



UNIVERSITÀ  
di CAMERINO



# Combining Text Classification and Fact Checking to Detect Fake News

by

Sajjad Ahmed

submitted in accordance with the requirements  
for the degree of

DOCTOR OF PHILOSOPHY

in the subject

Computer Science

at

School of Advanced Studies  
University of Camerino

Supervisor: Prof. Dr. Knut Hinkelmann  
Co-Supervisor: Prof. Dr. Flavio Corradini

July, 2021

## Dedication

Dedicated to my family—I love you all.

# Acknowledgment

First of all, I thank God for giving me the strength and encouragement I needed during all the challenging and stressful moments in completing this thesis.

I owe my gratitude to all the people who have made this work possible. I was fortunate to have Prof. Dr. Knut Hinkelmann as my advisor during the development of this thesis, whom I thank for giving me the freedom to develop and pursue my research ideas that have led to this work. I am very grateful to Knut for guiding me through my PhD. His encouragement and patience with me as a young graduate student, and then his support throughout graduate school, and his ever-enthusiastic and practical approach to research and life have invaluable shaped my perspective and will continue to do so in the future. I would like to thank Knut for giving me the freedom and opportunity to collaborate with other researchers during my PhD, especially for working with the intelligent information systems group at FHNW. Knut's patient and persistent approach towards research is very inspiring. Thank you Knut for the several productive brainstorming sessions that were a streaming flurry of ideas and creative, dynamic and bold approach to solving practical real-world problems has gone a long way in shaping my thinking.

The realization of this thesis would not have been possible without the indirect support of Prof. Dr. Flavio Corradini. I will always be grateful to him.

During my stay at FHNW, Switzerland, I attended Research Methods lectures given by Prof. Alta van der Merwe. Her one of the brilliant lectures "how to write a thesis", were the reason why I could imagine writing a PhD thesis in the first place.

My research journey has been exciting and fruitful thanks to the awesome bunch of researchers with whom I had the pleasure of working with and the privilege of becoming friends along the way.

I would also like to thank Dr. Deepayan Bhowmik from the University of Stirling, Scotland. During my stay and work under Dr. Bhowmik, I learned a lot, especially working with a group of researchers from Stirling who were working on sentiment analysis was helpful for me to understand my research problem.

I owe special thanks to Prof. Emanuela Merelli, Prof. Leonardo Mostarda, Prof. Michele Loreti, Prof. Andrea Polini, Prof. Barbara Re, Prof. Luca Tesei, Prof. Francesco Tiezzi, Prof. Diletta Romana Cacciagrano, Prof. Rosario Culmone, Prof. Roberto Gagliardi, Prof. Fausto Marcantoni, Prof. Andrea Morichetta, Prof. Grid Thoma, Dr. Marco Piangerelli and Dr. Fabrizio Fornari.

I am grateful to Dr. Michela Quadrini for her patient, thoughtful and honest research style that gave me an idea for my research during my early days in UNICAM. My journey to graduate school was made so much easier thanks to the outstanding staff at UNICAM and FHNW.

Finally, I would like to give special thanks to my sister Naheed Ch and brother Ch. Zahid Ali Advocate who were always ready to support me whenever I needed it. I would also like to thank my wife, who has always supported and encouraged me. They are the biggest source of my energy and have made me who I am today. I dedicate this thesis to my parents Ch. Said Muhammad Advocate and Mrs. Perveen Ch who have constantly encouraged me to reach new heights. They have always sacrificed their lives to raise me to be a better person.

Thank you everyone!

# Abstract

Due to the widespread use of fake news in social and news media, it is an emerging research topic gaining attention in today's world. In news media and social media, information is spread at high speed but without accuracy, and therefore detection mechanisms should be able to predict news quickly enough to combat the spread of fake news. It has the potential for a negative impact on individuals and society. Therefore, detecting fake news is important and also a technically challenging problem nowadays. The challenge is to use text classification to combat fake news. This includes determining appropriate text classification methods and evaluating how good these methods are at distinguishing between fake and non-fake news. Machine learning is helpful for building Artificial intelligence systems based on tacit knowledge because it can help us solve complex problems based on real-world data. For this reason, I proposed that integrating text classification and fact checking of check-worthy statements can be helpful in detecting fake news. I used text processing and three classifiers such as Passive Aggressive, Naïve Bayes, and Support Vector Machine to classify the news data. Text classification mainly focuses on extracting various features from texts and then incorporating these features into the classification. The big challenge in this area is the lack of an efficient method to distinguish between fake news and non-fake news due to the lack of corpora. I applied three different machine learning classifiers to two publicly available datasets. Experimental analysis based on the available dataset shows very encouraging and improved performance. Simple classification is not quite accurate in detecting fake news because the classification methods are not specialized for fake news. So I added a system that checks the news in depth sentence by sentence. Fact checking is a multi-step process that begins with the extraction of check-worthy statements. Identification of check-worthy statements is a subtask in the fact checking process, the automation of which would reduce the time and effort required to fact check a statement. In this thesis I have proposed an approach that focuses on classifying statements into check-worthy and not check-worthy, while also taking into account the context around a statement. This work shows that inclusion of context in the approach makes a significant contribution to classification, while at the same time using more general features to capture information from sentences. The aim of this

challenge is to propose an approach that automatically identifies check-worthy statements for fact checking, including the context around a statement. The results are analyzed by examining which features contributes more to classification, but also how well the approach performs. For this work, a dataset is created by consulting different fact checking organizations. It contains debates and speeches in the domain of politics. The capability of the approach is evaluated in this domain. The approach starts with extracting sentence and context features from the sentences, and then classifying the sentences based on these features. The feature set and context features are selected after several experiments, based on how well they differentiate check-worthy statements. Fact checking has received increasing attention after the 2016 United States Presidential election; so far that many efforts have been made to develop a viable automated fact checking system. I introduced a web based approach for fact checking that compares the full news text and headline with known facts such as name, location, and place. The challenge is to develop an automated application that takes claims directly from mainstream news media websites and fact checks the news after applying classification and fact checking components. For fact checking a dataset is constructed that contains 2146 news articles labelled fake, non-fake and unverified. I include forty mainstream news media sources to compare the results and also Wikipedia for double verification. This work shows that a combination of text classification and fact checking gives considerable contribution to the detection of fake news, while also using more general features to capture information from sentences.

# Publication List

The publications which contributed to this thesis are the following:

1. **Ahmed S.**, Hinkelmann K., Corradini F. (2022) Fact Checking: An Automatic End to End Fact Checking System. In: Lahby M., Pathan AS.K., Maleh Y., Yafooz W.M.S. (eds) Combating Fake News with Computational Intelligence Techniques. Studies in Computational Intelligence, vol 1001. Springer, Cham. [https://doi.org/10.1007/978-3-030-90087-8\\_17](https://doi.org/10.1007/978-3-030-90087-8_17)
2. **Ahmed S.**, Balla K., Hinkelmann K., Corradini F. (2021) Fact Checking: Detection of Check Worthy Statements Through Support Vector Machine and Feed Forward Neural Network. In: Arai K. (eds) Advances in Information and Communication. FICC 2021. Advances in Intelligent Systems and Computing, vol 1364. Springer, Cham. [https://doi.org/10.1007/978-3-030-73103-8\\_37](https://doi.org/10.1007/978-3-030-73103-8_37).
3. **Ahmed, S.**, Hinkelmann, K., & Corradini, F. Development of Fake News Model Using Machine Learning through Natural Language Processing. World Academy of Science, Engineering and Technology, International Journal of Computer and Information Engineering Vol: 14, No:12, 2020.
4. **Ahmed, S.**, Hinkelmann, K., & Corradini, F. (2019). Combining machine learning with knowledge engineering to detect fake news in social networks-a survey. In Proceedings of the AAI 2019 Spring Symposium on Combining Machine Learning with Knowledge Engineering (AAAI-MAKE 2019). Stanford University, Palo Alto, California, USA, March 25-27, 2019.

During the research period, I collaborated on other research works which resulted in the following publications:

1. Ghosh A., **Ahmed S.** (2021) Shared Medical Decision-Making and Patient-Centered Collaboration. In: Dutta G., Biswas A., Chakrabarti A. (eds) Modern Techniques in Biosensors. Studies in Systems, Decision and Control, vol 327. Springer, Singapore. [https://doi.org/10.1007/978-981-15-9612-4\\_10](https://doi.org/10.1007/978-981-15-9612-4_10)
2. Khan, M. A., Hong, L., & **Ahmed, S.** (2020). Hand Gesture Recognition of Dumb Person using one against All Neural Network (IJCSIS) International Journal of Computer Science and Information Security, Vol. 18, No. 04, April 2020.
3. Ghosh, A., Liaquat, S., & **Ahmed, S.** (2020). Healthcare-Internet of Things (H-IoT) can assist and address emerging challenges in healthcare. International Journal of Science and Innovative Research, Vol. 1, No. 02, Dec 20

# Table of Contents

Acknowledgment.....	i
Abstract.....	iii
Publication List.....	v
Table of Contents.....	vi
List of Figures.....	xii
List of Tables.....	xvi
<b>1 Introduction.....</b>	<b>1</b>
1.1 Problem Statement.....	2
1.2 Thesis Statement and Research Questions.....	5
1.3 Research Challenges .....	7
1.3.1 Challenge 1: Text Classification .....	7
1.3.2 Challenge 2: Check-Worthy Statements .....	7
1.3.3 Challenge 3: Fact Checking .....	7
1.4 Contributions.....	8
1.4.1 Contribution 1: A Procedure for Fake News Detection .....	8
1.4.2 Contribution 2: Detection of Fake News through Classifying the Text.....	9
1.4.3 Contribution 3: Identification of Check-Worthy Statements .....	9
1.4.4 Contribution 4: Development of a Fact-Checking Application .....	9
1.5 Structure of the Thesis .....	10
<b>2 Literature Review .....</b>	<b>12</b>
2.1 Fake News.....	12
2.2 Current Approaches for Fake News Detection .....	15
2.2.1 Manual Approaches .....	16
2.2.2 Role of Automation in Fake News.....	17



---

2.3	Classification for Fake News Detection.....	18
2.3.1	Classification Techniques .....	18
2.3.3	Similar Application Areas.....	20
2.3.4	Strengths and Weaknesses .....	21
2.4	Fact Checking .....	22
2.4.1	Check-Worthy Claims.....	23
2.4.1.1	The Context in Identification of Check-Worthy Claims.....	24
2.4.2	Knowledge-Based Approaches .....	26
2.4.3	Automated Approaches.....	26
2.5	Combination Approaches.....	28
2.6	Interdisciplinary Approaches .....	29
2.6.1	Media Literacy .....	30
2.6.2	Critical Thinking for Citizens .....	30
2.6.3	Empower Civil Society .....	30
2.7	Discussion.....	31
<b>3</b>	<b>Research Method .....</b>	<b>32</b>
3.1	Design Science Research .....	32
3.2	Research Approach .....	34
3.3	Research Design.....	35
3.3.1	Awareness of the Problem .....	37
3.3.2	Suggestion.....	38
3.3.3	Design and Development.....	40
3.3.4	Evaluation .....	40
3.3.5	Conclusion .....	40
3.4	Discussion.....	41
<b>4</b>	<b>Problem of Detecting Fake News and Overview of the Solution Approach.....</b>	<b>42</b>
4.1	Problem.....	42

---

4.2	Solution Approach .....	45
4.3	Approach and Objectives .....	46
4.3.1	Text Classification .....	46
4.3.2	Fact-Checking through Check-Worthy Statements.....	47
4.3.3	Automated Fact-checking .....	49
4.4	Solution Architecture .....	51
4.5	Discussion and Structure of the Research.....	52
<b>5</b>	<b>Fake News Detection through Classification.....</b>	<b>54</b>
5.1	Introduction.....	54
5.1.1	Role of Machine Learning in Fake News Detection.....	57
5.2	Methodology for Fake News Detection through Classification.....	59
5.2.1	Dataset Exploration.....	59
5.2.2	Missing Values and Correlation.....	62
5.2.3	Models Description .....	66
5.2.3.1	Naïve Bayes .....	66
5.2.3.2	Support Vector Machine .....	66
5.2.3.3	Passive Aggressive.....	67
5.2.3.4	Logistic Regression.....	67
5.2.3.5	Neural Network.....	68
5.2.3.6	Multilayer Perceptron (MLP).....	70
5.2.3.7	Recurrent Neural Networks (RNN) .....	70
5.2.4	Model Comparison.....	71
5.3	Model Development for Fake News Detection.....	72
5.3.1	Pre-Processing.....	73
5.3.2	Stop Word Elimination .....	74
5.3.3	Count Vectorization .....	74

---

5.3.4 TF-IDF .....	75
5.4 Experimental Setup and Evaluations .....	75
5.5 Evaluation Methods .....	80
5.5.1 Confusion Matrix .....	80
5.6 Results and Discussion .....	82
5.7 Conclusion .....	84
<b>6 Fact-checking: Identification of Check-Worthy Statements .....</b>	<b>87</b>
6.1 Problem Statement .....	88
6.2 Identification of Check-Worthy Statements .....	89
6.3 Methodology for Identifying Check-Worthy Statements .....	92
6.3.1 Dataset.....	93
6.3.2 Feature Analysis.....	95
6.4 Using Context to identify Check-Worthy Statements.....	98
6.5 Learning from an Imbalanced Dataset .....	99
6.6 Models.....	100
6.6.1 Logistic Regressing and SVM with Linear Kernel .....	101
6.6.2 Feed Forward Neural Network .....	102
6.7 Results.....	103
6.7.1 Final Test Set Results.....	104
6.7.2 Best Performing Features.....	107
6.7.3 Context Features .....	108
6.8 Discussion.....	109
<b>7 Automated Fact-checking for Fake News Detection.....</b>	<b>111</b>
7.1 Problem Statement .....	113
7.2 Fact-checking.....	114
7.2.1 Manual Fact Checking .....	115
7.2.2 Automated Fact Checking.....	115

---

7.3	Role of Knowledge Engineering in Fact Checking.....	116
7.3.1	Meta-Data .....	118
7.3.2	News Content Models.....	119
7.3.2.1	Knowledge-Based Content Models.....	120
7.3.2.2	Style-Based Content Models.....	122
7.3.2.3	Social Context Content Models .....	122
7.3.3	Drawbacks with Existing Fact-Checking Applications .....	123
7.4	Methodology .....	124
7.4.1	Proposed Approach for Automation .....	124
7.4.1.1	Automation Challenges.....	125
7.4.1.2	Linguistic Analysis .....	126
7.4.2	Dataset Exploration and Analysis .....	126
7.5	Discussion.....	129
<b>8</b>	<b>Development and Evaluation.....</b>	<b>131</b>
8.1	Web Application Development Task.....	131
8.2	Front End Display for our Fact-Checking System.....	134
8.2.1	Aggregation.....	135
8.2.2	Key Points for the Fact-Checking System .....	135
8.3	Text Retrieval.....	137
8.4	Source Collection.....	138
8.5	Fact Checking Module.....	140
8.6	Repository of Fact-Checked Claims .....	141
8.7	Results.....	141
8.7.1	Example 1: Fake.....	142
8.7.2	Example 2: Non-Fake .....	144
8.7.3	Example 3: Unverified Claim .....	146

---

8.8 Conclusion from the Evaluation .....	149
8.9 Discussion .....	151
<b>9 Conclusion .....</b>	<b>152</b>
9.1 Contributions.....	152
9.2 Future Directions .....	154
9.3 Concluding Remarks.....	156
<b>Bibliography .....</b>	<b>158</b>
<b>Appendix-A: Configuration of Fact Checking Query Submission .....</b>	<b>174</b>
<b>Appendix-B: Configuration of Fact checking application with Wikipedia and other news media organizations .....</b>	<b>176</b>
<b>Appendix-C: Search other news media sites for comparison .....</b>	<b>181</b>

# List of Figures

Figure 1.1:	Proposed diagram for fake news detection.....	8
Figure 1.2:	Thesis Map.....	11
Figure 2.3:	Research directions for fake news detection (Shu et al., 2016).....	13
Figure 2.4:	Fact checking process (Vlachos and Riedel, 2014).....	22
Figure 2.5:	Comparison sites and fake news sites (Allcott et al., 2019).....	28
Figure 3.6:	Design science research cycles (adapted from Hevner (2007).....	33
Figure 3.7:	Induction and deduction (adapted from Trochim (2006)).....	35
Figure 3.8:	General Methodology of Design Science Research (adapted from Vaishnavi and Kuechler (2004) and enhanced with elements from Peppers et al. (2008)).....	36
Figure 3.9:	Proposed research methodology.....	37
Figure 3.10:	Development of classification model steps.....	39
Figure 3.11:	Fact checking model steps.....	39
Figure 4.12:	World map displaying trust in platforms.....	42
Figure 4.13:	Text classification module (Data Driven) as part of overall procedure.....	46
Figure 4.14:	Identifying check-worthy statements as part of the overall procedure.....	48
Figure 4.15:	Fact-checking (Knowledge Driven) is the final part of the overall procedure.....	50
Figure 4.16:	Proposed combination diagram for fake news detection.....	51
Figure 5.17:	Text classification proposed diagram for fake news detection (General View).....	55
Figure 5.18:	General schema for machine learning methods.....	58
Figure 5.19:	Text classification development targeted area.....	59
Figure 5.20:	Dataset Row Structure Example.....	60
Figure 5.21:	(a): Class distribution (a) Kaggle Dataset (2016).....	60
Figure 5.21:	(b): Signal media news dataset 2016.....	61
Figure 5.22:	Dataset class labeling chart.....	61
Figure 5.23:	Fake and Real news sentence level comparison (Spline Plotting).....	62
Figure 5.24:	Data exploration (Correlation, Stability, ID-ness and Missing).....	63
Figure 5.25:	(a): Sentence wise data exploration (Line Plotting).....	64
Figure 5.25:	(b): Fake and real news data comparison (Bell Curve).....	64

Figure 5.25:	(c): Fake and real news data comparison (Step Area).....	65
Figure 5.26:	Fake news detection model developments.....	73
Figure 5.27:	Data cleaning steps in NLP starting from Raw dataset to Machine Learning models.....	74
Figure 5.28:	Word cloud for news articles.....	76
Figure 5.29	4-Fold Cross-Validation (Kohavi, 1995).....	77
Figure 5.30:	Accuracy comparison with different algorithms (a) PA with SVM (b) PA with LR (c) PA with SVM (d) PA with NB (e) SVM with NB (f) NB with SVM (g) SVM with LR (h) SVM with NB.....	78
Figure 5.31:	Confusion matrix for NB and SVM.....	81
Figure 5.32:	Performance metrics.....	83
Figure 5.33:	Metadata classification.....	85
Figure 6.34:	Proposed diagram for check-worthy statements (General View).....	87
Figure 6.35:	Proposed diagram for fact checking (Inner View).....	88
Figure 6.36:	Information verification pipeline (Pepa et al., 2019).....	90
Figure 6.37:	Proposed diagram for identification of Check Worthy Claims.....	91
Figure 6.38:	Class distribution of sentences in dataset.....	94
Figure 6.39:	Number of sentences by each speaker.....	94
Figure 6.40:	Syntactic dependence parse tree of example sentence.....	97
Figure 6.41:	Context window example from the dataset, trump state union speech.....	98
Figure 6.42:	Initial experiments with resembling methods (Logistic Regression).....	102
Figure 6.43:	Initial SVM experiments metrics score for check-worthy claims.....	102
Figure 6.44:	Initial SVM experiments on Hyperlane.....	102
Figure 6.45:	Loss and accuracy values during the training of the final FNN model.....	104
Figure 6.46:	Confusion matrix for FNN and SVM respectively.....	106
Figure 6.47:	All speeches results.....	107
Figure 6.48:	Context feature metrics results.....	109
Figure 7.49:	Proposed diagram for fact- checking (General View).....	112
Figure 7.50:	Proposed diagram for fact- checking (Inner View).....	113
Figure 7.51:	Example knowledge graph (Zhou et al., 2019).....	117
Figure 7.52:	News content Models.....	119

---

Figure 7.53:	System Framework for automated fact checking.....	126
Figure 7.54:	Class distribution of sentences.....	127
Figure 7.55:	Dataset class labelling chart.....	128
Figure 7.56:	Claim label.....	128
Figure 7.57:	Claim labelling percentage.....	129
Figure 7.58:	important factors that involved in dataset features.....	129
Figure 8.59:	Search Panel.....	134
Figure 8.60:	False, True and Unverified Statements Percentage.....	134
Figure 8.61:	Claim input panel for users.....	135
Figure 8.62:	Results Comparison with Wikipedia.....	137
Figure 8.63:	The Web-applications main interface.....	138
Figure 8.64:	Comparison of True and False Statements.....	139
Figure 8.65:	Source collection from main stream media and top search results.....	140
Figure 8.66:	Claim input panel for users.....	143
Figure 8.67:	An Example of a fake Prediction (General).....	143
Figure 8.68:	An Example of a fake Prediction with claim and evidence.....	144
Figure 8.69:	Claim input panel for users.....	145
Figure 8.70:	An Example of a Non fake Prediction (General).....	146
Figure 8.71:	An Example of a Non fake Prediction with claim and evidence.....	147
Figure 8.72:	Claim input panel for users.....	148
Figure 8.73:	An Example of an Unverified Prediction (General).....	148
Figure 8.74:	An Example of Unverified result Prediction with claim and evidence.....	149



# List of Tables

Table 3.1:	Comparison of design science research processes (Offerman 2009).....	34
Table 4.2:	Fact checking websites comparison.....	49
Table 5.3:	Seven types of fake news.....	56
Table 5.4:	Accuracies after applying machine learning models.....	83
Table 6.5:	US Presidential debate check-worthy statements example.....	90
Table 6.6:	Examples of ill-defined sentences from transcripts.....	93
Table 6.7:	Overview of extracted features from the target sentence.....	95
Table 6.8:	Entity types detected in check-worthy sentences and the whole dataset.....	96
Table 6.9:	Results for SVM and FNN model.....	105
Table 6.10:	Classification results for not check-worthy sentences.....	105
Table 6.11:	Metrics scores for each test file separately.....	107
Table 6.12:	Separate features scores in Feed Forward Neural Network. Ordered by the highest F1 score.....	108
Table 6.13:	Performance of FNN model without context features.....	109
Table 7.14:	Categorization of claims on the basis of facts Hassan et al. (2015).....	124
Table 7.15:	Dataset row structure example set.....	127
Table 8.16:	A Fake claim with assessment and explanation.....	140
Table 8.17:	A Non fake claim with assessment and explanation.....	143
Table 8.18:	Unverified claim with assessment and explanation.....	145



# 1 Introduction

Millions of people around the world use digital media and social networks to get their news. Fake news stories that spread on these sites quickly become a major problem for these readers. The term fake news has been defined by (Lazer et al., 2018a) as fictitious information that mimics the content of the news media in form but not in the organizational process. Other definitions define it as news articles that are intentionally false and could mislead readers (Allcott & Gentzkow, 2017). They consist of deliberate misinformation or hoaxes disseminated through traditional print and broadcast news media or online social media<sup>1</sup> with the intent to mislead and harm an agency, institution, or individual, and/or to benefit financially or politically (Himma 2017., Hunt et al., 2016; Schlesinger et al., 2017), often using sensationalist, dishonest, or fictitious headlines to increase readership, online sharing, and Internet click revenue. News portals used to be the main target of fake news but in recent years, the interest is directed towards social media, mostly Twitter or Facebook (Popat, Mukherjee, Strötgen, & Weikum, 2016).

With these false claims, words lose their meaning and then there is no more “real news” but only bigger lies. In many cases, people are not necessarily ignorant but the formation of news seems sufficiently legitimate to believe<sup>2</sup> (Haigh & Kozak, 2017).

(Flintham et al., 2018) report that one-third of their survey respondents from the United Kingdom had the experience of being exposed to fake news that they initially believed to be true. Many of the articles published during the 2016 United States Presidential elections were deliberately constructed to manipulate and influence the audience to lean toward a particular direction. According to Reuters Institute Report (Rasmus, 2019), only 24% of respondents think social media does a good job of separating fact from fiction, compared to 40% for news media. The lack of a combination of rules and viral algorithms leads to low-quality writing and allows fake news to spread quickly (Nielsen 2017). Social media platforms are the common breeding ground for fake news and sometimes they find their way into the

---

<sup>1</sup><https://www.change.org/p/department-of-information-and-communications-technology-misleading-and-fake-information-is-at-stake>  
<sup>2</sup><https://observer.com/2017/01/fake-news-russia-hacking-clinton-loss/>

mainstream media (Himma 2017). In the past, these types of fake news<sup>3</sup> have affected the economy with stock price losses and large-scale business and political damage (Vosoughi, Roy, & Aral, 2018).

## 1.1 Problem Statement

Fake news detection is considered a challenging task (Hassan, Li, & Tremayne, 2015) that requires multidisciplinary efforts (Lazer et al., 2018a). Due to the widespread prevalence of fake news in social and online news media, it has become an emerging research topic that has gained global attention. In news media and social media, information is spread at high speed without accuracy and therefore detection methods should be able to predict news quickly enough to deal with the spread of fake news. A report by the Pew Research Center identified the internet as an important resource of news for people under 30 in the U.S. and the second most important overall source after television (Pew Research Center, 2008). Social media sites are used for everyday chit-chat and for sharing news and other important information (Java, Song, Finin, & Tseng, 2007; Naaman, Boase, & Lai, 2010). More and more people are using social media as a source of news (Kwak, Lee, Park, & Moon, 2010; Stassen, 2011). Recent findings show that (i) 71% of U.S. adults have seen fake political news (ii) 88% of U.S. adults have felt confused about basic facts due to fake news stories and (iii) certain fake news stories have been more widely shared on social media than most popular real news<sup>4</sup> (Silverman, 2016). Detecting fake news requires knowledge and is typically done by humans, as researchers<sup>5</sup> explain that false information is spread faster, deeper and wider than truth in all categories (Parikh & Atrey, 2018). Fake news detection done by humans is a time-consuming process (Oshikawa, Qian, & Wang, 2018). Fake content producers are increasingly using more advanced methods to generate fake news so that readers think it is legitimate<sup>6</sup>. It is difficult for humans to detect fake news; one method would be to manually identify the news item and determine if it is fake through extensive research and/or knowledge of the topic being covered. To classify millions of text documents manually is an expensive and time-consuming process (Nidhi & Gupta, 2011). Traditional approaches based on verification by humans and expert

---

<sup>3</sup><https://socialsimulator.com/understanding-10-types-of-fake-news/>

<sup>4</sup><https://www.pewresearch.org/fact-tank/2020/04/21/how-americans-see-climate-change-and-the-environment-in-7-charts/>

<sup>5</sup><https://medium.com/data-from-the-trenches/text-classification-the-first-step-toward-nlp-mastery-f5f95d525d73>

<sup>6</sup><https://www.theverge.com/2019/2/14/18224704/ai-machine-learning-language-models-read-write-openai-gpt2>

journalists do not scale with the volume of news content generated online (Tschatschek, Singla, Gomez Rodriguez, Merchant, & Krause, 2018a).

Text classification is the fundamental task in Natural Language Processing (NLP)<sup>7</sup> and researchers have addressed this problem quite extensively (Conroy, Rubin, & Chen, 2015c). Fact checking effort can be reduced if we could focus only on news that is potentially fake (Rubin, Conroy, Chen, & Cornwell, 2016a), but detection done by humans is a time-consuming process; humans can perform a supportive role in identifying fake news identification (Burkhardt, 2017). A method can be developed to identify the text by first focusing on the content of the news and then checking the relevant features in-depth. Human expertise is used to check whether the news appears in other trusted media sources. For statement comparison, the goal is to build an assistant that accesses the knowledge base containing the needed facts so that we can compare check-worthy statements with known facts. This could be helpful to separate fake news articles from non- fake news articles.

