trying infected order spread Covid India
 Those protective mask manufactured France delivered Germany despite lack stock France Covid epidemic
 Japanese Nobel laureate Tasuku Hojjo worked Wuhan four year claimed SARS human made
 Government India released five phase road ease Covid restriction country
 person arrested violating lockdown night bail lockdown state Gujarat India
 Burnt french minister Health took chloroquine list drug sold counter Covid started spread
 show Covid patient lying dead ground India
 Posts social medium claim Democrats voted stimulus package help American family novel Covid crisis
 Police caught Muslims spreading Covid Delhi
 showing body Italian Covid victim street
 According statement appearing coming There pandemic epidemic exist vaccine needed Healthy people need glove face mask protective equipment There need lockdown curfew well called contact tracing virus cannot survive surface isolation demand government Each every organisation fitness
 State Florida announced measure workplace employee paid mandatory leave avoid spread Covid Covid starting March school close week March
 Social medium post shared thousand time claim White House adviser Anthony Fauci stand alone insisting hydroxychloroquine effectiveness Covid unproven Italy France Spain Brazil work
 road traffic control China detect people infected Covid
 neutralise Covid SARS exposing drinking beverage mask effective virus threat life part high risk group
 nation including want Indian Prime Minister Narendra Modi leader task force combat Covid pandemic
 Whatsapp chain message claim China responsible outbreak Covid country economic benefit price stock market Wuhan suddenly free Covid city case
 York hospital treating Covid patient vitamin
 Audio claiming India complete lockdown
 death reported Israel Covid found cure disease solution water baking soda lemon cure Covid
 large group citizen Republic North Macedonia every member party Social Democratic Union Macedonia SDSU quarantined allegedly contact director Skin Clinic Skopje infected Covid
 April jump Covid case United States related election
 Covid vaccine show virus behind current outbreak
 Indian news channel claimed Covid pandemic affect billion laborer India
 blog article stating Covid misdiagnosed death caused problem linked thrombosis first place According article scientist discovered received international prize
 Pakistani Prime Minister Imran Khan Chinese President Jinping Imran Khan wearing mask
 government Spain going declare state emergency

Figure D.1: Extracted fake-Covid class prototypical examples

True Class Prototypes:

Covid world country Covid case death http theguardian world Covid world country covid case death term Autofeed gminews Social source EchoBox
 CovidRise Covid case come year died Covid http indiastoday india story total Covid case india covid
 Covid test trialed hospital around London http Reuters article health Covid Britain test idKXZRN said
 Maharashtra number confirmed Covid case India followed Tamil Nadu Delhi Covid http indiastoday india story corona Covid lockdown social distancing live update term Autofeed gminews Social source EchoBox
 Australia Covid news Covid death Victoria state stages lockdown live update http theguardian world live australia Covid lockdown social distancing live update term Autofeed gminews Social source EchoBox
 India live Covid patient recovered Health Ministry http trib
 many people test positive Covid Maharashtra Thursday CovidnewsKeralaIdia http indiastoday Covid outbreak story maharashtra covid tally jump toll rise
 Kerala health department says Covid cases increase Covid Covid outbreak http indiastoday india story covid covid Karnataka case
 Another Covid death reported Punjab taking toll State nine according senior official Covid covid Punjab http trib
 What know pregnant covid pandemic http wapo
 Assam government today announced conducting Covid test medium person state IndiaFightsCorona CovidCrisis Covid http indiastoday india story assam provide free covid test journalist
 More Americans contracted Covid expect http theguardian news Covid case count term Autofeed gminews Social source EchoBox
 Three death reported national capital With number fatality Covid stand Delhi government CovidOutbreak http indiastoday india story delhi fresh case Covid total death toll recovery
 Perspective Trump miracle covid drug People take advice http wapo
 scientist make million potential Covid vaccine proof http Reuters article health Covid vaccine oxford idKXZRN said campaign truththem Trending Content truththem source idHwKRW0
 least nurse three doctor first hospital test positive Covid Covid positive staffer including nurse traced second hospital Mumbai CovidCrisis http indiastoday india story mumbai hospital declared containment zone several doctor nurse test positive covid
 Covid Palangana Kerala report Covid death http trib
 Government Covid Control Plan Kinds Care Centres http ndtv india news Covid government covid control plan kind care centre NDTVnewsseeps Covid CovidOutbreak eKlWV7
 During ongoing lockdown period amid Covid crisis discipline sacrifice provided strength fight Covid chief minister Patnaik manogaloiwa Odisha Covidlockdown http indiastoday india story odisha first indian state extend covid lockdown till april
 Exclusive nearly health worker died Covid toll rising http theguardian news health worker dying Covid stars data term Autofeed gminews Social source EchoBox
 Covid first number discharged surpasses Covid infection west Bengal Click link latest Covid India update http DWycC
 Analysis President Donald Trump also stepped attack China recent weak Covid outbreak struggling contain spread virus economic crisis triggered Covid DonaldTrump China http trib
 Avraedjios says if quarantine disruption Covid symptoms today covid Delhi http indiastoday india story delhi chief minister arrived kejrival Covid negative
 Just tally Covid positive person Rajasthan increase detection case district today highest number patient found Jaipur Rajasthan Covid
 AstraZeneca billion Covid vaccine deal http reut
 Fucci warns Covid case surge stay home order lifted quickly http theguardian news fauci Covid covid stay home term Autofeed gminews Social source EchoBox
 easing lockdown restriction TamilNadu Delhi Gujarat among state worst affected Covid fraught high risk prepared Click read http IndiastodayMagazine gpo0
 Prime Minister make request Navratni fight Covid Modi address Covid LIVE updates http Covid Covid CovidOutbreak MDWYTFan
 Global Covid prevention trial hydroxychloroquine resume http Reuters article health Covid hydroxychloroquine idKXZRN said campaign truththem Trending Content truththem source
 Covid test cost Centre private http ndtv india news Covid outbreak centre tell private covid test cost
 Beijing city raise Covid emergency response level http reut
 Black people four time likely Covid find http theguardian world black people four time likely covid find term Autofeed gminews Social source EchoBox
 Covid SkyRome Covid Pandemic Breaking First Covid case confirmed Mumbai' bhavani sahiljoshi join detail Watch LIVE LiveTV VIZBWGJ0
 Boris Johnson return work Monday bout covid report http wapo
 Madhya Pradesh government brought back migrantlabourers stranded States Covid lockdown http trib
 Some covid patient taken ventilator taking even week wake http wapo g9e00
 China report Covid case led to death of 70 people hospital city syngene Covid official Covid http indiastoday india story Covid india west bengal report third covid case
 Last week Europe' worst Covid pandemic http Reuters article health Covid Britain death idKXZRN said campaign truththem Trending Content truththem source hJldg8Hj
 plasma therapy cure affected Covid plasma therapy vaccine finally cure Covid Watch PrenticeBourby March fall http T3h3ud gminews Social source EchoBox
 Delhi high court directed medical equipment importer price Covid Rapid Test ordered Indian Council Medical Research rate http trib PRK
 Just Eight Covid positive case detected Bihar total number state risen Four Gopalganj district East Champaran district diagnosed virus report AnarabhiTeary Bihar Covid
 Zealand shopping party mark start Covid free life http reut
 must adhere 'lanta curfew' called Prime Minister metro rail state city service state remain closed tomorrow Yogi Adityanath Uttar Pradesh Chief Minister CovidIndia Covid More ndtv NDTV
 vacancy halve fall Covid lockdown economy business live http theguardian business live unemployment benefit claimant count Covid lockdown economy business Live term Autofeed gminews Social source EchoBox
 About people England currently infected Covid study http reut
 CovidCrisis present testing UttarPradesh designated Covid patient hospital melalashi http indiastoday india story handling Covid ramping testing special focus certain district
 Here fact anti malaria drug President Trump promoted Covid scant evidence efficacy despite warning cause heart problem Covid patient http nyti hWec
 ✓ Us completed at 9:31 AM

Figure D.2: Extracted True-Covid class prototypical examples