The above discussion shows that the phenomenon of fake news is an important topic that requires scientific attention to determine how fake news is spread. Different groups introduced different models; some of them used data-oriented models and others applied knowledge. Below I discuss both sides separately with existing applications; then our proposed combination approach is defined.

Text classification is mainly about extracting different features of the text, which are then included in the classification. Then the best algorithm is selected which performs well and distinguishes between fake and non-fake (Nidhi & Gupta, 2011). Emergent<sup>8</sup> is a real-time data-driven approach for rumor detection. It works automatically to track rumors related to social media, however, rumors that are input by a human are not automated. One problem with this is that most classification approaches are supervised, so we need a prior dataset to train our model but as mentioned earlier, obtaining a reliable fake news dataset is a very time-consuming process.

On the other side, fact-checking techniques mainly focus on checking the fact of the news based on known facts. Fact checking is a challenging and time-consuming process and with

---

<sup>7</sup><https://science.sciencemag.org/content/359/6380/1146>

<sup>8</sup>[www.emergent.info](http://www.emergent.info)

today's vast amounts of information, manual fact checking is not feasible (Wu, Cheng, & Chai, 2018). On the other hand, despite great efforts by researchers we still do not have automated and context-aware fact-checking engines that are trustworthy enough to replace human fact checkers. There are three types of fact-checking techniques; Knowledge Linker (Ciampaglia et al., 2015a), PRA (Lao & Cohen, 2010), and PredPath (Shi & Weninger, 2016). The prediction algorithms that use knowledge for fact checking are DegreeProduct (Shi & Weninger, 2016), (Katz, 1953), Adamic & Adar (2003) and Jaccard coefficient (Julian, 2016). Some fact-checking organizations offer online fact-checking services, such as Snopes<sup>9</sup>, PolitiFact<sup>10</sup>, Fiskkit<sup>11</sup>, and Hoaxy<sup>12</sup>. Collection, detection and analysis to check online misinformation are part of Hoaxy. The criteria they follow is to check whether the news is fake or not by simply referring to domain experts, individuals or organizations on that particular topic. They also follow unbiased information and data sources (e.g. peer-reviewed journals, government agencies or statistics). Translating the operations performed by human fact checkers into program code or rules is difficult and poses challenges, especially because these operations vary from case to case (Dey et al., 2017; Wu et al., 2018).

An important issue is the dissemination speed related to the information in social media networks; this is a challenging problem that requires attention and alternative solutions. Identification of check-worthy statements, one of the subtasks in the process of fact checking, reduces the time needed for fact check (Hassan et al., 2015). When news is identified as fake, the existing techniques block it immediately due to its functionality as we cannot replace it; but when news is identified as fake we need at least an expert opinion or verification before blocking that particular news. The existing fake news systems are based on predictive models that simply classify whether the news is fake or not. The major challenge in these cases is to train the model, but this is impossible due to the unavailability of corpora.

An alternative approach is needed that combines knowledge with data and fact checking of check-worthy claims that look deeply at the content of the news with expert opinions, and at the same time can detect the fake news. An important motivation for my research is an effort to introduce an automatic fact-checking application. In this thesis, I focus on political news as one of the domains most affected by fake news and contribute to fact checking.

---

<sup>9</sup>[www.snopes.com](http://www.snopes.com)

<sup>10</sup>[www.politifact.com](http://www.politifact.com)

<sup>11</sup>[www.fiskkit.com](http://www.fiskkit.com)

<sup>12</sup>[www.hoaxy.iuni.iu.edu](http://www.hoaxy.iuni.iu.edu)

## 1.2 Thesis Statement and Research Questions

Based on the previous statements of research problem and objectives, a thesis statement and research questions are defined. According to Creswell (2008) qualitative researchers usually write at least one main research question and sub-questions.

The following thesis statement guides the research project:

*“Combining text classification and fact checking of check-worthy statements allows detecting fake news”.*

The thesis statement can be phrased as a main research question:

*How can text classification be combined with fact checking of check-worthy statements to detect fake news?*

The main goal of this thesis is to combine text classification and identification of check-worthy statements before fact checking. Identification of check-worthy statement is a sub-task in fact checking that will reduce the time and burden of fact checkers.

From the main research question four sub-research questions are derived, which structure the research to provide the solution to the corresponding challenges discuss in Section 1.3.

*RQ1: What is the problem of detecting fake news?*

Objective: Understand the problem and design an overall approach

Based on the literature review (Chapter 2) and the problem formulation (Chapter 4), two tasks for fake news detection are derived: The first task is to identify potential fake news. This can be achieved through data-driven text classification. The second task is to check whether individual statements contained in the news are based on facts. This task is called fact checking and is itself composed of two subtasks: Identification of check-worthy statements and fact checking of these statements.

*RQ2: What methods can be used to identify potentially fake news?*

Objective: Explore alternative solutions and identify potentially fake news

To answer RQ2 first I examined existing techniques available for detecting fake and non-fake news articles (Chapter 5). I selected different text classification methods and applied them to a publicly available dataset. In the end, I compared the results with those of other existing solutions and concluded that our technique performed well and that the combination of text classification and machine learning improved the overall performance.

*RQ3: How can check-worthy statements for fact checking be automatically identified?*

Objective: To select and implement the determination of potentially falsified statements

The research question is answered in Chapter 6. The research started by identifying the problem of the time-consuming efforts when human-fact checkers verify a claim. To reduce the time required, the first task of fact checking is the identification of check-worthy claims that can be automated. We modeled it as a text classification task, that goes beyond the sentence-level approach seen in previous work (Hassan et al., 2015), by creating a window around the sentence. This window is referred to as the context for the classification model.

*RQ4: How can it be checked whether a statement is fact or fake?*

Objective: Do the fact checking of check-worthy statements

The research question is answered in Chapter 7 and Chapter 8. Misinformation can have serious consequences in just a few minutes; it is critical to detect it at an early stage (Heinrich & Borkenau, 1998). Current detection methods only provide the final result of whether the claim is false or not. It is important to provide a convincing explanation for misinformation and prevent its further spread. I have proposed text classification and fact checking which can be helpful in detecting fake news.

In the next section, I discuss the challenges that need to be overcome in creating an automated fact-checking system.



## 1.3 Research Challenges

Given the challenges associated with the research problem of detecting fake news, I first introduced the basic characteristics of the problem (Section 1.1) and then introduced the research questions (Section 1.2). In this section I discuss that fake news detection requires a lot of contextual information and domain knowledge. The challenges associated with my research are described below.

### 1.3.1 Challenge 1: Text Classification

The process of text classification is to assign tags or categories to the text according to its content. It is a fundamental task in Natural Language Processing (NLP) with wide applications such as sentiment analysis, stance detection, topic labeling, spam detection, and intent detection. The challenge is to use text classification to combat fake news. This includes determining appropriate text classification methods and evaluating how good these methods are at distinguishing between fake and non-fake news.

### 1.3.2 Challenge 2: Check-Worthy Statements

Fact checking is a multi-step process that begins with the extraction of check-worthy statements (Vlachos & Riedel, 2014). Identification of check-worthy statements is a subtask in the fact-checking process. Most work on fact checking focuses on determining the veracity of a claim, while the phase of identifying check-worthy statements has received less attention despite its importance. This problem is much more apparent today, where an enormous amount of information is rapidly disseminated across the globe and many people who see fake news stories report believing them (Silverman, 2016). The aim of this challenge is to propose an approach that automatically identifies check-worthy statements for fact checking, including the context around a statement.

### 1.3.3 Challenge 3: Fact Checking

The goal of fact checking is to assign a truth value to a claim. Fact checking has received increasing attention after the 2016 United States Presidential election; so far that many efforts

have been made to develop a viable automated fact-checking system. Fact checking is an intellectually demanding and time-consuming process and with today's vast amounts of information, manual fact checking cannot keep up (Wu et al., 2018). The challenge is to develop an automated application that takes claims directly from mainstream news media websites and fact checks the news after applying classification and fact-checking components.

## 1.4 Contributions

The contributions address the challenges presented in Section 1.3.

### 1.4.1 Contribution 1: A Procedure for Fake News Detection

The first contribution of the research is an overall approach for fake news detection as a contribution of classification and fact checking, where classification identifies potentially fake news which is then further analyzed for fact checking. Fact checking itself consists of two parts: First, check-worthy statements are identified in potentially fake news. These are then compared to known facts. While text classification is data driven, fact checking requires additional knowledge. The proposed overall approach is shown in Figure 1.1.

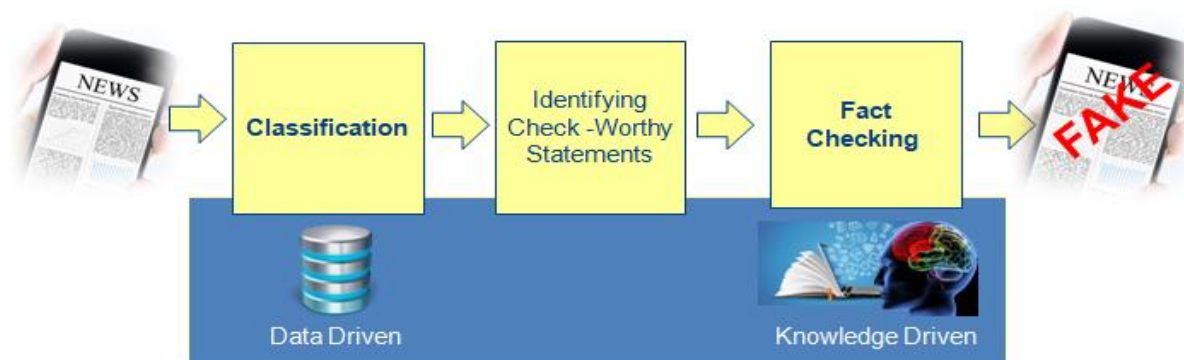


Figure 1.1: Proposed diagram for fake news detection

### 1.4.2 Contribution 2: Detection of Fake News through Classifying the Text

For text classification classifiers such as Passive Aggressive (PA), Naïve Bayes (NB) and Support Vector Machine (SVM) are compared. Experimental analysis using two publicly available datasets shows very encouraging and improved performance. The initial results gave an accuracy of 93% with the algorithm PA, 85% with NB and 84% with SVM. The developed system with accuracy up to 93% proved the importance of classification in detecting fake news.

### 1.4.3 Contribution 3: Identification of Check-Worthy Statements

We proposed an approach that focused on classifying statements into check-worthy and non-check-worthy, whilst taking into account the context around a statement. The approach starts with extracting sentences and context features from the sentences, and further classifying the sentences based on these features. The feature set and context features are selected after several experiments, based on how well they differentiate check-worthy statements. This work shows that the inclusion of context in the approach makes a significant contribution to classification, while using more general features to capture information from sentences. The results are analyzed by examining all the features used and which of these features contributes more to the classification.

### 1.4.4 Contribution 4: Development of a Fact-Checking Application

With the goal in mind, I have developed applications that directly integrate various components of fact checking starting from the collection of check-worthy statements from mainstream news media sources, through information retrieval from credible sources. Current search engines such as Google, Bing, Yahoo are used to search for claims that need to be fact-checked. These search engines collect the relevant claims from various sources such as an online encyclopedia (Wikipedia), major news sources (Fox News, CBS News, Washington Times, CNN, Huffington Post, New York Times) and forty other news channels. Then the news is fact checked based on known facts. I collect different news to expand the inventory of sources. The proposed system compares the statements and predicts the fact of the news and shows the aggregation of fake and not fake news.

## 1.5 Structure of the Thesis

The organization of this thesis reflects the order of the research process on fake news detection.

Chapter 2 provides the background to the research, in which a literature review is followed by the description of the evaluation metrics explanation. Chapter 3 illustrates how the design science research methodology is applied in this research. Chapter 4 starts with the problem relevance and then discussed the datasets used for all modules and benefits of problem relevance. Chapter 5 answers research question 2. The best model is selected after performing various classification tasks. Next, the tuned model is tested with publicly available datasets. The evaluation and results of these models are presented at the end. Chapter 6 answers research question 3. In this chapter, it is shown that identification of check-worthy statements is an important task in fact checking that can reduce the time and effort required to fact check a statement. Chapter 7 answers research question 4, examining the dataset and the proposed methodology for applying fact checking. Chapter 8 explains the identification of check-worthy statements and the evaluation results for the automated fact checking application. Chapter 9 presents the final conclusions, a summary of our findings, main contributions, and suggestions for extending this study.

The structure described can be seen in Figure 1.2.

<b>Introduction</b> . Motivation .Challenges . Problem Statement .Research Questions	Chapter 1	<b>Introduction and Awareness</b>
<b>Literature Review</b> . Fake News .Current Approaches . Classification Role .Fact Checking .Combination Approaches	Chapter 2	
<b>Research Method</b> . Design Science Research	Chapter 3	
<b>Problem of Detecting Fake News and Overview of the Solution Approach</b> . Solution Approach	Chapter 4	
<b>Fake News Detection through Classification</b> . Classification Approaches	Chapter 5	<b>Suggestion and Development</b>
<b>Fact Checking: Identification of Check-Worthy Statements</b> . Input for Fact Checking	Chapter 6	
<b>Automated Fact Checking for Fake News Detection</b> . Automated Fact Checking	Chapter 7	
<b>Development and Evaluation</b>	Chapter 8	
<b>Conclusion</b>	Chapter 9	

Figure 1.2: Thesis Map

## 2 Literature Review

This chapter gives an outline of the theoretical framework of the thesis. I target recent papers that deal with fake news and refer to the state of the art in fake news detection, the problem of fake news and the search for useful techniques (classification oriented and fact-checking oriented) that can help in the detection of fake news. I conclude that the useful method for automatically detecting fake news is not only a classical machine learning technique or latest fact-checking system, but the integration of these two could be more useful for detecting fake news detection and there is a need for a combination that unifies the different terminologies and definitions of the fake news domain. Starting from fake news (Section 2.1), types of fake news, current approaches to fake news detection (Section 2.2), the role of classification and classification approaches to fake news detection (Section 2.3), fact checking (Section 2.4), and finally, combination approaches (Section 2.5), Interdisciplinary approaches (Section 2.6). As stated in Chapter 1, the aim of this thesis is to develop an approach that detects fake news by combining text classification and fact checking.

### 2.1 Fake News

The increasing amount of fake information on the Internet, where any individual can post something, makes it difficult to evaluate credibility and trustworthiness. Fake news articles are intentionally written to convey false information for a variety of purposes, such as financial or political manipulation (Shu, Sliva, Wang, Tang, & Liu, 2017). The information that is repeated is more likely to be classified as true than information that has never been heard<sup>13</sup>. This is not the end as the false stories would lead to make the false memory<sup>14</sup>. Fake content in itself is not new, scams existed as early as the 16<sup>th</sup> century<sup>15</sup>.

Rubin, Conroy and Chen (2015) distinguish three types of fake information: a) serious fabrications (uncovered in mainstream or participant media, yellow press or tabloids); b)

---

<sup>13</sup><http://web.colby.edu/cogblog/2018/04/25/unraveling-the-mechanism-behind-a-lie-repeated-a-thousand-times-becomes-truth-a-cognitive-account/>

<sup>14</sup><https://ejop.psychopen.eu/index.php/ejop/article/view/456>

<sup>15</sup><https://www.cnn.com/2015/02/17/scams-hacking-spanish-prisoner.html>

large-scale hoaxes; c) humorous fakes (news satire, parody, game shows). Shu et al. (2016) make a distinction between fake news and different types of conspiracies.

The huge amount of information in the online world makes the time to evaluate each article limited. Therefore, the question arises whether this has an impact on credibility assessments. Existing work on fake news is based on linguistic approaches (Hancock, Santelli, & Ritchie, 2004), but linguistic analysis alone have a major drawback. They are limited because they do not take into account useful contextual information around a claim. Combining linguistic approaches with additional analysis such as semantic analysis (Feng & Hirst, 2013) is useful and improves classification performance, lexical and syntactic features detect writing styles commonly occur in fake news contents. Other work combines linguistic analysis with metadata attached to news stories. In a social network, metadata is used to analyze behaviours and patterns, that are often repeated in the spread of fake news (Cook, Waugh, Abdipanah, Hashemi, & Rahman, 2014). Social context-based methods combine features from user profiles (Castillo, Mendoza, & Poblete, 2011a), post content, news propagation (Wu & Liu, 2018a), and social networks. Despite very good results, this approach is only applicable in social media, where the timeline of information dissemination, can be easily tracked. Shu et al (2016) provided research directions for fake, which are shown in Figure 2.3.

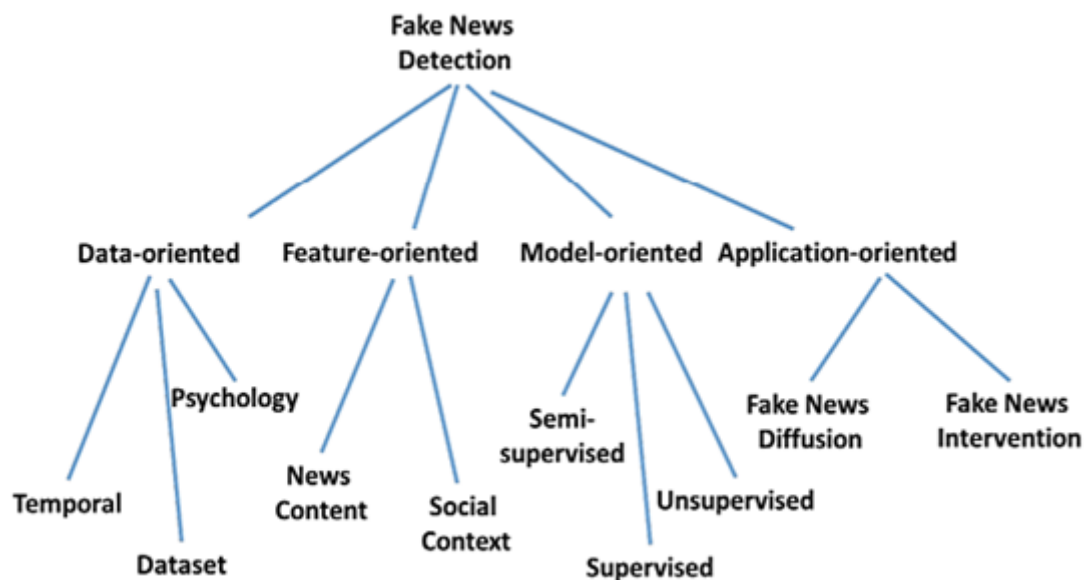


Figure 2.3: Research directions for fake news detection (Shu et al., 2016)

Fake news can be categorized into eight different types (Tandoc, Lim, & Ling, 2018).

- **Fabricated story** in which false evidence is used to deceive someone. These stories are completely disconnected from real facts and there is no evidence to support these claims (Rubin et al., 2016). An example of fabricated content was the story about Hillary Clinton, where an alien baby was adopted (Heller 2014). Another example of fabricated content was “Pope Francis supported Donald Trump” in U.S. 2016 Presidential Elections (Allcott & Gentzkow, 2017). News related to Donald Trump was shared about 30 million times on Facebook and news related to Hillary Clinton was shared 8 million times and half of the audience believed these stories (Allcott & Gentzkow, 2017). Teenagers from Macedonia participated in these conspiracies and automated advertisement bots to make money from these fabricated stories (Subramanian 2017).
- **Propaganda** stories are those stories in which Information is sourced towards a biased or misleading nature which is then used to promote a political cause or viewpoint. It refers to the news stories that are created by a political entity to influence public perception (Khaldarova & Pantti 2016). This type of news is not new as it was widely used during World War II and during the cold war. These stories are mostly used in election campaigns to mislead the audience; the main goal is to harm a particular political party (Jewitt et al., 2009). A recent example of this type of story is a propaganda campaign about an airstrike operation in Syria in 2018 (Medium.2018). Chen et al. (2013) proposed a study that examined individuals who were paid just for sharing their comments on social media sites and forums. In some cases, propaganda news is based on facts but contains a bias that promotes one side’s perspective.
- **Conspiracy** is a situation or event that creates a conspiracy without evidence (Fenster.1999). Usually, these stories refer to illegal actions carried out by individuals or at the government level. One of the popular examples of conspiracy theories is the Hilary Clinton election campaign involving a pedophile ring (Wikipedia 2017) and Seth Rich's email leaks (Wikipedia 2017).
- **Hoaxes** contain legitimate facts that are either false or inaccurate (Kumar, West, & Leskovec, 2016). These stories are a mixture of true and false content (Merriam-Webster. 2018). Famous examples of hoax stories include the false death of a celebrity Adam Sandler. Characterize hoax documents on Wikipedia and examine their impact in business (Kumar et al., 2016).



- **Biased** refers to stories that are one-sided. These can also be referred to as Hyper-partisan news that is biased towards one party or person (Martin et al., 2017). There are many examples that fall under this category, but few of them are discussed by (Tacchini, Ballarin, Della Vedova, Moret, & de Alfaro, 2017) such as the right-wing echo chamber and 4chans.
- **Rumors** are stories where the status is not yet confirmed or are ambiguous (Warren et al., 1951). Several studies have been conducted on rumors as it is a broad category. A famous example was during the time of 9/11 crisis when the child of Sandy Hook's child was killed during that incident and the suspect became a citizen<sup>16</sup>.
- **Clickbait** is the intentional use of false content on the web. This type of news refers to the newspaper era phenomenon known as yellow journalism (Chen, Conroy, & Rubin, 2015). This problem is rapidly increasing due to the proliferation of the web. Many users apply this technique to distort the content in order to get more traffic on the web<sup>17</sup>. Biyani et al. (2016) examine the unique linguistic styles found in clickbait articles.
- **Satire** refers to stories that contain a lot of irony and humor; they have no intent to cause harm but have the potential to deceive (Burfoot & Baldwin, 2009). Some popular examples of satire that publish satirical news are including The Onion<sup>18</sup> and Satire news<sup>19</sup>. Individuals who watch satirical news daily tend to be better informed about current events than those who consume other forms of news media Kohut, Morin, Keeter (2007).

## 2.2 Current Approaches for Fake News Detection

Current approaches focus mainly on content verification. As a result, they lack resilience to attempts to successfully verify a claim (Escrivá et al., 2013). Most existing work on fake news detection is based on linguistic approaches (Hancock et al., 2004), but linguistic

---

<sup>16</sup> <https://www.snopes.com/>

<sup>17</sup> <https://www.politifact.com/>

<sup>18</sup> <https://www.theonion.com/>

<sup>19</sup> <http://www.satirewire.com/content1/>

analyzes alone have a major drawback. They are limited as they do not take into account useful contextual information around a claim. Combining linguistic approaches with additional analysis, such as semantic analysis (Feng & Hirst, 2013) is more useful and improves classification performance. Lexical and syntactic features detect writing styles commonly found in fake news content. Other work combines linguistic analysis with metadata attached to news stories. In a social network context, metadata is used to analyze behaviors and patterns, that are often repeated in fake news propagation (Cook et al., 2014). Social context-based methods combine features from user profiles (Castillo et al., 2011a), post content, news propagation (Wu & Liu, 2018a), and social networks. Despite very good results, this approach is only applicable in the social media context where the timeline of information spread, can be easily tracked. News content models can be categorized into knowledge-based and style-based models. In content modeling, the main focus is on verifying the features and especially the factual sources which can help in detecting fake and genuine news (Shu et al., 2017). Before discussing the manual and automated approaches, I have presented the hierarchy of news content models in Figure 7.78 e.g. knowledge based, style based and social context based. All of these approaches and their sub approaches are described in Section 7.3.

### 2.2.1 Manual Approaches

Manual fact checking is a procedure that is done by people. It can be done by experts or ordinary people. It can further be further divided into expert based and crowd sourced based (Zhou et al., 2018). Expert-based manual fact checking is totally based on the experts in the field of fact checking; this is also the case when fact checkers authenticate specific news content. This approach is relatively simple and easy to perform, but very expensive as there is a limited number of professional fact checkers. The second approach is less reliable because it requires a large group of people to act as fact checkers. False news has become a major issue after the 2016 U.S. presidential election. Governments, newspapers, and social media organizations are working hard to separate fake and credible content. The first step in the identification phase is to understand what others are saying about the same topic (Ferreira et al. 2016). In stance detection, the estimation of the relativity of two different text pieces on the same topic and the stance of others (Mohammad, Sobhani, & Kiritchenko, 2017).

PHEME<sup>20</sup> was a three-year research project funded by European Commission from 2014-2017; it investigated natural language processing for rumor detection, stance classification (Lukasik, Cohn, & Bontcheva, 2015; Zubiaga, Aker, Bontcheva, Liakata, & Procter, 2018), contradiction detection and analysis of social media rumors. Existing stance detection approaches are based on embedding features on individual posts to predict the stance of that particular content. Feature extraction from text is integrated into classification models which then select the best algorithm that performs well (Nidhi & Gupta, 2011). Emergent<sup>21</sup> is a real-time data-driven rumor identification approach. It works automatically to track rumors associated with social media; however, rumors, where human input is required, have not been automated. The problem is that most classification approaches are supervised so we need a prior dataset to train our model but as mentioned earlier, obtaining a reliable fake news dataset is an extremely time-consuming process.

### 2.2.2 Role of Automation in Fake News

News producers are using new methods to distribute fake content because of the unique characteristics and challenges that make existing traditional ways ineffective or inapplicable. Another reason is that the existing systems are easily overwhelmed by the increasing fresh news content as it needs to be verified very often, especially in the case of social media. Fake news intentionally misleads readers and makes it difficult to detect that the information is false. Traditional media approaches cannot scale the volume, hence the need for automation. The role of automation is important in fact checking and automatic fact-checking methods are used to combat this problem (Thota, Tilak, Ahluwalia, & Lohia, 2018). Most of the automatic fact-checking systems consist of information retrieval and natural language processing. I discussed manual approaches in Section 2.2.1 highlighting the issues in the detection phase due to a large amount of data sharing. Different groups introduced different models; some of them worked data oriented, others worked only knowledge based. In the next section, I discuss maximum approaches used for automation in fake news detection.

---

<sup>20</sup>[www.pheme.eu](http://www.pheme.eu)

<sup>21</sup>[www.emergent.info](http://www.emergent.info)

## 2.3 Classification for Fake News Detection

Classification is important in detecting fake news and starts from text classification to detect fake news; thus maximum accuracy can be achieved (Araghavan, Wang, Guo, et al., 2020). Classification algorithms are used in different fields such as cancer tumor cell identification (Gligorijevic et al., 2014), drug classification in the medical field (Dunkel et al., 2008), predicting loan repayment of banks customers (Hamid et al., 2016), Sentiment analysis (Medhat et al., 2014), Email spam classification (Renuka et al., 2011), in recognizing pedestrian while driving (Yeo et al., 2009) and many others with promising results. In the next section, I discuss some working examples that show the importance of classification algorithms importance and the similarity of these similar application areas with fake news detection. The results show the importance of these algorithmic approaches and their role in automatic fake text detection. For further understanding, I discuss the role and usability of these approaches in other similar application areas.

### 2.3.1 Classification Techniques

There are many techniques used for text classification (Vlachos & Riedel, 2014). In the context of fake news detection, I discuss some of them with their strengths and weaknesses. Further details of each technique are available in different sections of this thesis.

- **Nearest Neighbor classifier:** Among the non-parametric methods, the Nearest Neighbor technique is popular. This technique is helpful in classification and regression prediction problems (Ahmed, 2017). It is known to determine the class of unlabeled documents (Parikh & Atrey, 2018). However, one problem with this technique is that when we have a high dimensions data set, the computational time increases (Vicario, Quattrociochi, Scala, & Zollo, 2019).
- **Support Vector Machine:** SVM gives good results when we compare it with the other algorithms, especially the speed of classification, learning speed, accuracy and tolerance to irrelevant features and noisy data (Goldani et al., 2020). I preferred SVM for fake news detection because it is a more researched algorithm nowadays<sup>22</sup>.

---

<sup>22</sup><http://www.journalism.org/2016/12/15/many-americans-believe-fake-news-is-sowing-confusion/>

However, it is still difficult to say which one is the best classifier for fake news as the selection of the classifier depends on the organizational requirements (Hiramath et al., 2019). Using the Support Vector Machine, I achieved 89% accuracy when applied to our proposed dataset. The details of the experiments and the results obtained are discussed in Chapter 5.

- **Classification using Neural Network:** This algorithm gives good results when we deal with multidimensional classification. However, for this reason, we need a large sample size and large storage space to achieve the maximum accuracy of the classifier. Moreover, it is intolerant to noise. Neural networks use special filters to detect the local structure of the image and identify whether it is a fake or not (Kan et al., 2015). CNN also performs well on semantic parsing (Scott et al., 2014), sentence modeling (Nal et al., 2014) and traditional NLP tasks (Ronan et al., 2011).
- **Bayesian Classification:** The basic idea of Naïve Bayes is that all features are independent (Lorent & Itoo, 2019). NB requires fewer records and less memory as it does not give good results when words are co-related with each other. Predicting Facebook posts by real or fake labeling can be done using NB (Jain et al., 2018). I applied the Naïve Bayes algorithm to my proposed dataset and I achieved 85% accuracy after implementation. The results are discussed in Chapter 5.
- **Term Graph Model:** The term graph model is an improved version of the vector space model (Salton et al., 1983). The term graph model is preferred especially when we have adjacent words and want to maintain a correlation between classes (Sebastiani et al., 1999). With regard to the term association vector space model weighting each term according to relative importance. Wang et al. (2005) presented a new model for text documents which comprises vector space and co-occurring together. The main idea behind this work is to mine the associations among terms and after that capture all information in a graph shape.

### 2.3.2 Applications for Fake News Detection

There is a large body of related works that address the problem of fake news detection. The methods are mainly based on predictive models for fake news detection. Fake news detection

using crowd signals approach by using crowd signals they took motivations from Facebook flags method. An algorithm DETECTIVE, which performs Bayesian inference for fake news detection while learning from flag accuracy (Andrea et al., 2014). It selects small subsets of news every day and sends them to an expert (for verification within that particular domain), who then stops the spread of fake news based on the opinion received from the expert (Vlachos & Riedel, 2014). We can divide news sources into two categories: writing quality and sentiment. Real news sources have higher writing quality (considering: misspelled words, punctuation and sentence length) compared to fake news articles which are likely to be written by unprofessional writers (Andrea et al., 2014). On the other hand, real news sources appear unbiased or contain neutral words, and describe events with facts. The development of a fake news classifier and comparing it to other classification methods is a difficult task (Fan, 2017). There are three commonly accepted features of fake news: the text of an article, the user response and the source; it is necessary to integrate all three in one place, and then they proposed a hybrid model. The first module captures the abstract temporal behavior of users, and measures response and text. The second component estimates the source for each user and then combines it with the first module (Ruchansky, Seo, & Liu, 2017).

### 2.3.3 Similar Application Areas

In this section, I discuss similar application areas that are related to the fake news detection problem.

- **Truth Discovery:** Truth discovery can be beneficial in several application domains, especially where we need to make critical decisions based on reliable information from multiple sources. A few examples of these areas are healthcare (Li et al., 2016), crowd sourcing (Tschitschek, Singla, Gomez Rodriguez, Merchant, & Krause, 2018b) and information extraction (Highet 1972).
- **Rumor Detection:** The goal of rumor detection is to classify a piece of information as rumor or non-rumor. The process of rumor detection is to collect and filter the posts that discuss specific rumors. These posts are considered important sensors for determining the accuracy of the rumor. Rumor detection can be further divided into four subtasks: stance classification, veracity classification, rumor tracking and rumor classification (Zubiaga et al., 2018).

- **Clickbait Detection:** Clickbait aims to grab the reader's attention and make them click on a particular link. Existing clickbait approaches use various extraction features from teaser messages, linked web pages and meta information of tweets (Potthast, Stein, & Hagen, 2016).
- **Email Spam Detection:** Spam detection in emails not only creates problems and brings financial loss to companies, but is also annoying to individual users. Different groups work with different approaches to detect spam in emails. According to the current state of the art different machine learning approaches are very helpful for spam filtering. Spam causes different problems which I broadly discuss above but more precisely spam causes misuse of traffic, computational power and storage space (Siponen & Stucke, 2006).

Similar application areas are closely related to fake news detection. Spam detection in emails and fake user detection on Twitter has become a research area in social networks (Masood et al., 2019). Ersahin et al., 2017 proposed a method to detect fake news similar to the spam account detection on Twitter by analyzing the user name, profile, content, description and the total number of sharing. Gupta et al. (2015) presented a method to detect spammers on twitter using classification techniques i.e., Naïve Bayes, Clustering and decision trees which could also be helpful in fake news detection. Zhang et al. (2020) proposed a method to check the level of clickbait headlines that attract users and the publisher who created the clickbait. Similarly, in a news story, fake content could be detected by the topic of the story and the producer who shares the story. Cao et al. (2020) suggested that just as in the truth discovery, fake text can also be helpful in detecting fake news. Tools and techniques used for these similar applications are discussed in Section 2.4.1. The identification of facts and non-facts is a related research area to my work with some differences. Looking at these areas may provide clues to interesting features and models to use in my work.

#### 2.3.4 Strengths and Weaknesses

In recent years automatic detection of fake news using classification techniques has gained popularity in academic communities as well as among the general public. However, existing approaches rely on machine learning algorithms with novel features to detect fake texts. One of the major limitations of these approaches is at an early stage of detection i.e., the required

information is unavailable or insufficient (Oshikawa et al., 2018). Linguistic features extracted from the text at an early stage are often insufficient, and when given to machine learning algorithms for prediction, the results may not be accurate. Existing approaches cannot be used to detect fake news when we have no text, only photos or videos. The prediction is based on the source i.e., the user who first shares the text regarding the relevant claim (Castillo et al., 2011a; Wu et al., 2014). Data-oriented applications in fake news detection when news is detected as fake, it is immediately blocked due to its functionality, we cannot replace it. If news is detected as fake, we need at least an expert opinion or verification before it can be blocked. Another reason is the speed of spreading of these types of information on social media networks is a challenging problem that requires attention and alternative solution. A combination of data and knowledge is urgently needed in the detection of fake news not only in this case but also in some other related problems.

## 2.4 Fact Checking

Fact checking can take as little as 15 to 30 minutes for a simple fact check; a full day for a more typical one; and two or more days for complicated fact checks (Hassan et al., 2015). The term fact is widely known and there are several definitions. For this work, I have considered the definition “A fact is something that has occurred or is correct”. In the context of news articles, events that have taken place and statements that claim to be true are factual; opinions and interpretations, on the other hand, are not. Manual fact checking nowadays is a disadvantage but automated fact checking can help to reduce the human burden. While end-to-end fact-checking solutions are not yet trusted to replace human fact checkers, automating fact-checking subtasks can assist human fact checkers and reduce the time required. Fact checking is often considered a multi-step process, including the extraction of check-worthy statements (Vlachos & Riedel, 2014). The fact-checking process starts with monitoring different media sources. From these sources, human fact checkers identify articles that contain relevant information. The detected check-worthy statements are normalized if necessary and then fact-checked. Finally, the results and verdicts from the fact-checking process are published to the general public. The fact-checking process can be seen in the below figure.



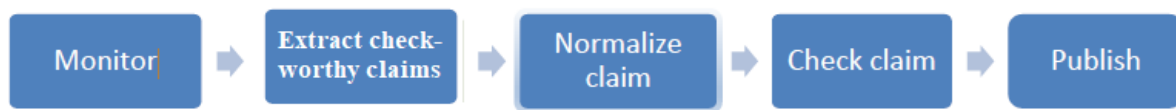


Figure 2.4: Fact-checking process

Creating an end-to-end fact-checking system is complex work, and these systems are not trustworthy without human intervention as the above literature shows. I decided to contribute to this particular task in the fact-checking process. Automation would assist humans and reduce their burden in fact checking. Not all factual statements are check-worthy, but only small subsets of them are check worthy. Most factual statements contain facts that are not important or not interesting for the general public to fact-check. A statement must meet three conditions to be considered check worthy: It should be factual not an opinion; interesting for the general public and should be possible to check. In the next section, I will discuss check-worthy claims as it is the subtask in fact checking.

### 2.4.1 Check-Worthy Claims

The literature on automating fact checking focuses on determining the veracity of a claim, while the phase of identifying check-worthy claims has received less attention. Hassan, Li and Tremayne (2015) considered the check-worthiness of a claim and in a follow-up work Hassan et al. (2017) presented an end-to-end fact-checking system called ClaimBuster. This system uses a supervised learning approach to tackle the identification of check-worthy claims. A dataset of 28,029 sentences was annotated by professors, journalists and students. The sentences were categorized into non-factual sentences, unimportant factual sentences, and check-worthy factual sentences. ClaimBuster assigned each sentence a score between 0 and 1. The higher the number, the more check worthy the sentence was. ClaimBuster used multiple categories of methods. Term frequency and inverse document frequency (TF-IDF) calculates a score for each word in a document by using an inverse ratio of the frequency of the word in a given document, and the percentage of documents in which the word occur (Ramos, Eden and Edu, 1999). Words with a higher TF-IDF score implied higher importance in a document. Additionally, ClaimBuster used part-of-speech (POS) tags, sentiment analysis, and word counting. A random forest classifier was used to avoid overfitting.

Gencheva et al. (2017) considered a different fact-checking organizations and then predict whether any or a particular fact-checking organization would select a sentence to check. Their dataset is publicly available and consists of four English political debates. A similar approach was used to find check-worthy claims; instead of creating their own annotations, they took existing annotations from fact-checking organizations to create the dataset. They then used additional features along with those used by ClaimBuster, including contextual features. To cover the context, they used features from the previous, current and next segment of the sentences. By segment they mean the number of consecutive sentences that a speaker says without interruption. Patwari et al. (2017) identified check-worthy statements in political debates; their dataset contains presidential debates and was annotated by consulting different fact-checking organizations. In difference from (Hassan et al., 2015) they divided statements into, check-worthy and non-check-worthy statements. As features, they extracted bag-of-words (BOS), which describe the occurrence of words within a document, Part-of-Speech (POS), which assigns grammatical categories to words, and named entity recognition. They also used POS tuples, taking into account that statements have a dependency structure (subject, verb, and object). They use a multi classifier system, assuming that these systems perform better than single classifier systems. (Jaradat, Gencheva, Cedeño, Márquez, & Nakov, 2018), an extension of the work of (Gencheva et al., 2017), developed Claim Rank, a working system for finding check-worthy claims. Taking a different approach from their earlier work, they instead focus on a single fact-checking organization and provide support for both Arabic and English.

#### 2.4.1.1 The Context in Identification of Check-Worthy Claims

Fact-checking organizations don't look at sentences in-depth; they just check them against what they know and then come to a. The same mindset is used when looking for check-worthy claims. However, in the pioneering work (Hassan et al., 2015) that first presented this task, no context is modeled, only sentence-level features are used to classify sentences. However, in their follow-up work (Hassan, Zhang, Arslan, Caraballo, & Jimenez, 2017) the end-to-end system includes an option that allows the user to make a decision when viewing a sentence. In contrast, Patwari et al. (2017) and Gencheva et al. (2017), also included context in these tasks; they extracted features from surrounding segments of the same speaker of the target sentence. Gencheva et al. (2017) also used discourse features and public reactions after a sentence, such as applause, laughter, or cross talk. It is difficult to narrow down satire in

academic literature. A method that can first translate the theories of humor, irony and satire into a predictive method for satire detection (Rubin, Conroy, Chen, & Cornwell, 2016b). The conceptual contributions of this work are to link satire, irony and humor. Then the fake news frames are selectively filtered based on their potential to mislead the reader. (Bajaj, 2017) proposed a new text classification approach that can predict whether the news is fake or not. The dataset used for this project was drawn from two different publicly available websites<sup>23,24</sup>; how fake news stories are shared on social media and other platforms and how to automatically identify the fake content presented by Janze and Risius (2017). Another method is to divide fake content into three categories: serious fabrication, large-scale hoaxes and humorous fakes. The authors provide a way to filter, vet and verify the news and discuss in detail the advantages and disadvantages of these news (Conroy, Rubin, & Chen, 2015a). Zhou, Cao, Jin, Xie, Su, Chu, et al. (2015) proposed a new mechanism because traditionally all rumor detection techniques are based on message level detection and analyze the credibility based on data but in real-time detection based on keywords then the system collects related microblogs using a data collection system that solves this problem. They proposed a model that combines user-based, propagation-based and content-based models and checks credibility in real time; the model then sends back the response within thirty five seconds.

Guha et al. (2017) proposed a new fact-checking mechanism that can help readers critically evaluate the news before making a judgment by performing a fact check. The goal of this work is not to provide readers with results that are fake or not, but to provide a mechanism for critically evaluating the news while reading it. They have introduced a fact-check corpus that can retrieve the runtime data of the article and compare it with the known facts. When the reader starts reading, the news fact-checking technique provides the reader with the opportunity to simultaneously read all related or linked stories to critically evaluate them. However, if the scoring measure falls below the threshold, the related fact-check is not displayed.

---

<sup>23</sup><http://www.kaggle.com>

<sup>24</sup><http://www.research.sianalmedia.co/newsir16/sianal->

### 2.4.2 Knowledge-Based Approaches

Some claims contain facts and finding these facts through text classification and comparing them with known facts to detect fake news is a difficult task (Hassan et al., 2015). Knowledge engineering could be helpful to create knowledge bases of the known facts which can play an important role in detecting fake content. Different groups introduced different models; some of them have been data-oriented while others have been knowledge-based only. The important point is the speed at which this kind of information spreads in social networks. It is a challenging problem that requires attention and an alternative solution. If the news is detected as fake, the existing techniques blocked it immediately based on its function as we cannot replace it; but if news is detected as fake, we need expert opinion or verification before blocking that particular news. This helps in bringing in third-party fact-checking organizations to solve the problem but this too is a time-consuming process. The existing fake news systems based on the predictive models simply classify whether the news is fake or not. Some models use source reliability and network structure so the major challenge in these cases is to train the model, which is impossible due to the unavailability of corpora. It is also possible to detect fake news with different known facts such as time, location, quality, and the stance of others. With these types of measurement similarities, we can detect the quality of news. Knowledge engineering helps to represent the knowledge of experts who are aware of this knowledge.

The goal of a knowledge-based approach is to use external sources to fact-check news content and the goal of fact-checking is to assign a truth value to a claim (Riedel, 2014). Many efforts have been made to develop some viable automated fact-checking systems. The details of knowledge-based approaches are discussed in Section 7.3.

### 2.4.3 Automated Approaches

Due to the diversity and the huge amount of data that keeps increasing, it is not possible to solve the problem of fake news in a manual or traditional. Pennycook & Rand, (2019) suggested Up-Rank algorithm content from a reliable media source that is suitable to automatically prevent the spread of misinformation on social media. One potential approach is for the social media platform to preferentially display content from news sources that users

rate as reliable. People across the political spectrum rated mainstream sources as far more reliable than either non-partisan or fake news sources.

Sterrett et al. (2019) proposed a method that can influence people's opinions about news on social media. It also tests the trust of the person sharing a story and tests the reliability of the news source reporting the story. They also suggested some valuable suggestions for researchers, citizens, and publishers on how to understand the evaluation and trustworthiness of news sources on social media and the possible impact of fake news. They emphasized that sharing the article rather than the source is the key factor in understanding the fake news dynamic. Zhou & Zafarani (2018) comprehensively and systematically review fake news research in terms of four perspectives. They discuss and summarize knowledge-based, style-based, propagation-based and credibility-based qualitative and quantitative analyzes of fake news. In addition detection and intervention strategies were also looked at. The review of false knowledge using (1) writing style (2) fake news characteristics (authenticity, intention) (3) various news related (e.g., headline, body text, creator, publisher), social related (e.g., comments, propagation paths, spreaders); (4) feature-based and relationship-based techniques to study fake news; and (5) available resources, e.g., fundamental theories, traditional websites, tools, and social platforms to support fake news studies.

On social networks, information occurs at such a rate that amplification of this false information can be the potential cause of a real-world crash. By providing web services, they benefit from their massive use in the long run (Figueira & Oliveira, 2017).

Human fact checking is quite good at finding the shortest path between concept nodes under semantic proximity metrics on knowledge graphs. Fact checking can effectively reduce simple network analysis problems that are computationally easy to solve but infeasible for humans. The result shows that the correct measurement of the truth content of statements depends on indirect, interconnected paths (Ciampaglia et al., 2015a).

Jaradat et al. (2018) proposed automatic identification and verification of political claims through check-worthy statements as another method to overcome the burden of human fact-checking. Task 1 focuses on predicting those claims that are included in a political debate or speech and should be prioritized for fact checking. Task 2 fact-checks and evaluates whether a politician's claim is factually true, half true, or false. Evaluation results showed that the

most successful fact-checking approaches used different neural networks (for task 1) and evidence retrieved from the web (for task 2).

Pennycook & Rand (2019) explains why people blatantly believe fake news headlines. They found that analytical thinking helps to detect fake news under standard experimental conditions. They also evidence of a relationship between analytical thinking and media truth, independent of the importance of nature and ability. Allcott, Gentzkow, & Yu (2019) studied the websites that spread fake news stories on Facebook and Twitter between January 2015 and July 2018. The data comes from BuzzSumo and is obtained directly from Facebook API and Twitter. According to the data, interactions with fake content on Facebook have declined sharply compared to Twitter, with a decreasing share of 60%.

Category	Site		
<b>Major News Sites</b>	cnn.com	nytimes.com	theguardian.com
	washingtonpost.com	foxnews.com	huffingtonpost.com
	usatoday.com	wsj.com	cnbc.com
	reuters.com	time.com	nypost.com
<b>Small News Sites</b>	usnews.com	cbsnews.com	chron.com
	asptimes.com	bakersfield.com	bendbulletin.com
	bnd.com	broadcastingcable.com	charlestoncitypaper.com
	chicagomaroon.com	collegian.psu.edu	columbian.com
	dailynebraskan.com	dailynexus.com	dailynorthwestern.com
	dailypress.com	dailyprogress.com	dailytexasonline.com
<b>Business and Culture Sites</b>	imdb.com	ign.com	rottentomatoes.com
	forbes.com	shutterstock.com	businessinsider.com
	webmd.com	psychologytoday.com	who.int
	9gag.com	jalopnik.com	timeout.com
	espn.com	cricbuzz.com	nba.com
<b>Fake News Sites</b>	dailywire.com	ijr.com	dailycaller.com
	occupydemocrats.com	express.co.uk	redstatewatcher.com
	thepoliticalinsider.com	thefederalistpapers.org	truthfeed.com
	bipartisanreport.com	rightwingnews.com	qpolitical.com
	madworldnews.com	yournewswire.com	uschronicle.com

Figure 2.5: Comparison sites and fake news sites (*Allcott et al., 2019*)

## 2.5 Combination Approaches

We know that machine learning helps to build artificial intelligence systems based on tacit knowledge because it can help us solve complex problems based on real-world data (Leonard et al., 1998). On the other hand, we know that knowledge engineering helps to represent the knowledge of experts who are aware of this knowledge. For this reason, I proposed that the

integration of text classification and fact checking of check-worthy statements can be helpful in detecting fake news. Karadzhov et al. (2017) presented a general purpose framework for fully automated fact checking using external sources, considering the entire web as a source of knowledge that can help to confirm or reject a claim. A deep neural network using LSTM text encoding methods has been shown to be important in achieving balanced predictions and better results. Text positions, reliability of sources, language style of articles, and sample worldviews. This model is much simpler than the information sources used. Overall, the robust performance of the model depends on two different fact-checking tasks corresponding to its generality and possible application formulations for fact checking. A fully automated end to end fact-checking system does not exist, but Hassan et al. (2017) proposed a fact-checking system that takes the claims as input from social media websites, debates, and other sources then after a few steps monitors, tracks, and matches with fact-checking reports, and finally checks keywords and then provides factuality.

Today's society has to contend with an unprecedented amount of falsehoods, hyperboles, and half-truths that are difficult to distinguish. One of the main sources is politicians and certain organizations that keep making these false claims. This model automatically translates the claims into questions against the knowledge base and reports whether they are verified; however, the main focus is only on political claims. Nguyen, A. et al. (2018) present the design and evaluation of a mixed-initiative approach to fact checking that combines human knowledge with information retrieval and ML approaches. They used predictive models to describe the user experience and directly automate predictions based on it. The classification is based on the item's position and verifying the truth of the claims. However, due to the predictive model, it is not possible to change a decision once it has been made. Adair, Stencel, Clabby & Li (2019) proposed a method for humans to use algorithms to increase productivity and improve the effectiveness of the algorithms, which requires a human contact to inform editors of possible political inaccuracies.

## 2.6 Interdisciplinary Approaches

The impact of news media on society involves other actors from different fields such as journalists, social scientists, and political scientists; they need to work together using media

platforms and artificial intelligence techniques to address the problem of fake news and find different ways to disinfect it.

### 2.6.1 Media Literacy

Media literacy is one of the important key points that can help in combating fake news. This type of lecture can be given at the school level or take advantage of social medial technique to deliver the message to a wide audience and train them to fight disinformation.

### 2.6.2 Critical Thinking for Citizens

Critical thinking, as an ideal from an educational point of view (Robert & Price, 1986), is thus desirable for society. But people who spread fake news without thinking may lose credibility. The reliability of fake news is better when people think thoughtfully, but deliberative thinking is more time consuming (Kahneman, 2013).

### 2.6.3 Empower Civil Society

News media and social media companies are the big beneficiaries of this type of fake content, as we have already discussed that fake content is a double-edged sword. These companies make a huge business from the heavy distribution of fake content but at the same time, these companies are one of the main causes of disinformation spreading in seconds without checks and balances. Anyone can create an account and start spreading such types of fake content on social media without any restriction. The big companies like Facebook and Twitter are now following the rules and regulations to detect fake content. For this purpose, they have introduced different types of verifications related to demographic data or other points to combat this problem. So, with the advantages of artificial intelligence techniques, civil society could play the role of an independent actor to tackle this problem.



## 2.7 Discussion

Fake news producers mislead readers so it is difficult to detect fake content in traditional ways. Social media platforms are multi-modal, have a large scale but most data is user-generated and sometimes anonymous users can create and share data without any checks and balances. There is less research that provides a systematic understanding of user profile characteristics useful for detecting fake news? For this reason, I discuss in my research questions and then propose the combination approach that combines both classification and fact checking. The literature review has shown that integrating text classification and fact checking of check-worthy statements is an important step toward fake news detection. I discuss different approaches that have been defined in recent years to address the problem of detecting fake news detections. Most of these approaches are based on supervised or unsupervised methods. Due to the unavailability of gold standard data set, these approaches have not yielded positive results, especially in training and testing classifiers. In most cases, fake datasets are tested rather than real datasets. However, there are different approaches that address this task and have achieved promising results; however one important part has not been sufficiently exploited, namely context. It has often been observed that classification and fact checking depend on context, and incorporating the combination of these two could be helpful in detecting fake news. This model could possibly improve the performance of classifiers. It is a fact that people's motivations and psychological state may be different from professionals in the real world. To address the problem of fake news detection, we need to incorporate both behavioral and social entities to combine classification and fact checking to distinguish fake content.

## 3 Research Method

This section provides information about the research design and methods used in this thesis. It starts with the design science research (Section 3.1), design science research process (Section 3.2) research approach (Section 3.3), proposed research methodology (Section 3.4), research guidelines and finally the research validity, reliability and trustworthiness of the research are described. The research design defines what specific methods and techniques will be used in the research.

### 3.1 Design Science Research

Design science is a research strategy with a goal to construct a new reality. It can be seen as an artifact creation. Such artifacts can be models, design theories and methods (Hevner & Chatterjee, 2010). With the help of these artifacts it is possible to generate knowledge. Design science research focuses on three inherent research cycles: Relevance, Design and Rigor (Hevner, 2007). These cycles contain different activities that are involved in the research project. The research is linked to the activities through a knowledge base of scientific foundation, expertise and experience. This process contains the environment, design science research and knowledge base where design science is in the middle; relevance and rigor cycles are covered with different activities. Figure 3.6 shows the connected activities involved in this process.

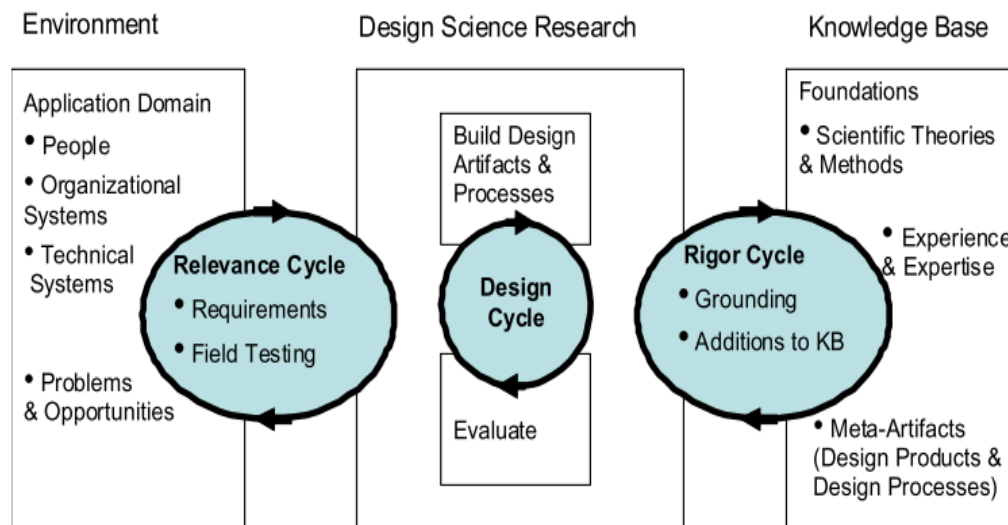


Figure 3.6: Design science research cycles (adapted from Hevner (2007))

To understand the Design Science Research Process Offerman (2009) has given a detailed comparison between other processes and his proposed process in Table 3.1. The table includes problem identification, design solution and evaluation phases. Peffers et al. (2007) include problem identification, design, development, demonstration and evaluation. Takeda et al. (1990) proposed an enumeration of problem, suggestion, demonstration, evaluation and decision phases. Nunamaker et al. (1991) proposed framework construction, system architecture and evaluation of the system. March et al. (1995) present the processes of design sciences. Vaishnavi and Kuechler (2004) proposed the phases of problem awareness, suggestion, development, evaluation and conclusion. Finally, after understanding the other process, Offerman (2009) revised and proposed problem identification, literature research, expert interviews, design artifacts, literature research, refine hypothesis, expert survey, laboratory experiments and summarize results. The results can be seen in Table 3.1.

Table 3.1: Comparison of design science research processes (Offerman 2009)

	Peffer et al. 2007	Takeda et al. 1990	Nunamaker et al. 1991	March and Smith 1995	Vaishnavi and Keuchler 2004/5	Offerman, P 2009
<b>Problem identification</b>	<ul style="list-style-type: none"> <li>• Problem identification and motivation</li> <li>• Define the objectives for a solution</li> </ul>	<ul style="list-style-type: none"> <li>• Enumeration of problems</li> </ul>	<ul style="list-style-type: none"> <li>• Construct a Conceptual Framework</li> </ul>		<ul style="list-style-type: none"> <li>• Awareness of Problem</li> </ul>	<ul style="list-style-type: none"> <li>• Identify problem</li> <li>• Literature research</li> <li>• Expert interviews</li> <li>• Pre-evaluate relevance</li> </ul>
<b>Solution design</b>	<ul style="list-style-type: none"> <li>• Design and development</li> </ul>	<ul style="list-style-type: none"> <li>• Suggestion</li> <li>• Development</li> </ul>	<ul style="list-style-type: none"> <li>• Develop a System Architecture</li> <li>• Analyze &amp; Design the System</li> <li>• Build the System</li> </ul>	<ul style="list-style-type: none"> <li>• Build</li> </ul>	<ul style="list-style-type: none"> <li>• Suggestion</li> <li>• Development</li> </ul>	<ul style="list-style-type: none"> <li>• Design artefact</li> <li>• Literature research</li> </ul>
<b>Evaluation</b>	<ul style="list-style-type: none"> <li>• Communication</li> <li>• Evaluation</li> </ul>	<ul style="list-style-type: none"> <li>• Evaluation to confirm the solution</li> <li>• Decision on a solution to be adopted</li> </ul>	<ul style="list-style-type: none"> <li>• Observe &amp; Evaluate the System</li> </ul>	<ul style="list-style-type: none"> <li>• Evaluate</li> </ul>	<ul style="list-style-type: none"> <li>• Evaluation</li> <li>• Conclusion</li> </ul>	<ul style="list-style-type: none"> <li>• Refine hypothesis</li> <li>• Expert survey</li> <li>• Laboratory experiment</li> <li>• Case study / action research</li> <li>• Summarise results</li> </ul>

## 3.2 Research Approach

Saunders et al. (2007) distinguish two research approaches: inductive and deductive, and explain the steps involved in these approaches (Figure 3.7).

**Inductive approach:** Starts with an observation and with an objective containing different patterns. Data collection is carried out and once the data is collected, the analysis phase begins. This is followed by a hypothesis and then a theory is formed.

**Deductive approach:** This approach starts with the hypothesis followed by conducting observations and then examining the results. If a change is needed during these phases, it can be made.

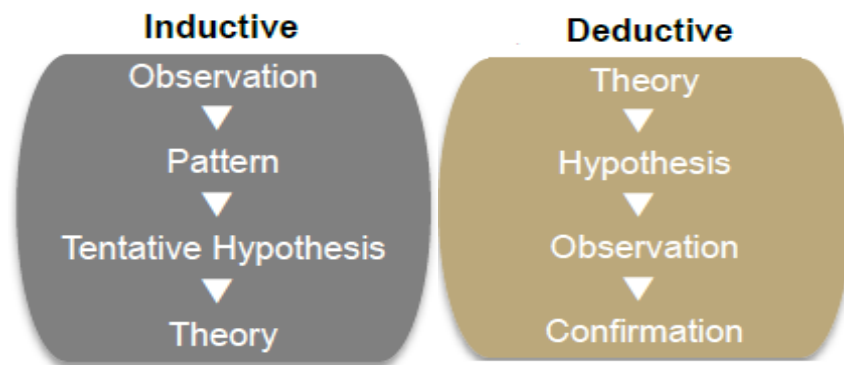


Figure 3.7: Induction and deduction (adapted from Trochim 2006)

Following a design-oriented research strategy, primarily an inductive research approach is used. For evaluation, a formal (using mathematics) or semi-formal (e.g. conceptual) deduction would be the ideal situation (Österle et al., 2010). However, in design science research it is rare that an artifact can be formally evaluated (Österle et al., 2010). It is more likely that the inference of a single case study is an example of an inductive approach within design-science research (Österle et al., 2010). I followed the inductive approach by starting from the fake news phenomenon and the impact on different domains, especially activities of daily living, and considering existing theories on fake news and fact checking.

### 3.3 Research Design

In my research, I followed the methodology of design science research (DSR) presented by Vaishnavi and Kuechler (2004) and extended with elements from Peffers et al. (2007) as shown in Figure 3.8.

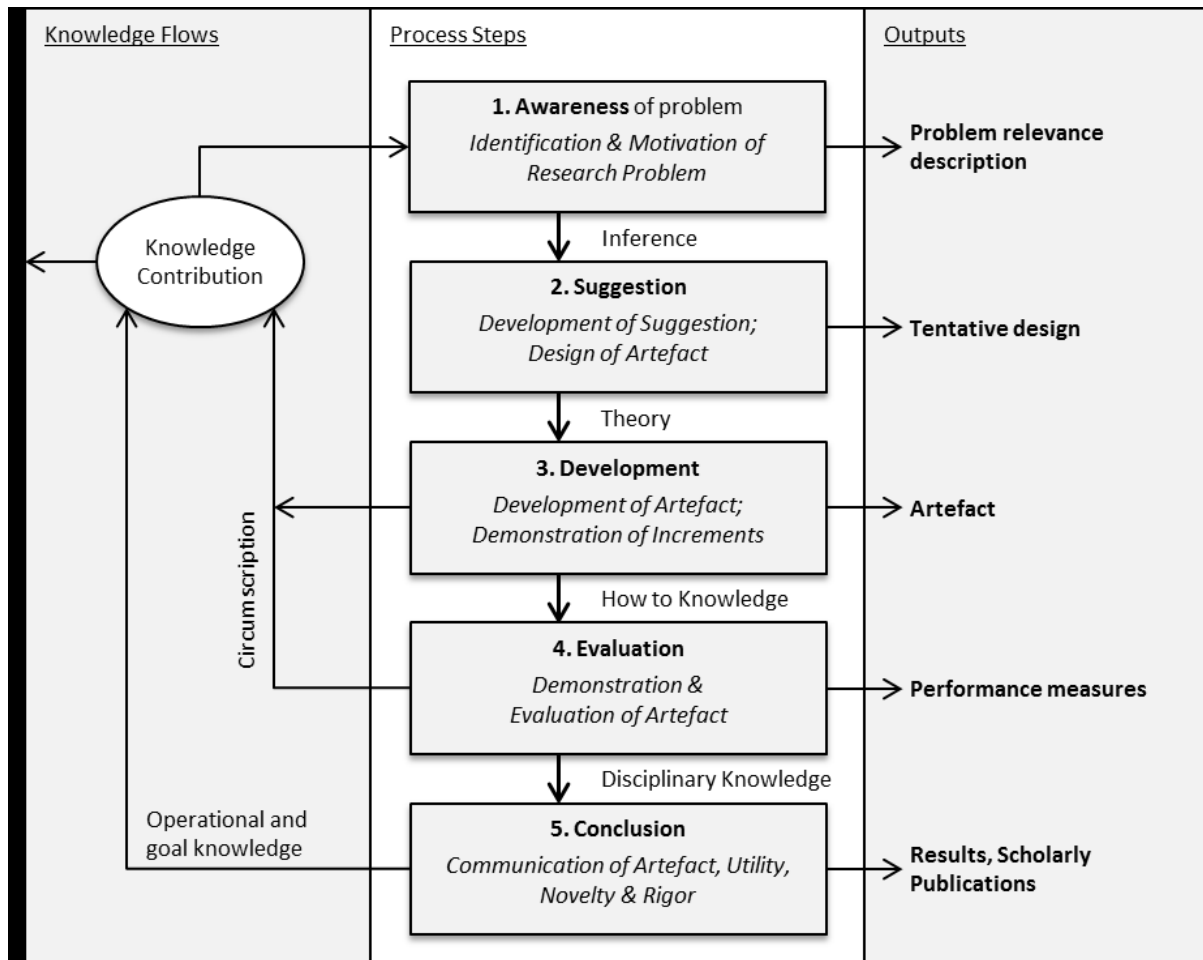


Figure 3.8: General methodology of design science research (adapted from Vaishnavi and Kuechler (2004) and enhanced with elements from Peffers et al. (2008))

Vaishnavi and Kuechler (2004) explain five steps that are included in my research methodology (Figure 3.9). These steps are awareness of the problem, a suggestion for a solution, the development of the artifact, the evaluation of the artifact and a conclusion to apply which includes novelty and rigor of the artifact. In my research I have developed three methods, text classification, identification of check-worthy statements and fact checking. For each method I did problem awareness based on literature and suggestions based on experiments. Finally, I combined them in the development phase. The complete process can be seen in Figure 3.9. In the next section, all these steps are described in detail.

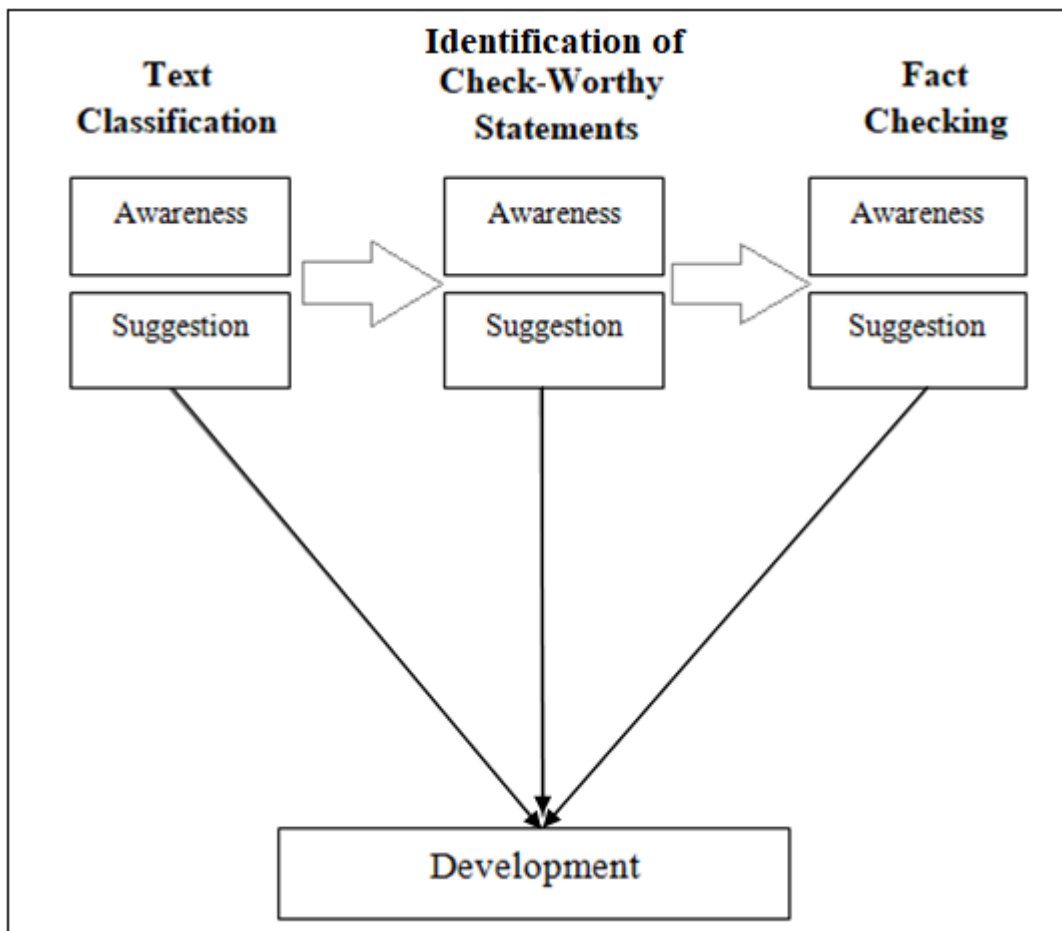


Figure 3.9: Proposed research methodology

### 3.3.1 Awareness of the Problem

The research begins with a clearly articulated problem, that can be solved by an artifact proposed by Vaishnavi and Kuechler (2004) and Ellis and Levy (2010). Fake news detection is a complex task and although there are different initiatives to create a system that can detect fake news and attempt to automate the problem, they are still not trusted when it comes to detecting fake news and to replacing human-fact checkers in terms of verifying the news. As can be seen from the literature review, fake news detection is a classification task that focuses on whether the news is true or not. However, one problem is that the classification in fake news detection is not completely accurate because the classification methods are not specialized for detecting fake news, because in a news item, only a part may be fake while the rest is not. There is a need for an alternative approach that combines classification and fact-checking to better detect fake news. This would reduce the time required, as existing

fact-checking methods are very time consuming, and would also be more accurate. This part of the problem awareness is described in Chapter 4.

For each phase of the proposed solutions, the problem is analyzed in more details doing separate literature reviews, which are described at the beginning of Chapters 5 to 7.

### 3.3.2 Suggestion

Based on the problem awareness, an overall process is suggested which consists of three phases: classification, identification of check-worthy statements and fact checking of these statements. For each phase, experiments have been conducted to suggest appropriate methods. These experiments are described in the Chapters 5 to 7 after the literature reviews for the problem awareness.

Fake news detection can be considered as a supervised text classification task. The goal is to investigate which features can be extracted from the sentence and surrounding sentences to obtain the information about the sentence and present the context. It is also necessary to investigate which supervised machine learning methods are best suited for classification in this task. For classification, I collected data from publicly available sites such as Kaggle<sup>25</sup> and Signal Media<sup>26</sup> (see Chapter 5). In the first dataset, I had 18000 news articles collected from different news organizations. These articles were then sorted by binary labels fake, non fake and unclear. The second dataset contained 5000 articles collected mainly from mainstream news sources. For identifying check-worthy statements (see Chapter 6), data was collected by consulting different fact-checking organizations. These provided online transcripts of speeches and debates, which included the fact-checked statements. From each of these transcripts, each sentence was annotated. A sentence is considered check -if it has been checked by at least one of the fact-checking organizations as to whether it is true or not. Most of the available data came from the political domain and focused on the 2016 U.S. presidential election. After obtaining the dataset, I visualized the data to better assess the correlation between different features of a sentence and the class they belonged to; whether they were check worthy or not. Feature extraction was the next step. Each sentence was represented as a vector of features; these are not only sentence-level features but are also extracted from surrounding sentences to include context.

---

<sup>25</sup> <https://www.kaggle.com/>

<sup>26</sup> <https://research.signal-ai.com/newsir16/signal-dataset.html>



To get a better idea of which classification models are best suited for classification modules and which are best suited for fact checking, I conducted experiments with different classification models used in related text classification and fact-checking tasks. Some of them were not suitable for this work and were discarded; however, there were three classification models - Passive-Aggressive (PA), Naïve Bayes (NB) and Support Vector Machine (SVM) - that gave promising results (see Figure 3.10).

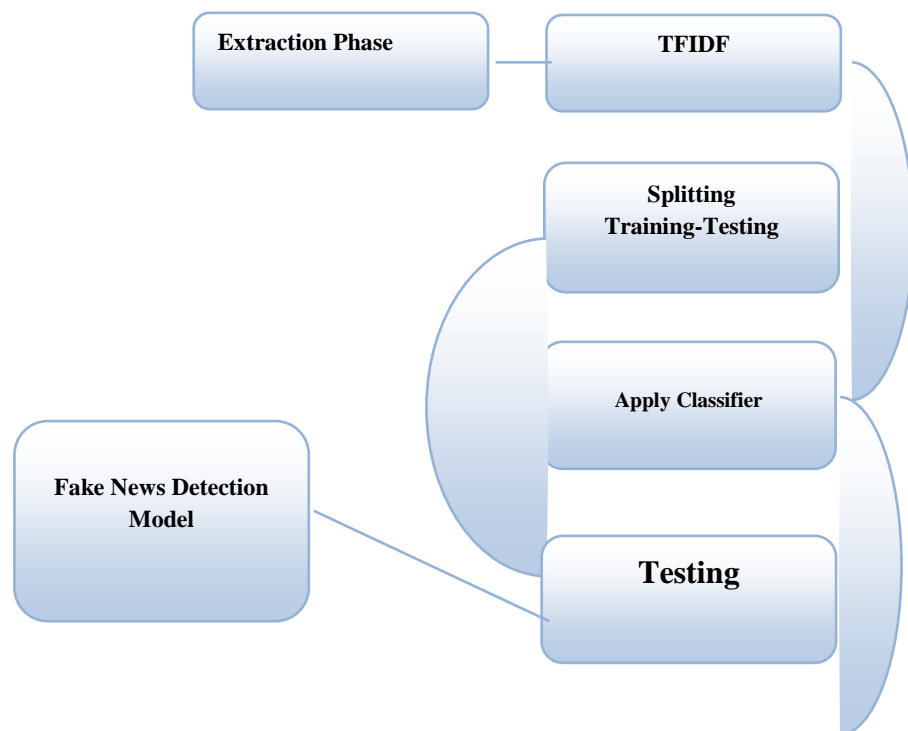


Figure 3.10: Development of classification model steps

For the fact-checking module (see Chapter 7), I continued to work on Support Vector Machines (SVM) (Joachims, 1998) and Feed Forward Neural Network (FNN) (Bengio et al., 2003).



Figure 3.11: Fact checking model steps

### 3.3.3 Design and Development

Finally, a prototype was implemented that combines the three phases into a single system. The classification models Passive Aggressive (PA), Support Vector Machine (SVM) and Naïve Bayes (NB) were used to classify sentences by feeding these models with extracted features. On the other hand, Support Vector Machine (SVM) and Feed Forward Neural Network (FFN) were used for identifying a check-worthy statement which is the subtask is in fact checking that reduces the burden on human fact checkers. After determining that the models worked as expected, I evaluated the models using 4-fold cross-validation (Kohavi, 1995), a method that ensured that despite the small data set, each of the speeches and debates were used for both testing and training. This approach helped to see how the models behaved in different sets of new data, rather than just using a fixed training and testing set.

### 3.3.4 Evaluation

The models were evaluated using different performance metrics (Accuracy, Precision and Recall) to better understand their behavior. The classification module achieved accuracy up to 93% which is highest. The evaluation results are discussed in Chapter 5. On the other side check-worthy statements were evaluated through the performance metrics which is highlighted and presented in Section 6.7.1. In some cases, it was necessary to iterate the design of the classification model and the fact-checking model to assess which features performed best and to adjust their parameters depending on the performance metrics. Finally, the combination part for the above two modules evaluated through an automated tool which I developed. The results are presented in Chapter 8. The developed system compares the statements and predicts the fact of the news and shows the aggregation of fake and non-fake news.

### 3.3.5 Conclusion

The conclusion of the findings of this thesis was carried out by following the structure of the research process step by step. At the end, as part of the communication, a conclusion is derived for the whole research. Communicating the results and conclusions of the research is part of contributing to the body of knowledge without documenting and properly

communicating the results (Hevner & Chatterjee, 2010). Throughout the work, the problem, the modeling of the task, the reasoning behind it and the experiments were documented and communicated.

### 3.4 Discussion

Research addresses an acknowledged problem, builds upon existing literature, and makes an original contribution to the body of knowledge. All these points are valid for my research, the chosen task is a well-known problem, all the research was based in existing literature, methods and the modeling of the task is an original contribution in this area. My research started with fake news detection through classification task as fake news detection is pure classification problem as per the literature. After classification it is important to identify the problem that comes from the huge amount of time that takes human-fact checkers to check a claim. To reduce some time, the first task of fact-checking, identification of check-worthy claims can be automated. I modeled it as a binary text classification task, that goes beyond sentence-level approach that was seen before in previous works (Hassan, Li and Tremayne, 2015), by creating a window around the targeted sentence, thus including context. Literature should be used for supporting the research, to find the problem, emphasize its impact and importance, and identify its cause (Ellis and Levy, 2008). A problem statement should outline the problem the study addresses and should argue its validity. A careful literature review was done for this work. Indeed the review was very useful in identifying a gap in previous related works and to have an overview of the approaches and methods used in related tasks. One main contribution of the thesis is creating an approach that considered context around a sentence when finding check worthy claims. Additionally, the characteristics below distinguish research-level development from product development as stated in the corresponding literature:

- Systematic documentation of the process and the design choices, different options considered during the process, and the rationale for the selected solution (Akker, 1999).
- Use of well-established research methods throughout the process (Hevner *et al.*, 2004).
- Empirical testing of the developed artifact (Hevner *et al.*, 2004).
- Communication of results (Hevner *et al.*, 2004).

The next chapter will answer research question 1.

## 4 Problem of Detecting Fake News and Overview of the Solution Approach

The purpose of this chapter is to provide an introduction into the problem. It will answer the first research question:

*RQ 1: What is the problem of detecting fake news?*

I present news from American politics as an application scenario to demonstrate the research problem. Then I show the drawbacks of existing solutions for fake news detection. From this I determine a solution approach consisting of three components, each of which is determined by a research question. At the end the chapter I present the overall solution architecture.

### 4.1 Problem

As social media has played an increasingly large role in our lives, it has allowed for the rapid and viral spread of ideas and opinions. This has generally helped users to be aware of what is happening locally and globally. The facilitation of such rapid dissemination of content has also led to the spread of misinformation. In 2016, the term “fake news” first came to the attention of general public. The trend of fake news was first noticed before the United States presidential election, when 140 fake news sites were identified that had attracted a lot of traffic from the social media platform Facebook<sup>27</sup>. The term, popularized by American President, Donald Trump, was initially used to refer to the coordinated spread of misinformation; predominantly via social media. The term was widely misused with some using the term to criticize individuals and news organizations with whom they disagreed. This has led to the creation of more formal, academic definitions to describe the different variants of fake news (Tandoc et al., 2018). For this report, I have used the definition of Collins Dictionary for the term “fake news”:

<sup>27</sup><https://www.buzzfeednews.com/article/craigsilverman/how-macedonia-became-a-global-hub-for-pro-trump-misinfo>

“noun: false, often sensational, information disseminated under the guise of news reporting” Since the U.S. presidential election, there have been many allegations that false and/or misleading social media content has been used to influence elections and referendums around the world. In March 2018, a company called Cambridge Analytica hit the headlines when a joint investigation into the company by Channel 4 and the Guardian found that it used fake news as a means of spreading misinformation in several elections. Cambridge Analytica<sup>28</sup> alone is alleged to have influenced elections in India, Kenya, Malta, Mexico, the United Kingdom and the United States of America. This is a major problem for democracy as a democratic country depends on people being informed about the workings of the political authorities. The following is a world map showing trust in platforms (Figure 4.12).

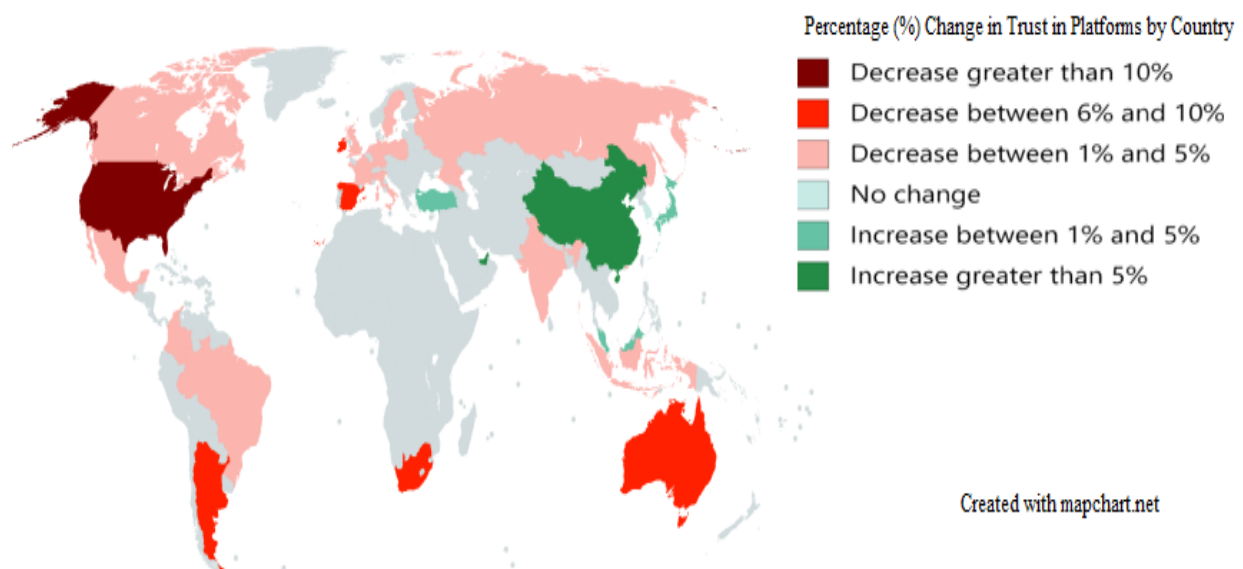


Figure 4.12: World map displaying trust in Platforms

The proliferation of misinformation on social and news media has created a demand for solutions that can accurately distinguish between genuine content and misinformation. In 2020 Google<sup>29</sup> has started investing in fact-checking and plans to invest \$6.5 million to implement fact-checking and misinformation tools. Twitter is also investing large sums of money to stop fake content as Twitter is used as a news source by many people (Pear Analytics, 2009; Naaman et al., 2010). Kwak et al., 2010 show that majority of the trending topics on Twitter are usually related to a news story. A few months ago, Twitter flagged two

<sup>28</sup><https://www.theguardian.com/uk-news/2018/mar/22/cambridge-analytica-scandal-the-biggest-revelations-so-far>

<sup>29</sup><https://www.axios.com/google-6-million-fact-checkers-misinformation-coronavirus-94a57d0f-fcbe-46c0-88f7-000ff05a3777.html>

fact-checking tweets that came from the President of the United States, Donald Trump. After fact-checking the tweets, Twitter determined that Trump falsely claimed that mail-in ballots would lead to a rigged election.<sup>30</sup> This has led to several efforts to build machine learning models across over different modalities, including text and integrate them on the other side into fact-checking applications as highlighted in Figure 4.22. Fake news detection is a relevant research problem as highlighted by the research community as it is one of the major challenges nowadays which is increasing day by day. American fact-checkers, such as PolitiFact<sup>31</sup>, typically take a claim from a political speech or opinion article and ask academic experts to rank it. The result is summarized in a ranked list. While “True” and “False” are options, claims are often ranked as “Mostly True” or “Mostly False” and, occasionally “Pants on Fire” (Moran 2018). PolitiFact, an American fact-checker, usually considers claims from political speeches or opinion pieces, which are then rated by academic experts. The result of the same is summarized with specific rankings. The rankings are categorized as follows – Mostly True, Mostly False, Pants on Fire. In this case there are only two options- “True” and “False”. However, the third ranking category “Pants on Fire” are used occasionally (Graves 2016b). The exact rating often depends on the analysis of intent. In certain cases even if a claim is technically true, it may still be presented in a misleading manner or according to an unconventional measure of economic growth. This method of fact-checking relies on broader institutions of liberal Western democracy that are not fully developed in Ukraine: journalists criticized a particular politician’s claim, within the framework of an ostensibly political opinion. StopFake<sup>32</sup> despite adopting the identification of fact-checking performs something different from others. This highlights the difference between the claims assessed by PolitiFact and those evaluated by StopFake. American fact-checking was designed to keep politicians honest, not to counter the systematic and coordinated work of a state sponsored propaganda machine. PolitiFact focus is specifically on political claims, but journalists are assumed to report accurate and honest news. Unlike PolitiFact, StopFake evaluates the work of journalists and looks for misleading stories based on fake evidence. Volunteers emphasize that they work only with “facts”, paying no attention to opinions. This approach contrasts with the PolitiFact approach of seeking expert opinion. In the next section, I have highlighted the problem and possible solutions as desired.

---

<sup>30</sup><https://www.cNBC.com/2020/05/28/twitter-ceo-stands-by-fact-check-on-trumps-tweets.html>

<sup>31</sup><https://www.politifact.com/>

<sup>32</sup><https://www.stopfake.org/en/main/>

## 4.2 Solution Approach

Fake news detection is considered a challenging task (Hassan et al., 2015) that requires multidisciplinary efforts (Lazer et al., 2018a). It requires skills about Natural Language Processing and knowledge about the domain of discourse. Therefore, false information is spread faster, deeper and wider than the truth (Pavleska, Školkey, Zankova, Ribeiro, 2018). Fake news detection performed by humans is a time consuming process (Oshikawa et al., 2018). Existing approaches mainly focus on extracting various features from text and then incorporating these features into classification models e.g. Decision tree, SVM, logistic regression, K nearest neighbour (Nidhi & Gupta, 2011). Emergent<sup>33</sup> is a real-time data-driven approach for rumor detection. It works automatically to track rumours that are associated with social media but where human input requires has not been automated. The problem is that most classification approaches are supervised so we need a prior dataset to train our model but as I have discussed earlier, obtaining a reliable fake news dataset is a very time consuming process.

When we look at fake news from a knowledge perspective, the main thing that comes to mind is fact-checking, which was originally introduced in journalism. The goal was to compare news with known facts or true knowledge. Besides being prolonged and expensive, the solution also requires journalists to check claims with evidence based on previously spoken or written facts. An example of this is PolitiFact which considers reports from three different editors to verify the authenticity of the news. As the Internet community and the speed of the information dissemination are growing rapidly, automatic fake news detection on the Internet has gained interest in the research community of Artificial Intelligence. The motto of automatic fake news detection is to limit human intervention and prevent the spread of fake news. The task of fake news detection has been studied from different perspectives with the development in subfields of computer science, such as machine learning (ML). Fake news detection is a binary classification task that determines whether a news story is fake or not (Bajaj, 2017). News is sometimes presented as a mixture of stories from different a source which makes it difficult to categorize whether it is real or fake. To solve this problem, adding additional classes is a common practice. Mainly, a category for the news, which is neither completely real nor completely fake, or, more than two degrees of truth is set as additional

---

<sup>33</sup> <http://www.emergent.info/>

classes. When these datasets are used, the expected outputs are multi-class labels, and these labels are learned as independent labels with assumptions (Rashkin et al., 2017). While sufficiently labelled data is one of the conditions for fake news classifiers to perform well, obtaining reliable labels requires a lot of time and work. Therefore, semi/weakly-supervised and unsupervised methods are proposed (Rubin & Vashchilko, 2010). Classification is not entirely accurate in fake news detection (Liu et al., 2017) because classification methods are not specialized for fake news detection. Fake news contains information that may be false or inaccurate (Zannettou, Sirivianos, Blackburn, & Kourtellis, 2019), and separating false text from real text is a challenging and difficult task (Lazer et al., 2018b).

### 4.3 Approach and Objectives

Through an analysis of the fake news detection problem described in the previous sections and based on the application scenario, generalized goals for the proposed approach are derived.

#### 4.3.1 Text Classification

Text classification is mainly about extracting various features of the text which are then used in the classification. The open nature of the web and social media, in addition to recent advances in computer technology, simplifies the process of expressing oneself bluntly with sheer pessimism. While it is easier to understand and track the intent and impact of fake news, the intent and impact of creating propaganda through the spread of fake news cannot be easily measured or understood. The proposed diagram for the classification module is shown in Figure 4.13.



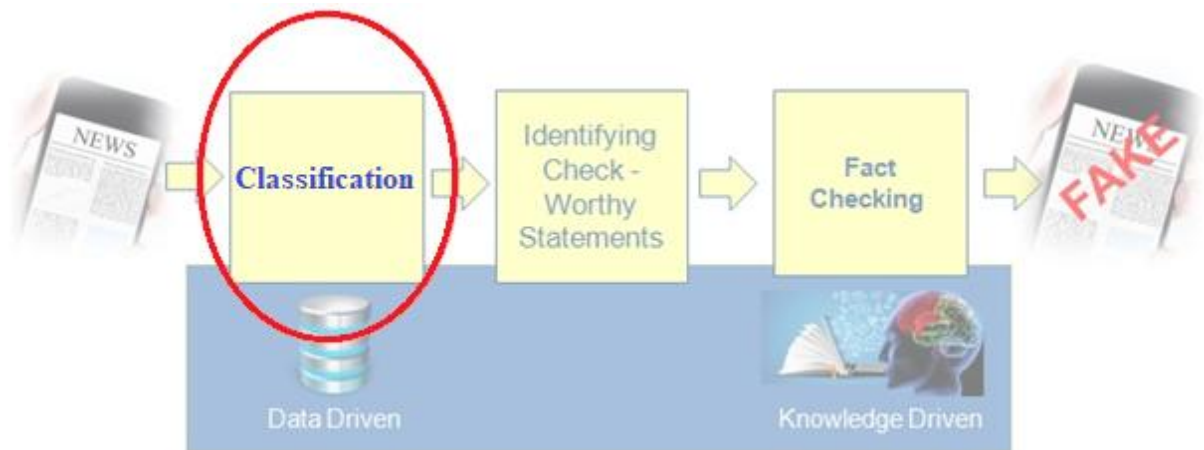


Figure 4.13: Text classification module (Data Driven) as part of overall procedure

Based on the problem relevance discussed above, the following goals were defined for the classification module:

- Focusing on the different features of text extraction and incorporating these features to detect fake news.
- Development of a fake news model using machine learning and natural language processing techniques.
- Train a classification model for fake news detection after several experiments.

These goals lead to the following research question:

***RQ2: What methods can be used to identify potential fake news?***

#### 4.3.2 Fact-Checking through Check-Worthy Statements

I have already discussed that the classification is not completely correct in cases where we have a mixture of news e.g. part of the news is true and the other part is fake. In this respect, it is desirable to be able to automatically distinguish between true and fake news with a high accuracy in new ways. Separating fake text from real text is a complicated process (Tambuscio, Ruffo, Flammini, & Menczer, n.d.) but fact-checking can help us in these situations because in traditional simple classification it is not possible to check the verdict of the news (Graves & Cherubini, 2016). Fact-checking is the task of evaluating the truthfulness of claims made in a non-fictional text to determine their accuracy (Riedel, 2014). Fact-

checking can be achieved by comparing individual statements of the news with known facts. Therefore, a prerequisite for fact-checking is to identify these check-worthy statements. Another work related to the detection of fake news is to use both news content and social contextual features surrounding the news content such as the news' diffusion patterns (Vedova, Tacchini, Moret, Ballarin, & Dipierro, 2018). The approach illuminated the study and effectiveness of demonstration of the chatbot solution Facebook Messenger. Multiple datasets were used to validate and implement the comparison to avoid fake content. To obtain news content from a web-page, Vedova et al. (2018) used HTML pages, which were then stemmed and represented as a vector-based on term frequency-inverse document frequency (TF-IDF). The classification was performed using a logistic regression algorithm and achieved an accuracy of 81.7%. This high accuracy results from considering both the natural language content of a post and its surrounding social context. Traditional methods that have only studied the effectiveness of NLP using neural networks, have often neglected the social context in the surrounding content. It is interesting to understand how other machine learning models could be applied to social context signals, and what impact they would have on accuracy (Vedova et al., 2018).

It is difficult to distinguish fake content from real content because we need to verify the facts of the news that can assess the veracity of claims. In general fake news detection focuses on news events, while fact-checking always remains an act of in-depth data analysis (Thorne and Vlachos 2018). For comparing the statements that are potentially fake, we need to create an assistant or access the knowledge base that contains all the necessary facts so that we can compare check-worthy statements with known facts. In order to understand how to assess the credibility of information; it is important to conduct studies and research before making decision. We have more information available to use than ever before (IBM, 2012) and the amount of information makes it even more difficult to determine what is trustworthy. As the terms "post-truth", "fake news", and alternative facts become more prevalent in social discourse and in the public sphere; we must develop the ability to critically evaluate the information we receive. Using credible sources to support an argument in research lends credibility to the writing. High-quality sources that support arguments are more likely to produce better results on assignments. Conversely, poor quality references will be noticed and are likely to have a negative impact on the results. Fact-checking is easy to use and produces highly accurate results; however it is a costly and time-consuming process that sometimes requires sifting through large amounts of online information. For this reason,

before addressing fact-checking we need to focus on check-worthy statements which is the subtask of fact-checking (see Figure 4.14). With this task we can reduce the burden of fact-checkers by focusing on the statements which can be potentially falsified.

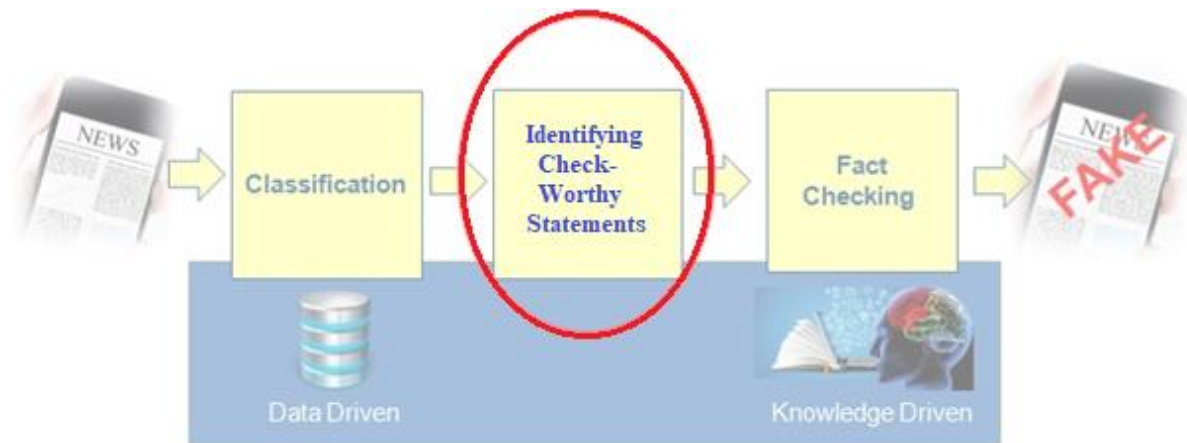


Figure 4.14: Identifying check-worthy statements as part of the overall procedure

To reduce the time and burden on fact-checkers, the following goals have been defined for identifying check-worthy statements:

- Automatically classifying statements into check-worthy and not check-worthy.
- Reviewing the context around a statement to better identify check-worthy statements.
- Determine which context features are useful to identify the check worthiness of a statement.

These goals lead to the following research question:

***RQ 3: How can check-worthy statements for fact checking be automatically identified?***

### 4.3.3 Automated Fact-checking

There is a need for an alternative approach that combines knowledge with data and requires automation of fact-checking that looks deeply at the content of the news with expert opinion in the same place to detect the fake news. While in classification and check-worthy statement identification the focus was document level, fact-checking reviews individual sentences.

Most of the existing fact-checking organizations based on predefined criteria. A comprehensive list of fact-checking websites is provided by Duke Reports<sup>34</sup>, where two hundred and ninety fact-checking websites across countries and languages have been available so far. The following table lists the first ten well known fact-checking websites and how they operate.

Table 4.2: Fact-checking websites comparison

Name	Topic	Content	Labels
Snopes <sup>35</sup>	Political and Social issues	-News Articles -Videos	True, False, Mixture, Unproven, Outdated, Scam, Mostly True, Half True
FactCheck <sup>36</sup>	American Politics	-Debates -Speeches -Interview -TV ads	True, False, No evidence
PolitiFact <sup>37</sup>	American Politics	-Statements	True, Mostly True, Half True, False, Mostly False, Pants on fire
The Washington Post <sup>38</sup>	American Politics	-Statements -Claims	One Pinocchio, Two Pinocchio, Three Pinocchio, Four Pinocchio, Verdict Pending
FullFact <sup>39</sup>	Economy, health and education	-Articles	Not Clear
TruthOrFiction <sup>40</sup>	Politics, Religion, nature, food, medical	-Email Rumors	Truth, Fiction
HoaxSlayer <sup>41</sup>	Not specific	- Articles -Messages	Hoaxes, scams, malware, fake news, true, humor, spams
RealClearPolitics <sup>42</sup>	Politics Defense Energy Heath	-News	Not specify
Our.news <sup>43</sup>	Politics	Articles- -News	Accepts, Rejected, Left Spin, No Spin, etc.
Media Bias <sup>44</sup>	Politics Media	-News	Bias, Least Biased, Right, Right Center

<sup>34</sup><https://reporterslab.org/fact-checking/>

<sup>35</sup><https://www.snopes.com/>

<sup>36</sup><https://www.factcheck.org/>

<sup>37</sup><http://www.politifact.com/>

<sup>38</sup><https://www.washingtonpost.com/news/fact-checker>

<sup>39</sup><https://fullfact.org/>

<sup>40</sup><https://www.truthorfiction.com/>

<sup>41</sup><http://hoax-slayer.com/>

<sup>42</sup><https://www.realclearpolitics.com/>

<sup>43</sup><https://our.news/>

<sup>44</sup><https://mediabiasfactcheck.com/>

The criteria they follow is to check whether the news is fake or not by forwarding it to the domain experts, individuals or organizations on that particular topic. Fact-checking is the third part of the overall process as shown in Figure 4.15.

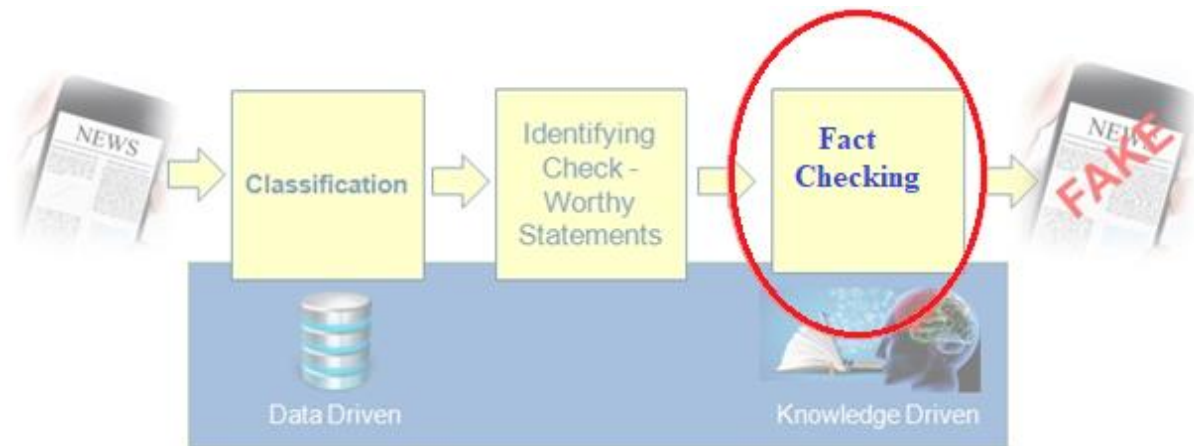


Figure 4.15: Fact-checking (Knowledge Driven) is the final part of overall procedure

The following objectives were defined for automated fact-checking:

- Review existing machine learning techniques for detecting fake news.
- Review existing knowledge driven techniques for fact-checking.
- Propose an approach that combines data with the knowledge to check the fact of the news.
- Develop an application for fake news detection.

These goals lead to the following research question:

***RQ 4: How can it be checked whether a statement is fact or fake?***

## 4.4 Solution Architecture

As mentioned earlier there is as yet no universally accepted definition of fake news, it is highly debatable in both practice and research. The proposed approach is based on relevant problems and application scenarios, as described in Section 4.1. Figure 4.16 shows the overall architecture of the proposed solution. It consists of two components, each having two

subcomponents. The sub-components correspond to the phases and research questions described in the previous sections while text classification and the identification of check-worthy statements are data-driven, fact-checking is based on knowledge.

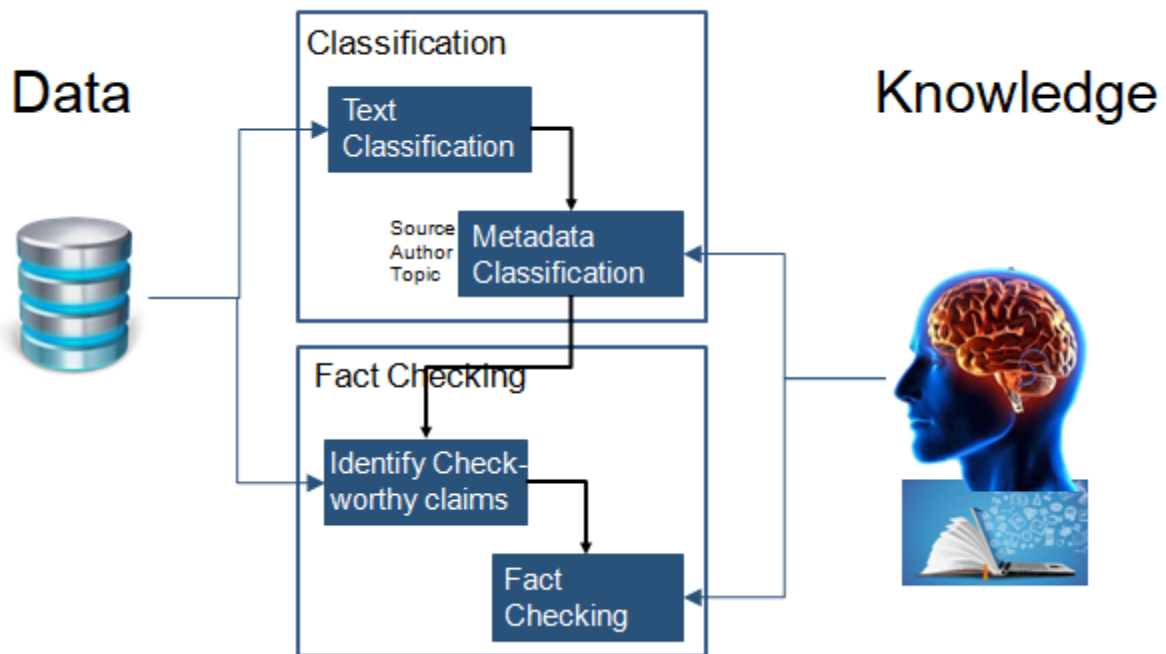


Figure 4.16: Proposed architecture for fake news detection

## 4.5 Discussion and Structure of the Research

In this chapter the research question-What is the problem of detecting fake news is answered and more detailed research goals and three additional research questions were derived. The research questions correspond to the three phases of fake news detections. These questions are answered in the next chapters.

Chapter 5 deals with classification to distinguish fake and non-fake news. Chapter 6 is about identification of check-worthy statements and chapter 7 describes the fact-checking solutions. Each chapter starts with a literature review. In contrast to the literature review in Chapter 2,

which was about fake news detection in general, these literature reviews provide the state of the task dealt with the corresponding chapter. Then in each chapter my research and the resulting solution is described.

For determining appropriate classification algorithms, I used the document as input and after applying different classification algorithms, I was able to obtain results on whether the news is fake or not. All the results of the experiment are discussed and presented in Chapter 5.

As discussed in Section 4.2, classification is not always completely correct and sometimes news contains both fake and non-fake part. Therefore the classification is complemented by fact-checking of potential fake news. Fact-checking is based on analysing the content and meaning of the text. This requires effort and is time consuming. Therefore, the goal of the task is to reduce the time and burden of fact-checkers. All experiments and results are presented in Chapter 6.

The final step is fact-checking. I have developed a system to check the facts of the news based on known facts additionally using metadata such as source, author, topic. For this purpose I have collected statements which have already been checked by different fact-checking organizations. The detailed analysis and review of fact-checking can be found in Chapter 7, and a prototype is presented in Chapter 8.

The next chapter will answer research question 2.

## 5 Fake News Detection through Classification

This chapter presents the first phase of the fake news detection approach- the detection of fake news through classification as shown in Figure 5.17. This chapter will describe this approach in detail. In addition to dataset exploration, I will also briefly explain the feature extraction phase in machine learning. Considering the previous related works and similar application areas, I discuss in Section 2.3.3 that the extracted features are suitable for the task and will help to classify the statement into fake or non fake news articles. The following research question is answered conceptually in this chapter.

***RQ2: What methods can be used to identify potential fake news?***

This research question is answered conceptually within this chapter after introducing related work and devising an approach and a methodology. The related work presented in this chapter is different from the Chapter 2 which is related to the awareness of the problem but the literature discussed in this chapter is related to the suggestion and development.

### 5.1 Introduction

Machine learning can help to solve complex problems such as fake news detection especially in cases where we have tacit knowledge or unknown knowledge (Leonard et al., 1998). It is difficult to detect fake news especially satire (Banko et al., 2007). For this reason people may be intentionally or unintentionally deceived. The problem of fake news can be solved or at least overcome with machine learning and artificial intelligence. In general, fake news detection is considered a challenging task (Lazer et al., 2018b) that requires multidisciplinary efforts (Nørregaard, Horne, & Adalı, 2019). I applied three classifiers such as Passive Aggressive, Naïve Bayes and Support Vector Machine. Simple classification is not completely correct in fake news detection because classification methods are not specialized for fake news (Meel et al., 2020). With the integration of machine learning and text-based



processing, we can detect fake news and build classifiers that can classify the news data. Text classification is mainly about extracting various features of the text and then incorporating these features into the classification.

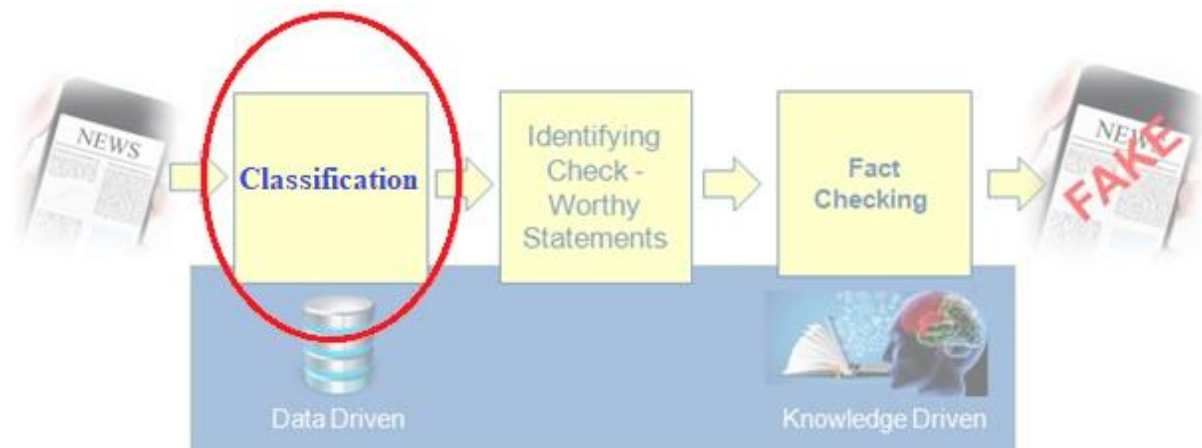


Figure 5.17: Text classification proposed diagram (General View)

The classification module contains text classification and metadata classification as shown in Figure 5.17. I briefly describe fake news detection through classification in this chapter. Fake news detection is a binary classification task that examines whether a news item is fake or not (Bajaj, 2017). A method that can first convert the theories of humor, irony and satire into a predictive method for satire detection (Rubin et al., 2016). The conceptual contribution of their work is to link satire, irony and humor. Fake news frames are then selectively filtered based on their potential to mislead readers. Traditionally rumor detection techniques are based on message level detection and then analyze the credibility based on data but real time detection based on the keywords then the system will gather related microblogs with the help of a data acquisition system that solves this problem. Zhou et al. (2015) propose a model that combines user based, propagation based and content-based models and checks real time credibility and returns the response within thirty-five seconds. It is difficult to cover up all types of fake news so my work is focuses on political news such as misleading content, false context, manipulated content and fabricated content as shown in Table 5.3.

Table 5.3: Seven types of fake news

Sr. No	Type	Details
1	False Connection	When headlines, visuals or captions don't support the content.
2	*False Context	When genuine content is shared with false contextual information.
3	*Manipulated Content	When genuine information or imagery is manipulated to deceive.
4	Satire	No intention to cause harms but has the potential to fool.
5	*Misleading Content	To frame an issue.
6	Imposter Content	When genuine sources are impersonated.
7	*Fabricated Content	New content that is 100% false, designed to deceive and do harm.

\*Types of news come under a political domain (Source: Stop FAKE.org)

Rubin et al. (2015) provide a way to filter, review and verify news. Their method which can divide fake content into three categories: serious fabrication, large scale hoaxes and humorous fakes. To introduce a hybrid model, three features (text of an article, user response and the source) of fake news are incorporated at one place. The first module captures the abstract temporal behavior of users, measures response and the text. The sources for each user are estimated by the second component value, which is further combined with the first module. Finally, the proposed model allows CSI to output predictions separately (Ruchansky et al., 2017). Since fake news can be easily shared on social media platforms, it can be difficult to automatically identify fake content. Information sources (visual cues & cognitive cues) and social judgment (cognitive, behavioral & affective) from Facebook data, specifically during the 2016 U.S. presidential election, explored that machine learning classifiers can be helpful to detect fake news (Janze & Risius, 2017). During stance detection, uses headlines based on n-gram matching to check binary classification with “related” vs. “unrelated” pairs. This methodology can be applied in fake news detection, especially in clickbait detection. For the experiments a fake news detection dataset published by fake news challenge (FNC)<sup>45</sup> on stance detection for experiments. To achieve the best results, evaluations are conducted presenting, Deep learning with natural language processing for fake news detection and different models applied (Bajaj, 2017). The lack of an efficient method to distinguish between fake or non-fake is the major challenge in this area. I have used three different machine learning techniques for experimental analysis based on the

<sup>45</sup> <http://www.fakenewschallenge.org/>

existing dataset which showed very remarkable and improved performance. Natural Language Processing (NLP) was used by us for pre-processing the data and the Python programming language was used for development. Passive aggressive, Naïve bayes & Support vector machine classifiers are useful for text based processing. The classifiers rank the text and convert it into three classes such as fake, not fake, and unclear.

### 5.1.1 Role of Machine Learning in Fake News Detection

We knew that in machine learning the main focus is on algorithms and these algorithms can improve automatically through experience. These algorithms rely on training data to make predictions or decisions without being explicitly programmed to do so (Bishop, 2006). Machine learning is data-driven programming (Liviu Ciortuz). Today, a wide range of emerging machine learning tools can be used to analyze data and extract accurate, relevant, and useful information to facilitate knowledge discovery and decision making (Jordan & Mitchell, 2015). Everyone cannot know about the world situation so we only rely on the news but the problem is that we do not know whether the news is true or not. I have already discussed the importance of fake news, so it is the big problem that needs to be addressed. Supervised learning in machine learning is the task of taking the input and predicting the output. When we talk about fake news here, the text of a news article is the input and in turn, it can be the binary number '0' or '1' or true or false or fake or not fake. Many approaches have been used to classify the text as fake or not fake but in this chapter, I have focused on natural language processing (NLP) basic method TF-IDF vectorizer (Section 5.4). I focused on three models: Naïve Bayes, Support Vector Machine and Passive aggressive algorithms for evaluating our proposed approach. I briefly discuss these models in (Section 5.2.3). In the following figure I describe the general schema of machine learning methods.

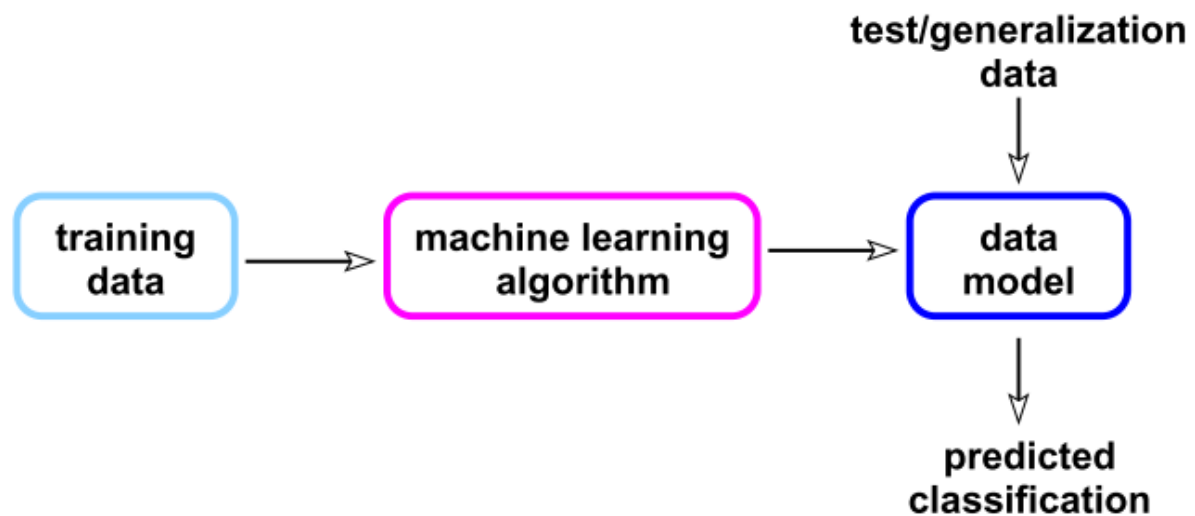


Figure 5.18: General schema for machine learning methods

Although machine learning based approaches are helpful in detecting fake news, we are also aware of the fact that some machine learning techniques are the reason for the generating fake texts. Neural fake news is a concept that intentionally generates fake articles. Some NLP based frameworks were created just to spread false information and mislead readers such as BERT, GPT, etc<sup>46</sup>. These are platforms that spread propaganda and miscommunication on the Internet. As shown in Figure 5.19, in the next sections I will discuss in detail about text classification and its challenges, which is my starting point for detecting fake news.

---

<sup>46</sup> <https://www.analyticsvidhya.com/blog/2019/12/detect-fight-neural-fake-news-nlp/>

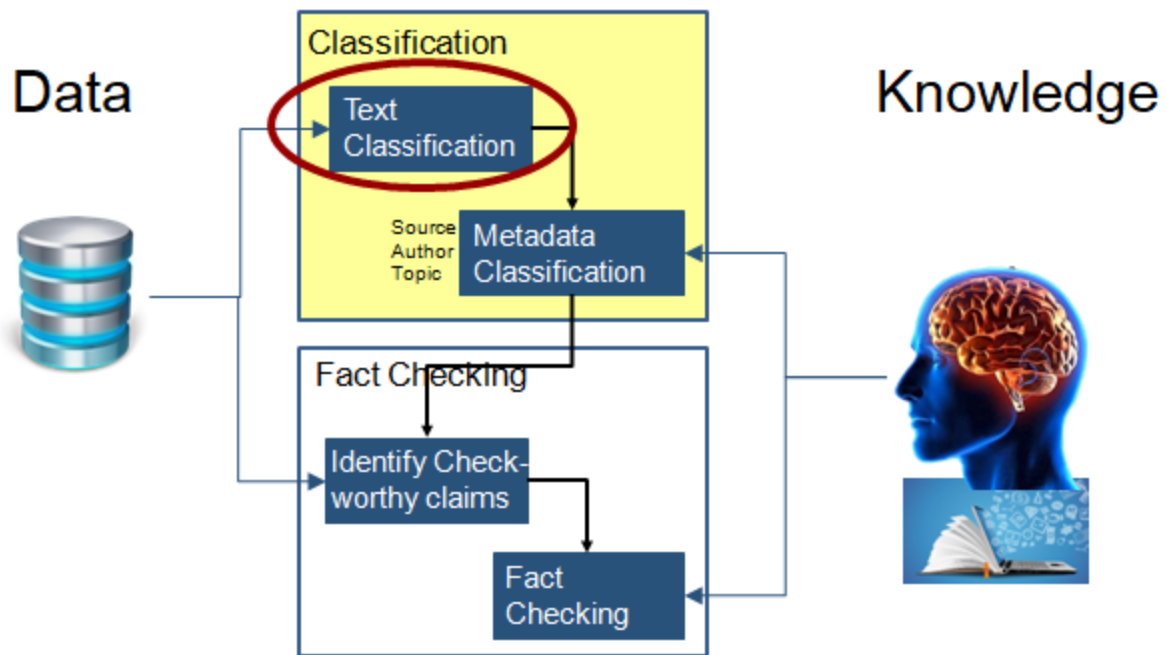


Figure 5.19: Text classification development

## 5.2 Methodology for Fake News Detection through Classification

In this section, I am going to start by explaining how I gathered the dataset used in this work, and different analyses performed on it to have a better overview. I model fake news detection as a classification task and follow a supervised learning approach to tackle it. I focus on the available datasets and evaluation metrics for this task. The next section describes the dataset collection, exploration, model development, model evaluation and experiments.

### 5.2.1 Dataset Exploration

The dataset used for this task was drawn from a public domain. Fake news articles were collected from an open-source Kaggle<sup>47</sup> dataset published during the 2016 election cycle. The collection consisted of 18000 news articles. These articles were collected from news organizations such as NYT, Guardian and Bloomberg during the election period. The articles were separated by binary labels 0 and 1. The dataset was already qualitatively sorted using the labels fake, non-fake and not clear (see Figure below).

<sup>47</sup><https://www.kaggle.com/mrisdal/fake-news>.

id	title	text	label
10294	Watch The Exact Moment Paul Ryan Committed Political Suicide	Google Pinterest Digg LinkedIn Reddit Stumbleupon	FAKE
3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Monday that	Not Fake
10142	Bernie supporters on Twitter erupt in anger against the DNC after Kaydee King (@KaydeeKing) November 9, 2016		Not Clear

Figure 5.20: Dataset row structure example

This classification can be seen in the figure below, where I have 15,115 articles from fake categories and 1,846 from the true category. The rest of the articles were not clear for other reasons, e.g. missing unique id, unclear source, etc. The task itself leads to a quite imbalanced dataset, as can be seen in Figure 5.21 (a) where out of the total number of articles, about 12% are from the TRUE category, i.e. non fake. This imbalance can also be seen in previous similar work (Gencheva et al., 2017; Ingrid Yanuar Risca Pratiwi, 2017).

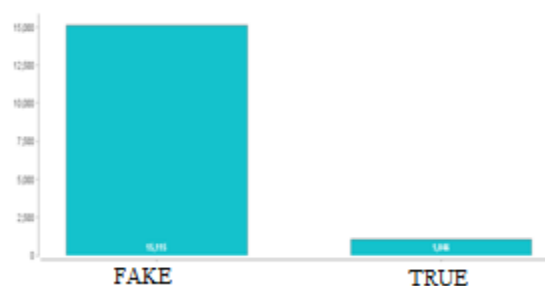


Figure 5.21: Class distribution (a) Kaggle Dataset (2016)

Collecting the fake news dataset was easy as I highlighted above, but getting the real news for the fake news dataset is a difficult task, so we need a second dataset for this purpose. I gathered a real news dataset containing 5000 real news articles from the Signal Media News dataset<sup>48</sup> of which 2,541 belong to the fake class and 299 belong to the true class (see Figure 5.21 (b)). The collected articles were from news media organizations such as the Guardian, Bloomberg, the New York Times, NPR, etc. The dataset was published in 2016 before and after the U.S. presidential election; our focus is on political news so the combination of these articles is important for training and testing the model.

<sup>48</sup><https://research.signal-ai.com/newsir16/signal-dataset.html>

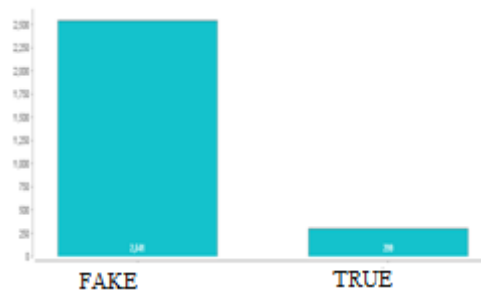


Figure 5.21: (b) Signal Media News Dataset 2016

I used RapidMiner<sup>49</sup>, a powerful machine learning tool, for data exploration, preparation, information extraction, result visualization and result optimization. I analyzed the fake and real sentences through RapidMiner and initial results can be seen below:

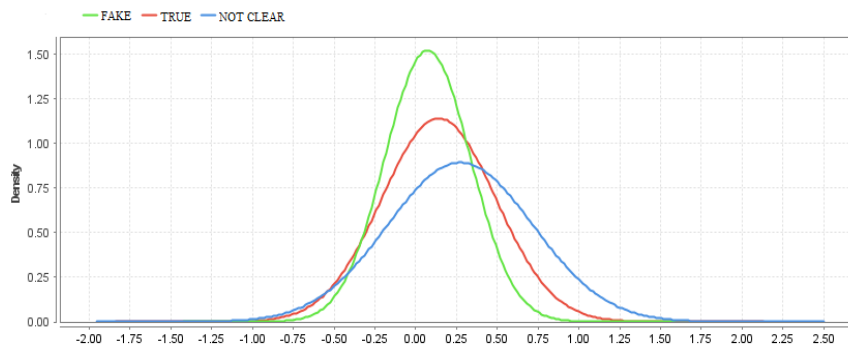


Figure 5.22: Dataset class labeling chart

Figure 5.22 shows dataset labeling chart respectively the combination of fake, true and not clear claims and the percentage of those claims.

<sup>49</sup><https://rapidminer.com/>

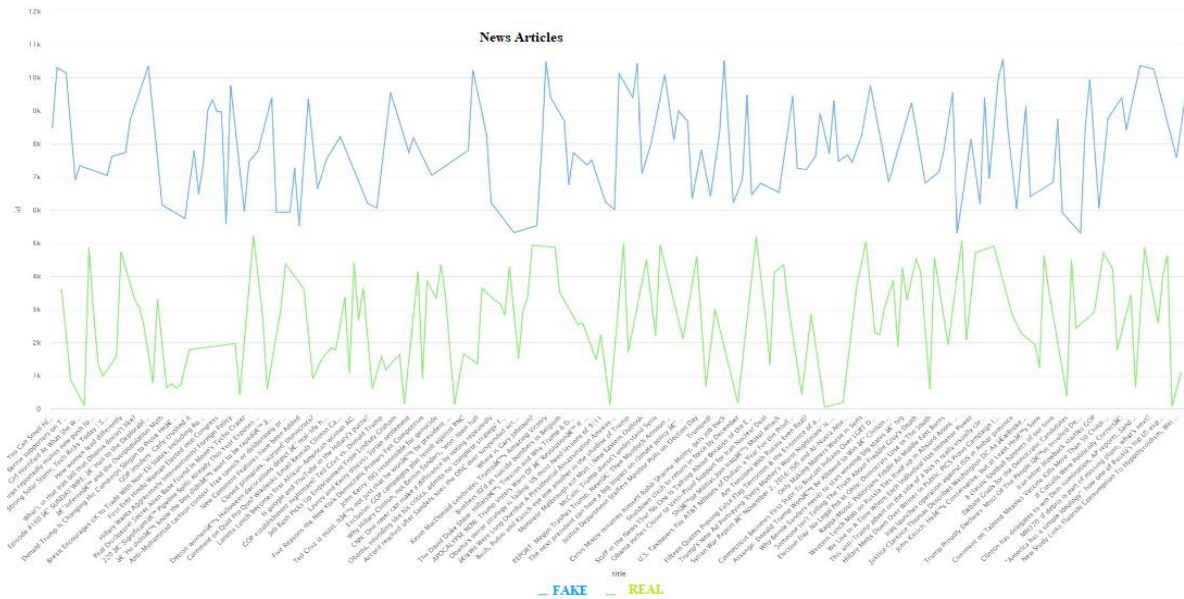


Figure 5.23: Fake and Real news sentence level comparison (Spline Plotting)

Using RapidMiner, I performed the sentence level comparison just to examine the dataset to understand the claims in the document. The comparison results are shown in Figure 5.23.

### 5.2.2 Missing Values and Correlation

In the dataset not all columns are helpful for prediction, so we check the patterns of the dataset by Rapid Miner. This will help us to understand the dataset and get an external view before going in-depth. I checked the correlation between the columns and the missing values in the columns. Rapid Miner auto modelling also helped us to look at the different values in the columns and the stability of the column values shown in Figure 5.24.





Quality	Name	ID-ness	Stability	Missing	Text-ness
	Body ID	86.55%	0.56%	0.00%	88.57%
	articleBody	57.34%	0.81%	33.23%	90.07%

Figure 5.24: Data exploration (Correlation, Stability, ID-ness and Missing)

I considered only those sentences that were correctly labeled and had a unique id, and excluded those that had missing values before data exploration. The dataset contains the columns unique id (number), title (the title of the news), text (body of the news), and label (fake or not fake). I observed that in some cases the body of the news was not in detailed as the title of the news so in this case, I checked the missing values and the correlation between the sentences. I have already discussed and highlighted this in Figure 5.24. In the next step, I checked the quality of the sentences. For this reason, I used an auto model which is an extension of the rapid miner and helps us to check the sentence levels and relationships between sentences. In the next diagrams, I performed some experiments for a better understanding of each sentence level. Figure 5.25 (a) highlighted sentences, (b) checking sentences stability by curve view, and (c) comparing real and fake labeled articles, these are the examples of those experiments where we compare the sentences individually and then with both real and fake sentences. This will help me later in modelling the task. The dataset contains different topics, but mainly the articles were related to politics.



Figure 5.25 (a): Sentence wise data exploration (Line Plotting)

The figure above shows how all the sentences in the dataset are measured. Each row shows which sentence data is complete or incomplete since some sentences contain missing values that were not considered in the classification. The goal of this section is to visualize the dataset for understanding.

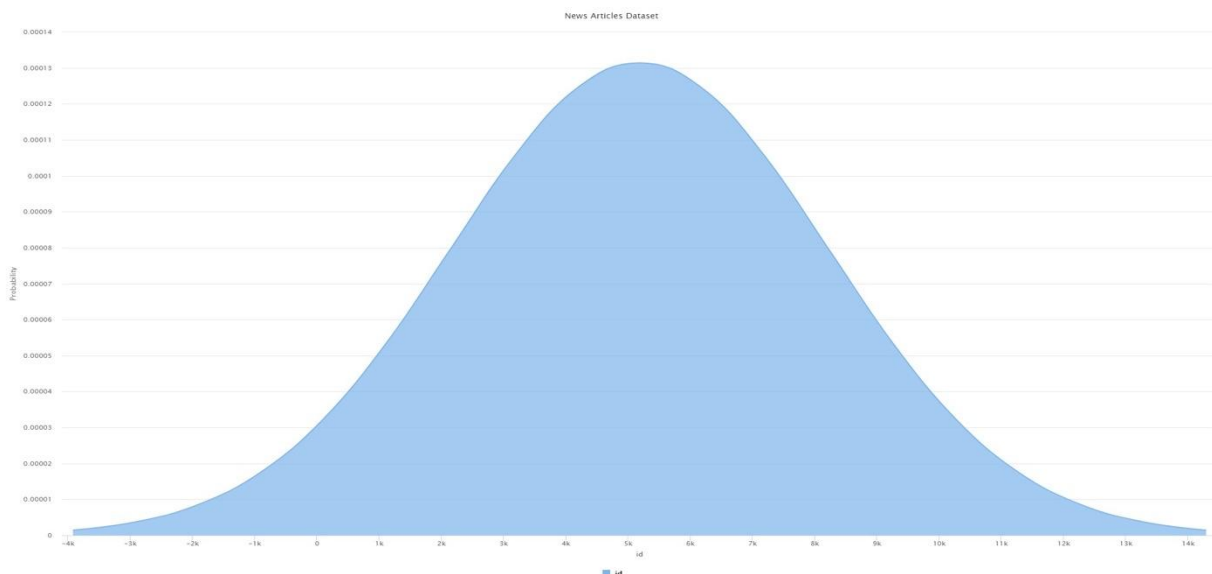


Figure 5.25 (b): Fake and real news data comparison (Bell Curve)

In the next step, I checked the stability of the sentences in the dataset through RapidMiner's bell curve data exploration technique. With the bell curve, I follow the convenient way of estimating the calculations. As in the previous figure, I explored all the sentences which are fake or non-fake for the data visualization, but here in this method, I compare the fake and real sentences according to the unique id which will help us to compare in the next step.

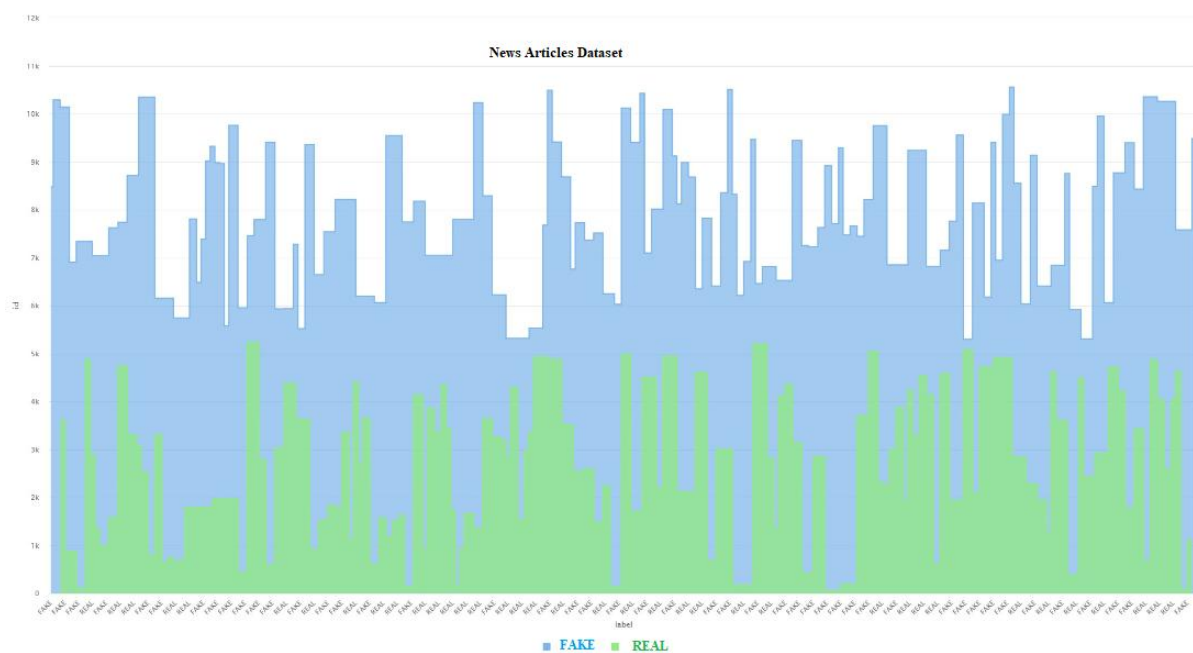


Figure 5.25 (c): Fake and real news data comparison (Step Area)

In Figure 5.25 (c) we can see the fake and real sentences in different colors; as I discussed above, we have 2541 sentences out of a total of 5000 sentences from the false class and 299 from the true class. So here we can see that the dominant class in the dataset is the false class, which I will discuss in more detail in the following sections.

When encoding a text for classification, it is common to represent words in a continuous space as vectors that embed linguistic information, called word vector embeddings (Naseem et al., 2021). Distributed representations of words in a vector space help learning algorithms achieve better performance in natural language processing tasks by grouping similar words (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013). Word embeddings are constructed based on the distributional hypothesis, which states that words used in the same contexts tend to have similar meanings. Word vectors capture many linguistic regularities for identification, for example vector operations  $\text{vector}(\text{'Paris'}) - \text{vector}(\text{'France'}) + \text{vector}(\text{'Italy'})$  when we

compare the results in a vector that is very close to the target vector ('Rome'), and vector ('king') – vector ('man') + vector ('woman') is close to the vector ('queen') (Mikolov et al., 2013). GloVe is a word embedding model trained by a team of Stanford researchers using global word-to-word co-occurrence statistics (Sharma et al., 2017). This vector representation model is essentially a “count-based” model. The main intuition underlying the model is the simple observation that ratios of word-to-word co-occurrence probabilities have the potential to encode some form of meaning (Pennington, Socher, & Manning, 2014).

### 5.2.3 Models Description

Different classification models can be applied but in order to select the most appropriate one and tune its parameters, we conducted several experiments with different models. I have experimented with classification models that have proven to be effective and give good results in related sentence classification tasks. Some of the models did not give good results and were discarded; one of them was Logistic Regression, but Support Vector Machines, Naïve Bayes and Passive Aggressive gave promising results and I continued experimenting with them. To check the accuracy, I compared our results with other datasets through performance metrics.

#### 5.2.3.1 Naïve Bayes

Naïve Bayes is a powerful classification model that performs well when there are fewer records and less memory (Ng et al., 2014). It does not give good results when the words are co-related to each other (Inggrid Yanuar Risca Pratiwi, 2017). It is derived from Bayes' theorem. This classifier assumes that all labeled values are independent of a pair of features. The Naïve Bayes classifier is a fast and accessible technique but one of the major drawbacks of the method is that it determines all features separately. Due to this problem, it is difficult to determine the news due to the lack of coordinated analysis. Qin (2018) discusses how to implement fake news discovery on different social media sites with the help of Naïve Bayes. They used Facebook, Twitter and other social media applications as data sources for news.

#### 5.2.3.2 Support Vector Machine

Support Vector Machine (SVM) performs well on the problem of detecting fake news (Banerjee, Chua, & Kim, 2015). It performs supervised learning on data for regression and

classification. When we give data to SVM it computes the data and converts it into different categories. The way it works is that given two classes of vectors in my dataset, we can define a hyperplane that effectively separates two classes that are fake or not fake. This optimal hyperplane is determined with two other support vectors that are parallel to the given hyperplane. These support vectors are in line with the data item closest to the proposed division hyperplane and are equidistant from the proposed division hyperplane. The optimal hyperplane is determined by finding the solution that maximizes the distance between the two support vectors and the division hyperplane. Unfortunately, the decision boundary, in this case, cannot always be linear. One way to solve this is to use something called “Kernel Trick”. The kernel trick essentially works by setting a decision boundary in a higher-dimensional space, which can be done after applying a function to the data. SVM is another very popular choice for classification. The “Kernel Trick” can improve one of the major limitations of logistic regression, by allowing non-linear decision boundaries. Although it works very well for small training sets, SVMs are not usually used for large training set problems because they are quite inefficient to train and run. The advantages of a Support Vector Machine are speed of classification, speed of learning, accuracy, and tolerance to irrelevant features and noisy data (Davuth, N., & Kim, S. R. 2013). Basic SVM models are dealing with situations where the exact values of the data points are known. This paper presents a survey of SVM when the data points are uncertain (Wang et al., 2014).

#### 5.2.3.3 Passive Aggressive

Passive aggressive algorithms are mainly used for classification (Kostakos, Nykanen, Martinviita, Pandya, & Oussalah, 2018). A classifier is considered useful in the context of fake news detection if it achieves both high precision and recall. The performance of the classifier has been shown to be superior to many other alternative methods such as Online Perceptron and MIRA. It examines the space of weighted vectors that satisfy the decision criteria. Using the MIRA or SVM does not go ahead to any further enhancement over the perceptron but the use of ranking as opposed to classification leads to a 0.4% reduction in word error rate (WER) which is statistically significant (Dikici et al., 2013).

#### 5.2.3.4 Logistic Regression

One algorithm commonly used in discrete natural language classification is logistic regression (Friedman et al., 2000). Logistic regression not only provides discrete

classification, but also probability value associated with that classification. The reason logistic regression is because of the way linear regression classifies outliers. Because linear regression uses a fairly simplified, linear “decision boundary” to classify instances, outlier variables can be misrepresented. Logistic regression ameliorates this problem by applying an activation function to each variable before applying the decision boundary. This reduces the impact that outliers can cause. The activation function most commonly in logistic regression is the sigmoid function. To measure the performance of the solution, the loss function Cross Entropy should be used since the classification provided is a probability. The sigmoid function is used not only to improve the accuracy of a classification, but also to return a number between 0 and 1 that represents the probability. Logistic Regression is fairly simple to understand making it a popular choice for simple, largely linear classifications. Logistic regression is well suited to solve the problems where we have a large and uniform set of features (Tacchini et al., 2017). It is used to estimate the relationship between variables after applying statistical methods. It performs well in binary classification problems because it uses classes and requires a large sample size for initial classification. For the logistic regression model, the C and solver parameters were studied. The C parameter is the inverse of the regularization strength. This is the value by which the model attempts to minimize the number of misclassifications, at the expense of decreasing the distance between the decision boundary and the different classes.

#### 5.2.3.5 Neural Network

Neural networks perform well when we work with multidimensional data. But for this reason, we need a large sample size and memory to achieve the maximum accuracy of the classifier. Also, it is intolerant to noise. A neural network is a function that consists of a collection of basic features (neurons) and weighted connections organized in a network layout (Svozil et al., 1997). The network is organized by a “training” process that changes the weights based on the output error produced. Neural Networks consist of different “layers”, of which there are usually three main sections. The “input layer” receives input variables, which are then passed to the “hidden layer”, which contains one or more layers of function nodes (Dongare et al., 2012; Yan et al., 2006). Each layer of nodes in the hidden layer can contain one or more neurons. There is no hard and fast rule about how layers in the hidden layer should be organized in basic neural networks, however, a neuron in the hidden layer usually receives input from neurons in a preceding layer. This previous layer can be another hidden layer or

the input layer. To keep the network from becoming too complicated, the neurons in their respective layers are regularized by only receiving inputs from the established previous layer and sending outputs to the next layer in the network. The outputs from the hidden layers then go into an output layer that generates the prediction. The function used for each node in a neural network is called “activation function”. It is called an activation function because when a certain value is reached, the output increases significantly. This allows neurons to “fire” when the input parameter approaches a certain value. An example of this is the sigmoid function (described in Equation E1). The activation function is applied to the sum of the products of the previous outputs  $O_i$ , and their respective weights,  $w_{ij}$ . This gives the output which is then passed to the next layer. This process is used to make a prediction. This is important for the process of training a neural network.

$$O_j = F\left(\sum w_{ij}O_i\right) \quad (\text{E } 1)$$

When training a neural network, the dataset used for the training process should be randomly divided into a training set and a test set (Mazurowski et al., 2008). The test set is used to evaluate the model after training. When training a neural network, all the rows of data from the training set are repeatedly passed through the network. The weights between the nodes are adjusted after each repetition based on the error of the output. The network produces a prediction  $y$  (as described below) which is subtracted from the actual output value in the training set  $z$  to produce an error value  $\delta$

$$\delta = z - y \quad (\text{E } 2)$$

Using a process called “backward propagation” the error is fed back through the neural network to produce an error  $\delta$ , values associated with each neuron in the hidden and output layers (Nguyen et al., 1990). This is done by multiplying the sum of the previous errors by their respective weights,  $w$  as highlighted in the Equation (E3).

$$\delta_i = \sum w\delta \quad (\text{E } 3)$$

This process is repeated for each neuron in the hidden layer (Karsoliya 2012). In Equation (E4) the weights between the input and the first hidden layer are then updated by using the current weight  $w$  and adding the product of the learning rate  $\eta$ , the error value assigned the

neuron  $\delta$ , the derivative of the sigmoid function  $f'$  and the value of the input  $x$  that was originally passed through.

$$w' = w + (\eta\delta f'x) \quad (\text{E } 4)$$

This process is then repeated for each weight in the network; however, for weights connecting neurons in the hidden and output layers, the input variable is denoted by  $y$  instead of  $x$ . This process should be repeated until the error between iterations stops decreasing or the desired number of iterations is reached. Neural networks have become increasingly popular in recent years because they are very effective in classifying non-linear data (Ripley 1994). Neural networks are also useful for building a model based on a large number of different input features. However, they have limitations, as they require a lot of computational resources and it is very difficult to identify which input features are most important in the classification, and how the network determines the final classification. Therefore, it is difficult to debug a neural network-based model if it performs poorly.

#### 5.2.3.6 Multilayer Perceptron (MLP)

A multilayer perceptron is an example of a deep neural network classifier (Lin et al., 2013; Savalia et al., 2018). This means that the hidden part of the network consists of multiple layers of nodes. Variables are passed forward, while error values are repeatedly passed backward until the error value produced cannot be reduced any further, which is called "convergence". MLP is an example of an artificial neural network that is helpful for the solution of different problems like pattern recognition and interpolation (Noriega 2005).

#### 5.2.3.7 Recurrent Neural Networks (RNN)

A neural network is a machine learning technique in which layers of nodes are created in a network, with associated weights connecting each layer of nodes (Niklas 2016). Simple neural networks have three main layers: input, hidden and output. Data that we want to classify is fed into the input layer before being passed to the hidden layer. The hidden layer may contain several other layers. It is then passed to the output layer, which performs classification. As the data is passed through the network, an activation function (e.g. a sigmoid activation function) is applied to the data at each node. Weighting values are also



applied to the data at each link between nodes. These weighting values are determined through a process known as “training”. Using a dataset where the classification is already known, each data item is individually passed through the network and an output is generated. The classification obtained at the end of each iteration is then used to calculate an error. This error is again passed through the network, distributing the error among the individual weighting values and changing them. This is done over and over again, gradually increasing the performance of the network and making it more accurate (Abdelzaher et al., 2002; Niklas 2016). Once the algorithm is trained, it can be further used to create a classification for a new dataset. A recurrent neural network (RNN) differs from simple “feed forward” networks in that each node can have memory. This means that a prediction can take into account previous inputs to improve the prediction, and some context can be added. This means that the output of a node is influenced by the current value and previous values. Mallya et al. (2018) proposed that RNN-based models are fit for the task and performed well. LSTMs work particularly well with data that is sequential, such as natural language, because they allow longer context to be represented in the prediction, as opposed to considering only the immediately preceding output. A very popular RNN implementation is known as long short-term memory (LSTM) (Tian et al., 2015; Kratzert et al., 2018). This is achieved by training LSTM layers within the network when to retain or forget information. This means that the network can retain information at a variable rate (Heaton, 2018). This long memory, allows for more accurate prediction. Within an LSTM cell in a node, lies the ability to store and forget previous input. The LSTM contains three gates to decide whether to forget information about a previous input: an input gate, an output gate, and a forget gate. Each time data is passed through the cell, the information about the previous inputs is applied to the current input based on the actions of these gates.

#### 5.2.4 Model Comparison

Fake news data collection could vary significantly due to different research purposes. Some models used source reliability and network structure so the big challenge in those cases is to train the model. We examined the performance of machine learning algorithms: Naïve Bayes, Support Vector Machine, Passive Aggressive, Logistic Regressions, Neural Networks, Multilayer Perceptron, and Recurrent Neural Network. The obtained results verify the pros and cons of the compared different machine learning algorithms when they have been used

specifically in detecting fake news. Naïve Bayes is used for classification tasks. It can be used to check whether the news is authentic or fake (Pratiwi et al., 2017). Another supervised machine learning algorithm that learns from the labeled dataset is Support Vector Machine (Singh et al., 2017). Authors applied various classifiers of machine learning but the Support Vector Machine has given the best results in detecting fake news. The decision tree algorithm of machine learning can break the dataset into different smaller subsets (Kotteti et al., 2018). Kotteti et al. (2018) used different machine learning algorithms but they found good results through decision tree. Kaliyar et al. (2020) have used a neural network to detect fake news in their work. The main problem occurs during the training of these algorithms if the training dataset is imbalanced (Wang et al., 2020). We intend to train the dataset on different algorithms to determine which algorithm performs well. After comparison, we have come to know that three classifiers Naïve Bayes, Passive Aggressive and Support Vector Machine performed well (see Table 5.4). The reason for the good performance of these algorithms is that they perform well on the text-based dataset. Passive Aggressive computes conditional probabilities of two events on the basis of text occurrence individually and differentiates each event/class accordingly. The Passive Aggressive algorithm is better than other algorithms due to its functionality. The accuracy of up to 93% is good as we evaluated the trained model with different evaluation measures which are discussed in Section 5.5. The overall obtained results and comparison with other classifiers highlighted in Table 5.6.

### 5.3 Model Development for Fake News Detection

My proposed model starts with the extraction phase and then includes four main steps. The first step is related to natural language processing models where I measure the frequency of words and build the vocabulary of known words in fake news datasets. Next, fake news is detected using NB, SVM and PA classifiers. Finally, I tested our models with different experiments and some other datasets and proposed the final model for detecting fake news. Figure 5.26 shows the flowchart of our model.

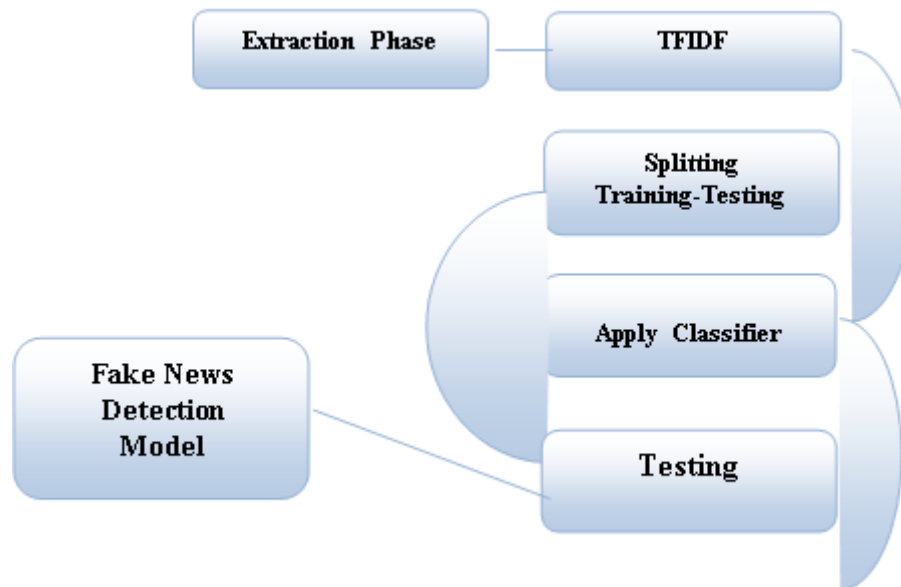


Figure 5.26: Fake news detection model

### 5.3.1 Pre-Processing

The goal of this process is to reduce the size of the actual dataset by removing irrelevant information that is not necessary for classification. Then the data was modified for processing so that the first half of the data was tagged with a fake label set and the second half was tagged with a real label, which would not cause impartiality when applying machine learning methods to this data. A common task in NLP<sup>50</sup> is tokenization, where a text or set of texts is decomposed into individual words. In this step, our goal is to convert words into their basic form in order to understand them better (Torunoglu, Çak, Ganiz, Akyoku, & Gürbüz, 2011). Then, I applied stemming which reduces the number of words based on word type and class. Suppose I have three similar words in a dataset such as running, ran, and runner; these are reduced and the word run is changed. I have used stop word removal as it removes common words used in articles, prepositions and conjunctions (K. & R., 2016). There are different stemming algorithms but I used Porter<sup>51</sup> because of its high accuracy rate. The overall data cleaning process is shown in Fig 5.27.

<sup>50</sup><https://www.nlp.com/>

<sup>51</sup><https://www.nltk.org/howto/stem.html>

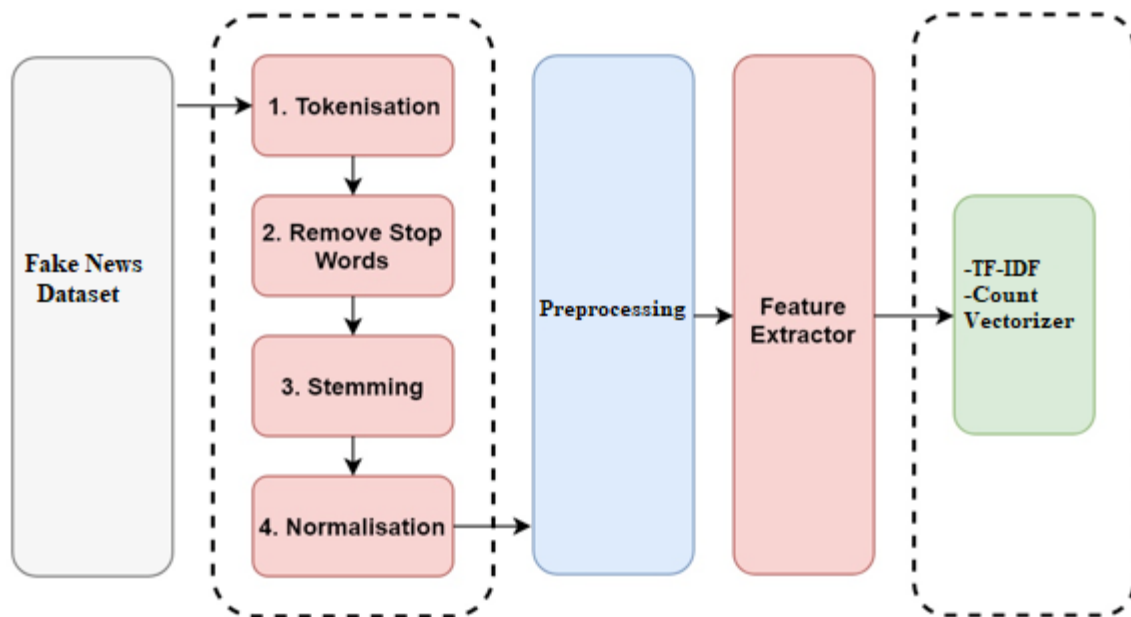


Figure 5.27: Data cleaning steps in NLP starting from raw dataset to machine learning models

### 5.3.2 Stop Word Elimination

Irrelevant and redundant features in the dataset have a negative impact on the accuracy and performance of the classifier. Therefore, in these cases, I performed feature reduction to reduce the text feature size which reduces words like “the”, “and”, “there”, ”when” and focus is only on those words which appear a given number of times. This is done by using n-number of words, lower case and removing stop words as we knew that the sensitivity of the problem increases every second without check and balance.

### 5.3.3 Count Vectorization

Count Vectorizer provides an easy way to collect text documents and help build the vocabulary of known distinctive words, but also to encode new documents using this vocabulary (Vijayaraghavan, Wang, Voong, et al., 2020). Give a collection of text documents,  $s$  to CountVectorizer and it will generate a sparse matrix  $A$  of size  $m$  by  $n$ , where  $m$  = total number of documents,  $n$  = total number of distinctive words used in  $S$ .

### 5.3.4 TF-IDF

To measure a term in documents over a dataset I used term frequency-inverted document frequency. A term importance increases in the document which appears in the dataset and also the frequency of the words. So with the help of this method, we can weight metrics that I used for information retrieval (Gilda, 2017). TF-IDF for the word for document  $d$  and corpus  $D$  is calculated in Equation (E5).

$$TF-IDF(w)d, D = TF(w)d \times IDF(w)D \quad (E 5)$$

Suppose we have a document with 100 words and we need to calculate TF-IDF for the word that one is “rumor.” The word “rumor” occurs in the document 4 times; then we can calculate,  $TF=4/100=0.04$ . Now, we need to calculate the IDF; let us assume that we have 200 documents, and “rumor” appears in 100 of them. Then,  $IDF(\text{rumor}) = 1 + \log(200/100) = 0.5$ , and  $TF-IDF(\text{rumor}) = 0.04 \times 0.5 = 0.02$ .

## 5.4 Experimental Setup and Evaluations

The development work was done in Python<sup>52</sup> using different available tools and libraries. I have highlighted the tools and libraries we used for the implementation and provided a corresponding reference in each related section. For performance testing, I used the Sklearn<sup>53</sup> Grid Search functionality for task utilization. I have observed that the relative frequency of words may also be the reason for dividing them into fake and non-fake classes. Using a word cloud visualization we observe the corpus trend shown in Figure 5.28. The word cloud visualization reflects important word entities. For example, we can observe the very common words Political, Americas, 2016, President, Obama and presidential debates from the dataset. I used different news sources for the test and training datasets so that we can observe how well our models generalize to unseen data points. In the first step, I applied the text extraction features included the text classification module.

---

<sup>52</sup><https://docs.python.org/3.7/>

<sup>53</sup><https://scikit-learn.org/>



### 5.4.1 Cross-Validation

Performing a train-validation split on the training set can give varying results, especially in a dataset like in this work. The validation set may be a subset of the dataset that is very easy to predict, or in the other case, very hard. Instead of splitting the training set into fixed train and validation sets, I chose cross-validation to tune in the hyperparameters of the models and chose the best-performing ones.

In  $k$ -fold cross-validation the dataset  $D$  is split into  $k$  exclusive subsets, so-called folds:  $D_1, D_2, D_3, \dots, D_k$ . The models during validation are trained and tested  $k$  times, each time  $t \in \{1, 2, 3, \dots, k\}$ , trained on  $DD_t'$  and tested on  $D_t$  (Kohavi, 1995).

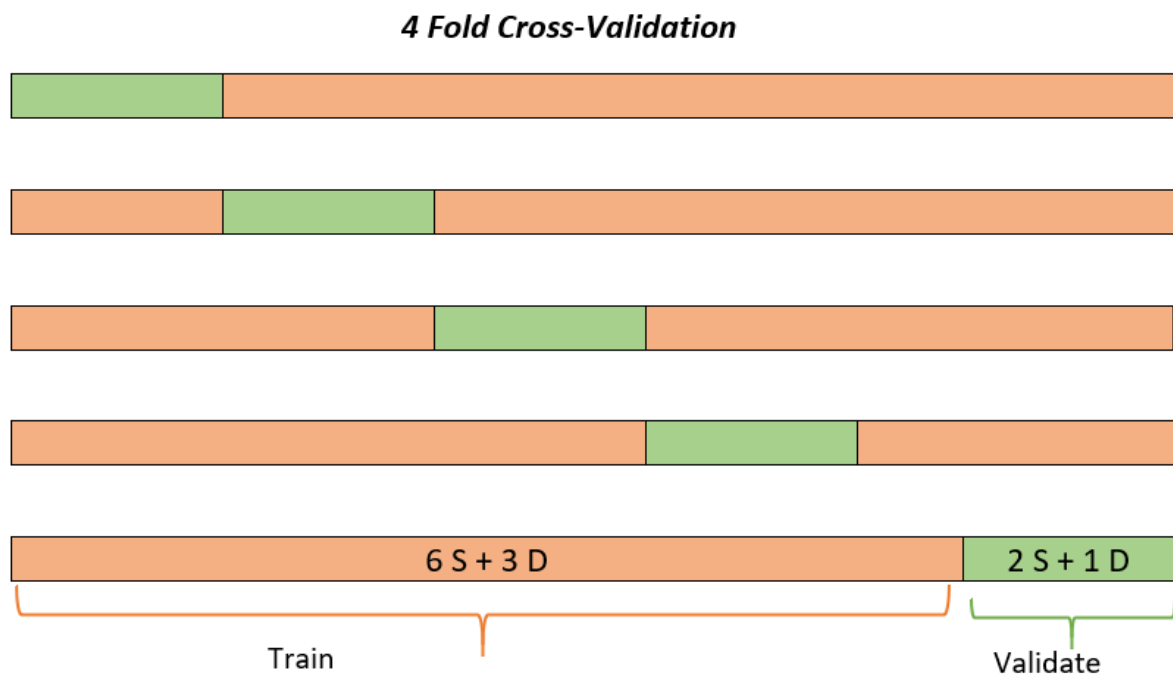


Figure 5.29: 4-Fold Cross-Validation (Kohavi, 1995)

In this work, I have used 4-fold cross validation as visualized in Figure 5.29. It is important that each of the folds is a good representation of the dataset. The ultimate goal is that the models should learn from the training set and from that to be able to generalize well when encountering new data. By using cross-validation, each of the debates and speeches is predicted once, allowing to see how the models perform in different validation sets containing new data. Furthermore, the validation set is also close to 20% for each fold, roughly the same as in the final test set. The mean of measures achieved from cross-validation can give good insights into how well the models perform on average and present

more robust scores for the classification models. The results for the models after comparing with each other the final hyperparameters are shown in figure 5.30. The distribution of the folds, with the total number of articles that are fake, non fake and not related. In each of the folds, the proportion of non fake articles in the training and validation sets makes roughly 12% of the sentences as shown in Figure 5.29. Despite being trained on smaller datasets than the final models, they show good results and generalizations for the validation sets in each fold. Comparing SVM, NB and PA, the latter has better results with an F1 score 93% higher than the SVM model. The expectations are that the models are going to perform in a similar way when trained in the while train set and tested in the final sets.

For further analysis, I applied different combinations to check the accuracy of our model with other models in Figure 5.30. The objective is to check the performance metrics individually to get a clear overview of which model performs well in which metrics and which decreases performance. Accuracy comparison of PA (93%) and SVM (89%) (a), PA (93%) and LR (78%) (b), PA (93%) and SVM (84%) (c) with different datasets, PA (93%) and NB (85%) (d), SVM (84%) and NB (85%) (e), NB (85%) and SVM (71%) (f), SVM (89%) and LR (78%) (g) and SVM (89%) and NB (85%) (h).



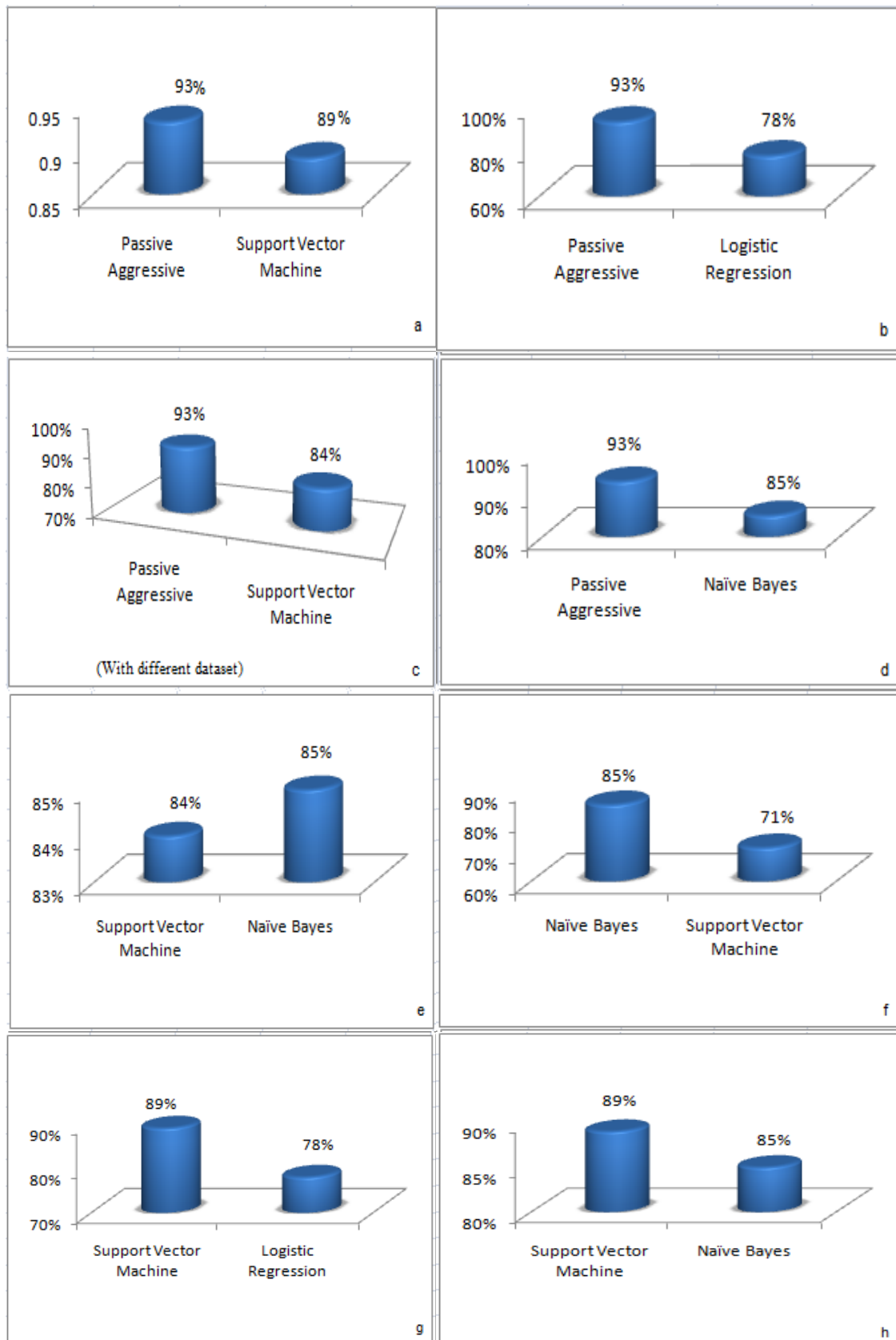


Figure 5.30: Accuracy comparison with different Algorithms (a) PA with SVM (b) PA with LR (c) PA with SVM (d) PA with NB (e) SVM with NB (f) NB with SVM (g) SVM with LR (h) SVM with NB

It is important to clarify that the cross-validation method was just used to select models and tune in hyperparameters, but not to train the final models. All the models that were created during cross-validation were not used at the end. Instead, final models with best-performing hyperparameters observed during cross-validation, were trained in the whole dataset used during cross-validation. To test the models I used evaluation methods which are discussed below. The performance of the models reported is the ability of the final models to predict the test set.

## 5.5 Evaluation Methods

In this section, I discuss how to evaluate the performance of fake news detection algorithms through classification. A classification model can achieve very high accuracy, but these high values come from the dominant class which accounts for more than 90% of the entire dataset, while the accuracy of predicting sentences can be very low. Some metrics have been developed to tell a truthful story when working with imbalanced classes, to get a better view of how the classifier predicted each class. For this work, I will use the metrics presented below, that provide more information about the performance of the model.

### 5.5.1 Confusion Matrix

A confusion matrix (Stehman, 1997; Visa et al., 2011) shows all the predictions that the classifier makes about the data. To evaluate the performance of the different models, a set of metrics must be used. These metrics are calculated based on the results of a test, where ‘True Positive’ and ‘False Negative’ are values that were correctly classified and ‘True Negative’ and ‘False Positive’ are values that were misclassified. In this case, correct values are instances that are labeled as fake news and incorrect values are labeled as real news. Positive values are data points that have been classified as fake news and negative values have been classified as real news. True predictions are on the diagonal of the table, the higher the numbers, and the better the classification. In this particular task, it would be more important to have fewer false negatives, or true predictions than fewer false positive one. The results of the confusion matrix can be seen in Figure 5.31.

	Negative Prediction	Positive Prediction
Actual Negative Class	True Negatives (TN)	False Positives (FP)
Actual Positive Class	False Negatives (FN)	True Positives (TP)

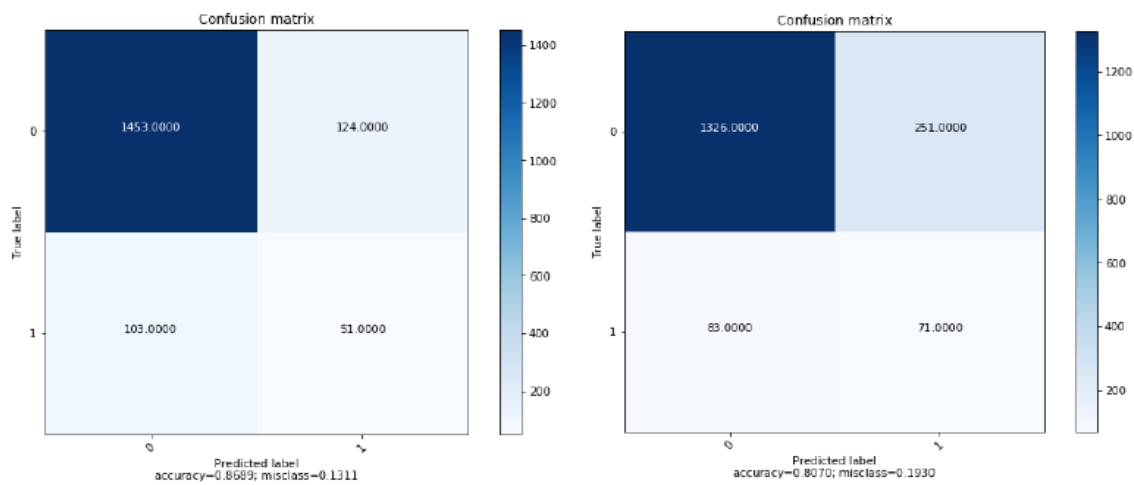


Figure 5.31: Confusion matrix for NB and SVM

**Accuracy:** The accuracy of a model is defined as the proportion of correctly classified instances. This is represented by dividing the number of True Positives (TP) and True Negatives (TN) by the total number of values. The model is given as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (E 6)$$

**Precision:** Precision is a measure of the accuracy of a classifier, it measures how many of the sentences were classified correctly as fake or non-fake (Powers, 2011; Saito and Rehmsmeier, 2015). The precision formula can be shown below where true positive sentences are based on the combination of true positive plus false positive.

$$Precision = \frac{TP}{TP + FP} \quad (E 7)$$

**Recall:** Recall is a measure of classifier completeness, it is often considered the most important measure in computational journalism (Powers, 2011; Saito and Rehmsmeier, 2015). Recall measures how many of the total sentences are classified based on true positive on the basis of true positive plus false negative, which is more important in this task sentences being classified as unclassified. This is represented as:

$$Recall = \frac{TP}{TP + FN} \quad (E 8)$$

**F1 Score:** The F1 score is the weighted average of precision and recall, it can take values between 0 and 1 (Powers, 2011). The relative contribution of precision and recall to the F1 score is the same, as in the formula shown below. The F1 score is used as the primary evaluation metric to select the best model.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (E 9)$$

## 5.6 Results and Discussion

I conducted experiments with different feature set combinations. As explained in Section 5.2.3 with details of each model I used. Figure 5.32 shows the details of the classifiers and then the performance measures of accuracy, precision and recall accordingly. As can be seen in the table, all classifiers achieve performance well above the baseline 0.50. The best performing classifier is PA, when we check the performance by accuracy and precision. The recall is slightly lower due to the noise in the dataset or the functionality of the classifier in processing the dataset. I applied the three classifiers mentioned below on the datasets 1 and 2, which are from the Kaggle dataset and signal media. Details of the datasets can be found in

Section 5.2.1. The next section describes the results when I compare the proposed combination with other datasets and other classifiers, but in the same domain.

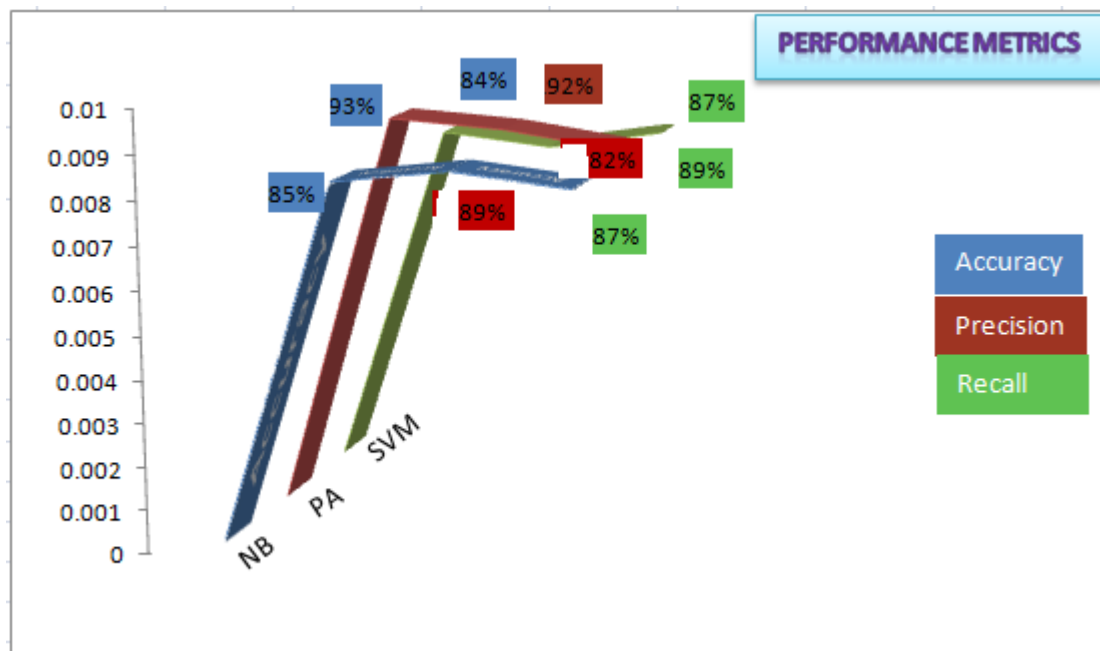


Figure 5.32: Performance metrics

The detailed comparison of models is discussed in Section 5.2.3. It can be observed in Table 5.4 that our proposed models perform well and achieve the highest accuracy up to 93% with Passive Aggressive, 85% with Naïve Bayes and 84% with SVM. We achieved Precision up to 92% with Passive Aggressive, 89% with Naïve Bayes and 82% with SVM and Recall up to 89% with Passive Aggressive, 87% with Naïve Bayes and 87% with SVM. The results can be seen in Figure 5.32.

Table 5.4: Accuracies after applying machine learning models

Sr. No.	Classifier	Features	Performance Metrics	Score
1	Passive Aggressive	News Articles	Accuracy	<b>93%</b>
2	Naïve Bayes	News Articles	Accuracy	<b>85%</b>
3	Support Vector Machine	News Articles	Accuracy	<b>84%</b>

Despite the significant results achieved by the proposed dataset, there is still room for improvement which is to compare our model using other fake news datasets. I compare my results with the same models e.g., NB, PA and SVM but with different datasets and different features, which are discussed below. Ott et al. (2011) used SVM with features LIWC+ bigrams and achieved up to 89% accuracy. Similarly when they changed the features and achieved 84% accuracy. On the other hand, Horne and Adali (2019) achieved 71% accuracy when they applied text-based features. It has been found that our proposed combination improves the existing performance in some categories. I further investigated and compared our results with (Feng et al., 2012) when they applied a combination of context-free grammar (CFG) and n-gram accuracy in deception detection, where they achieved 85% - 91%. Nevertheless, our presented results are better in the context of fake news detection and our proposed classifiers achieved the maximum accuracy.

Despite the results showing good generalization for the test set, a high number of misclassifications were observed for all three models, as shown with the confusion metrics in Figure 5.31. The model LR model has a higher number of correctly classified sentences, but on the other hand SVM and NB have also performed well compared to the others. It has been found that the dataset type and the size of the dataset affect the classifier performance. The results show that when using dataset 1, the performance increased only when dataset 1 has a larger number of words. Also, I found that the Passive Aggressive classifiers achieved higher performance when we increased the features. The other two datasets, Support Vector Machine (SVM) and Naïve Bayes (NB) performed well, but when we tested these algorithms with other datasets, the performance decreased. All the three classifiers showed good performance with respect to dataset 1 as it contains 18000 thousand news articles. It is observed that our approach outperforms most of the existing works as discussed above.

## 5.7 Conclusion

The research question “*What methods can be used to identify potential fake news?*” is answered. I conclude that my approach is beneficial as it helps in classifying fake news and identifying key features that can be used to detect fake news. The proposed technique suggests distinguishing fake and non-fake news articles; it is worthwhile to consider alternative machine learning methods that can examine the news in depth. The developed system with an accuracy of up to 93% proves the importance of the combination; we need to

look at other methods to detect fake news, other than simple text classification. Driven by the need for text classification in today's large amount of misinformation, in this chapter I contribute to the development of a fake news model using machine learning through natural language processing. I applied three different machine learning classifiers to two publicly available datasets. Experimental analysis based on the existing dataset shows very encouraging and improved performance. Different natural language processing techniques and machine learning methods were used to extract information from the dataset, combined with appropriate classification models to end up performing well with the machine learning approach. Fake content producers use different techniques to hide it and there is a possibility to mislead the readers.

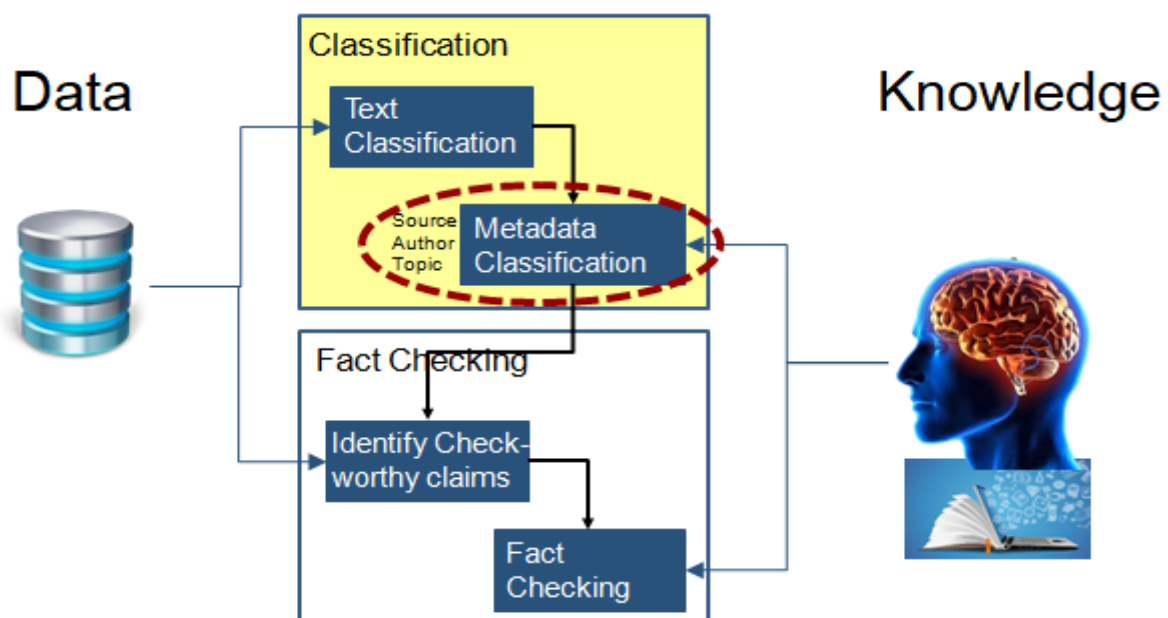


Figure 5.33: Metadata classification

Classification of news is a complex task even if we follow standard procedures, each single piece of news has different characteristics. Considering the above scenario, we can achieve good results by integrating text-based techniques and machine learning models. For fake news detection, we can add many other clues/known facts that can help us in detecting the news status. These features can be the source of the news, topic, associated URLs, publishing medium, geographical location, year of publication and others. Classification is not sufficient in detecting fake news (Ruchansky et al., 2017) because classification methods do not

provide an explanation and cannot compare the news with known facts. A model is only as good as the data, but due to the amount of data freely available, data scientists do not pay much attention to collecting this data. I have presented a state-of-the-art block diagram that represents the combination of data (Text classification) and knowledge (Fact-checking). As I discussed earlier the important open problem is the unavailability of a gold standard dataset and a predefined benchmark as well as the collection of large amounts of datasets with fake articles. So based on the points I have highlighted, it's fair to say that in the age of Big Data the problem still has not received the attention it deserved. So keeping in mind that classification can be used to separate fake text from non-fake our goal is to develop a method specialized for fake news (Samonte, 2018). I have proposed a classification approach for detecting fake news but we need an approach that examines the news and compares it to known facts. To compare the news with known facts we need to develop fact-checking applications. Identifying check-worthy statements is a subtask in fact-checking that reduces the burden on fact-checkers during fact-checking. Before moving to fact-checking we need to identify the claims that are potentially fake and can help in fact-checking. It could be more interesting if metadata classification is performed on debates and speeches which verify the factuality of the statement with source, author and topic-wise classification.

The next chapter describes the identification of check-worthy statements, which also includes the context around the statement and provides answers to research question 3.



## 6 Fact-checking: Identification of Check-Worthy Statements

In addition to the detection of fake news through classification described in the previous chapter, the other major module of fake news detection is fact-checking. The identification of check-worthy statements is a subtask in the fact-checking process. The following research question is answered in this chapter after introducing related work, the proposed approach and methodology.

***RQ 3: How can check-worthy statements for fact checking be automatically identified?***

In this chapter, I briefly explain the feature extraction phase and how context is modeled. Considering the previous work and similar applications discussed in Section 2.3.3, the extracted features are suitable for the task and contribute to it when fact-checking is performed. Check-worthy statements would reduce the time and effort required to perform fact-checking.

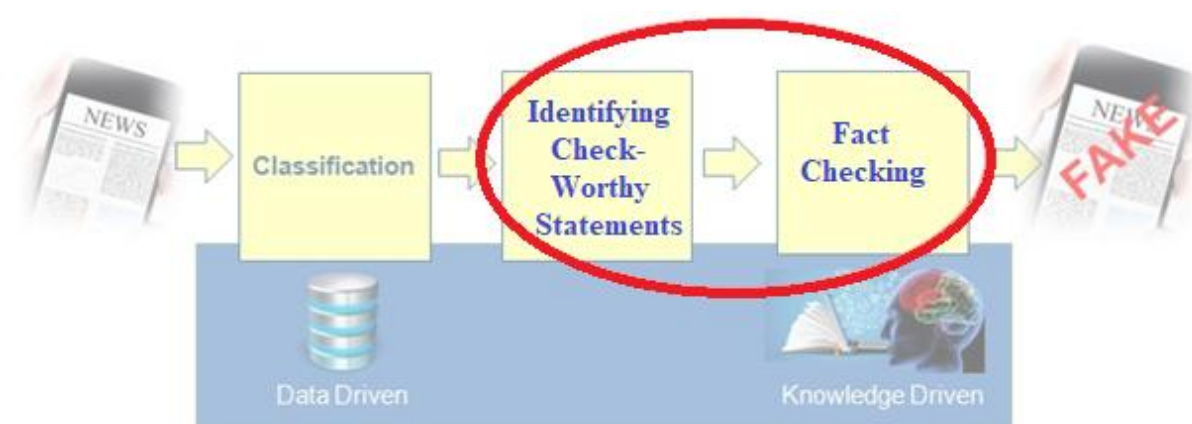


Figure 6.34: Proposed diagram for check-worthy statements (General View)

Fact-checking consists of identifying check-worthy statements and fact-checking as shown in Figures 6.34 and 6.35. I briefly describe how to identify check-worthy claims in this chapter.

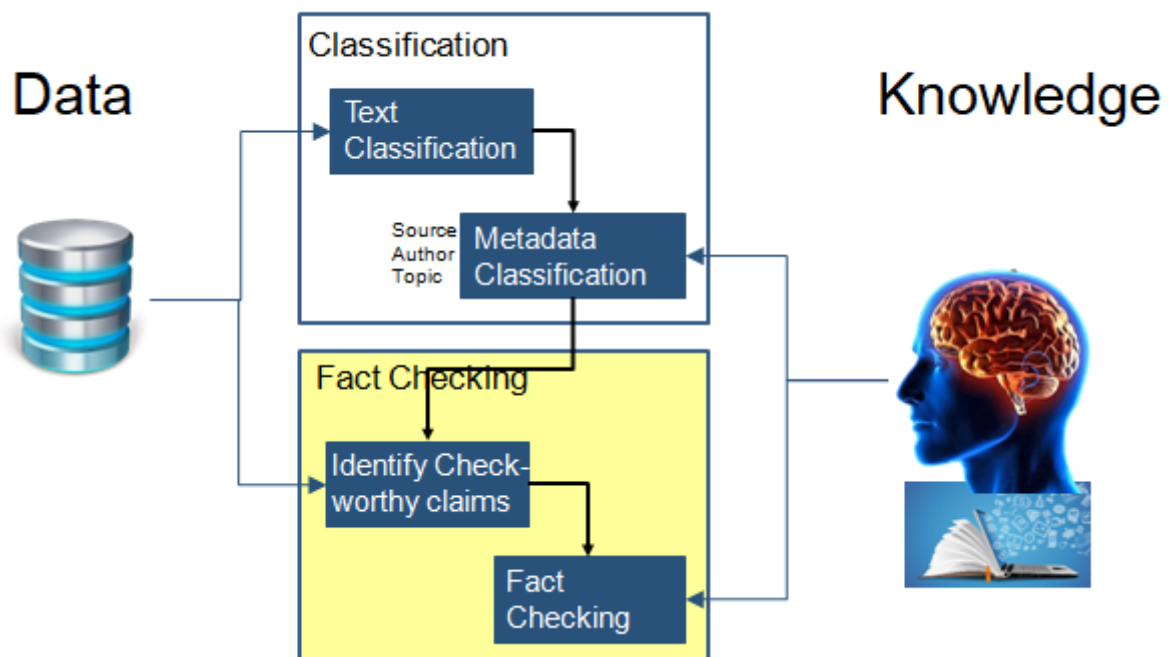


Figure 6.35: Proposed diagram for fact-checking (Inner View)

## 6.1 Problem Statement

Fact-checking is an intellectually demanding and time-consuming process, and with today's huge amount of information, manual fact-checking simply cannot keep up (Wu et al., 2014). On the other hand, despite great efforts from researchers we still do not have automated and context-aware fact-checking engines that are trustworthy enough to replace human fact-checkers (Sarr and Sall, 2017). This problem is much more evident nowadays, where an enormous amount of information is rapidly spread across the globe and many people see fake news stories that they believe (Silverman, 2016). There is a time gap between the moment the statement is made and when the fact-check is finally published; this can also lead to many statements going unchecked (Hassan et al., 2017). Translating the operations performed by human fact-checkers into program code or rules is difficult and presents many challenges, especially because these operations vary from case to case (Wu et al., 2014). Political claims are an integral part of media coverage of political news. In political fact-checking the main focus is on the accuracy of the information and the statements made by politicians. The goal is to prevent the repetition of statements when news organizations report them. Fact-checkers must be accurate and unbiased (Fact Check, 2015, PolitiFact, 2015 and The Washington Post,

2013). Some fact-checking organizations use graphical representations to verify the accuracy of claims such as PolitiFact, The Washington Post and Truth-O-Meter. Some other fact-checking organizations feel that fact-checking through graphical representation is not fully correct, so they add subjectivity to check the facts of the news (FactCheck.org 2012). In the United States of America, fact-checking organizations primarily target political claims, but internet hoaxes, urban legends, social media memes and other statements are also subjected to fact-checking. Fact-checkers use a variety of techniques to fact-check. The techniques depend on different criteria such as selection of the topic, analysis, evaluation and subsequent judgment. Snopes.com was the first fact-checking organization founded in 1994 after the 1992 U.S. presidential election. FactCheck was founded in 2003 before the 2004 election. The Washington Post and PolitiFact were founded in 2007 before the 2008 elections (Reporters Lab 2015). In the next chapter, I will discuss a complete overview of the existing fact-checking organizations that are currently operating. They point out that formalizing the intuitions of fact-checkers in assessing quality of statements is not an easy task. Sarr and Sall (2017) mention some of the challenges such as the subjectivity of reliability, quality of data, semantics and the identification of factual claims. End-to-end fact-checking systems are not trusted, but the fact-checking process can be divided into subtasks, Hassan et al. (2015) presented an approach for detecting check-worthy statements; one of the subtasks in the process of fact-checking, that reduces the time required for fact-checking. They acknowledge the need for context around a statement, but do not implement it in their work.

## 6.2 Identification of Check-Worthy Statements

Fact-checking is a multi-step process that begins with the extraction of check-worthy statements (Riedel, 2014). Manual fact-checking has proven to be very time consuming and slow, so there is a need for a method that speeds up the process. Existing fake news systems are based on predictive models that simply classify whether the news is fake or not (see also Chapter 5). Fact-checking can take into account many aspects of news, such as content, time, author, source and location. In this research I focus on the content that is the factual statements contained in the news.

Not all statements in a news are check-worthy, but only a small subset of them. Most factual statements contain facts that are not important or uninteresting for the general public to fact-

check. A statement must meet three conditions to be considered check worthy (Hassan et al., 2015):

- It should be factual, and not represent an opinion.
- It should be interesting to the general public.
- It should be verifiable.

Table 6.5 shows an example of check-worthy and non-check-worthy statements. The first statement is considered not check worthy, as it expresses an opinion rather than a fact; it is also not possible to check. In contrast, the following two statements are check worthy. They contain facts that are check worthy and interesting for the public.

Table 6.5: US Presidential debate check-worthy statements example

I built an unbelievable company.	<i>Not check-worthy</i>
You've taken business bankruptcy six times.	<i>Check-worthy</i>
Murders are up.	<i>Check-worthy</i>

Different research groups have worked on checking claims through automated methods (Castillo, Mendoza, & Poblete, 2011b; Karadzhov, Nakov, Márquez, Cedeño, & Koychev, 2017a; Rubin et al., 2016b; Zubiaga et al., 2018). Pepa et al. (2019) investigated steps involved in the fact-checking process, as shown below in Figure 6.36.

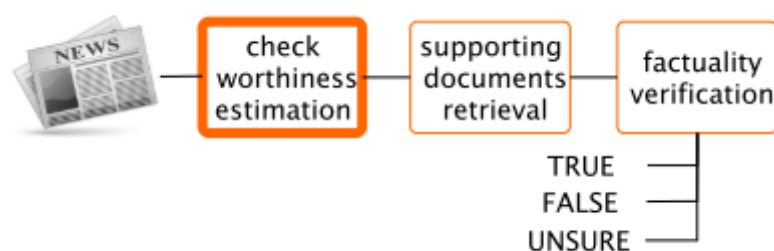


Figure 6.36: Information verification pipeline (Pepa et al., 2019)

I have already mentioned that identifying check-worthy claims is part of fact-checking. In the following sections, I discuss in detail the identification of check-worthy claims, challenges and possibilities that help us to identify potential claims that we can use for fact-checking to save time and effort. Figure 6.37 shows the focus point in the next sections.

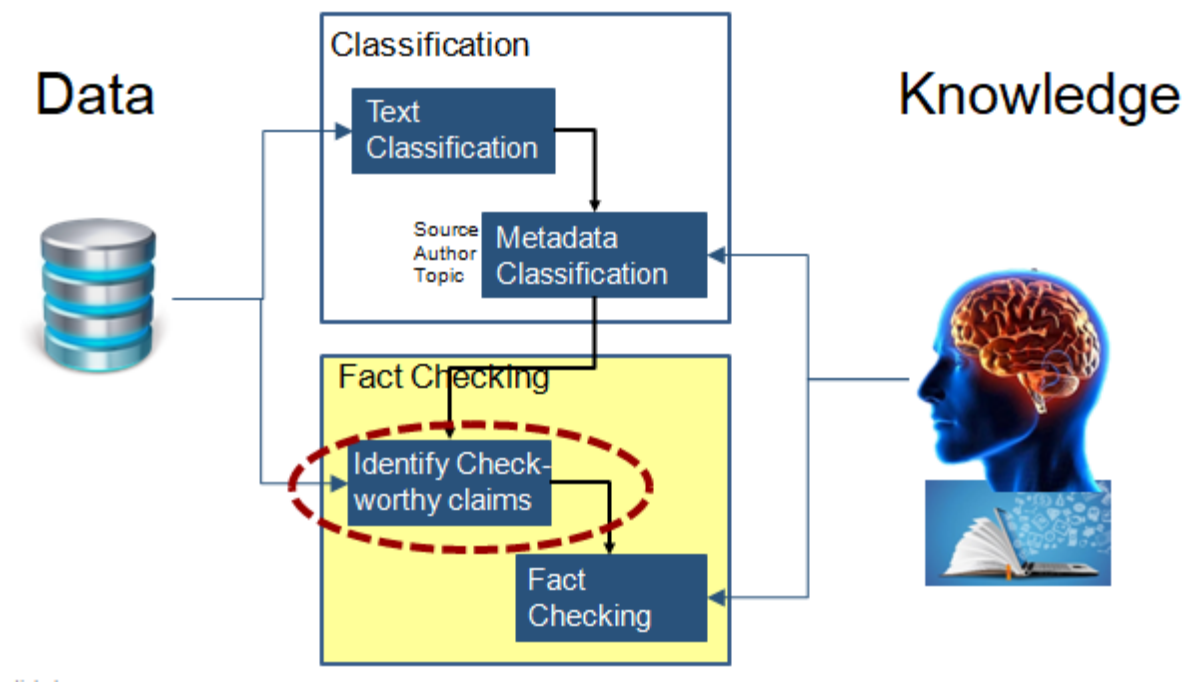


Figure 6.37: Proposed diagram for identification of check-worthy claims

A previous work related to my proposed work for check-worthy statements is that of (Hassan et al., 2015). They presented the first end-to-end system called ClaimBuster which takes the sentences as input and assigns them a value between 0 and 1 depending on how worthy they are for fact-checking. The ClaimBuster dataset was annotated with the guidelines obtained from domain experts rather than the real websites. They then added two websites and evaluated them against CNN (Hassan et al., 2017). Ennals et al. (2010a) focused on linguistic cues of disagreement between the author of the claim and people's beliefs. They proposed a classifier that assigned the pattern using the text, and for the evaluation, they obtained the dataset directly from the web. Le et al. (2016) used a convolutional neural network as the problem was the Bag of Words due to overlap among words. They assigned tags for each named entity to represent, for example person, location and organization. I focused on facts and other similarity measures that are useful for identification of check-worthy statements.

Looking at related work on distinguishing facts from non-facts provides clues to interesting features and models that I can use in my work. Several previous works have attempted to separate factual and opinionated text, or subjective and objective text. Yu and Hatzivassiloglou (2003) separate fact from opinion, at both sentence and document levels. At the sentence level, they consider three different methods: the similarity approach, an approach that uses a Naive Bayes classifier and another that uses multiple Naive Bayes classifiers; they achieved up to 91% precision and recall. They also investigated an automatic method for assigning polarity information to individual words and sentences, by distinguishing between positive, negative, and neutral opinions. Stepinski and Mittal (2008) classified news articles as either fact or opinion. Each sentence was classified as fact or opinion using the Passive Aggressive algorithm trained on unigram, bigram, and trigram features. The total score was calculated based on these sentence labels.

Wiebe and Riloff (2010) worked on developing subjective and objective classifiers at the sentence level. They worked with unannotated data. For this reason, they first classify the sentence with rule-based classifiers and generate training data for learning algorithms used later. Naive Bayes classifiers were trained with these patterns and other features including subjective cues and Part-of-Speech features. The classifier obtained after retraining on the new training set had subjective precision and recall of 71.3% and 86.3% respectively. The objective precision and recall were 77.5% and 57.5% respectively.

Yang and Cardie (2015) proposed a contextual method for sentence-level sentiment analysis. Their method uses both unlabeled and labeled data, to improve learning. They study sentence-level binary sentiment, but also a third class, neutral. They incorporate rich discourse information at both global and local levels and use a rich set of contextual posterior constraints for sentence-level sentiment analysis using lexical and discourse knowledge.

### 6.3 Methodology for Identifying Check-Worthy Statements

The methodology for identifying check-worthy statements reuses work of the master thesis of Balla (2019), which was the basis for a joint paper (Ahmed, Balla, Hinkelmann, Corradini, 2020). In this section, I summarize the results of this work. First, I am going to explain how the dataset was collected and used in this work as well as the different analyses performed. Later I discuss the features extracted and how the context was modeled.

### 6.3.1 Dataset

A dataset containing information from the 2016 U.S. Presidential Election and the following year's election was created using an approach similar to that used in the work of Patwari et al. (2017) and Gencheva et al. (2017). The speeches and debates were collected by consulting different fact-checking organizations, including CNN<sup>57</sup>, FactCheck.org<sup>58</sup>, NPR<sup>59</sup>, PolitiFact<sup>60</sup>, The New York Times<sup>61</sup> and ABC News<sup>62</sup>. Because of the sentence level review, transcripts must be broken down into sentences. A sentence is check worthy until it has been validated by at least one of the fact-checking organizations. In some cases, the statements validated by the organizations were expanded into two or more sentences; as a result, the corresponding sentences were noted as check worthy (Gencheva et al., 2017).

The sentences in the transcripts of the debates were considerably small and contained ill-defined sentences (see example in Table 6.6), which were manually deleted. The number of sentences decreased from 9187 to 8804, but as all of these sentences were not check worthy, there was no change in the number of check-worthy sentences.

Table 6.6: Examples of ill-defined sentences (Balla 2019)

ID	Speaker	Sentence
110	Clinton	Well, let me.....
111	Sanders	We have.....
112	Clinton	Let me just say .....
113	Sanders	Inaudible....
114	Clinton	Let me-let me say....

Out of these 8804 sentences, only 647 are check worthy, which is about 7% of the total sentences (see Figure 6.38). This imbalance is similar to previous research (Gencheva et al., 2017; Patwari et al., 2017).

<sup>57</sup> <https://edition.cnn.com/specials/politics/fact-check-politics>

<sup>58</sup> <https://transcripts.factcheck.org/>

<sup>59</sup> <https://www.npr.org/sections/politics-fact-check>

<sup>60</sup> <https://www.politifact.com>

<sup>61</sup> <https://www.nytimes.com/spotlight/fact-checks>

<sup>62</sup> <https://abcnews.go.com/Politics>

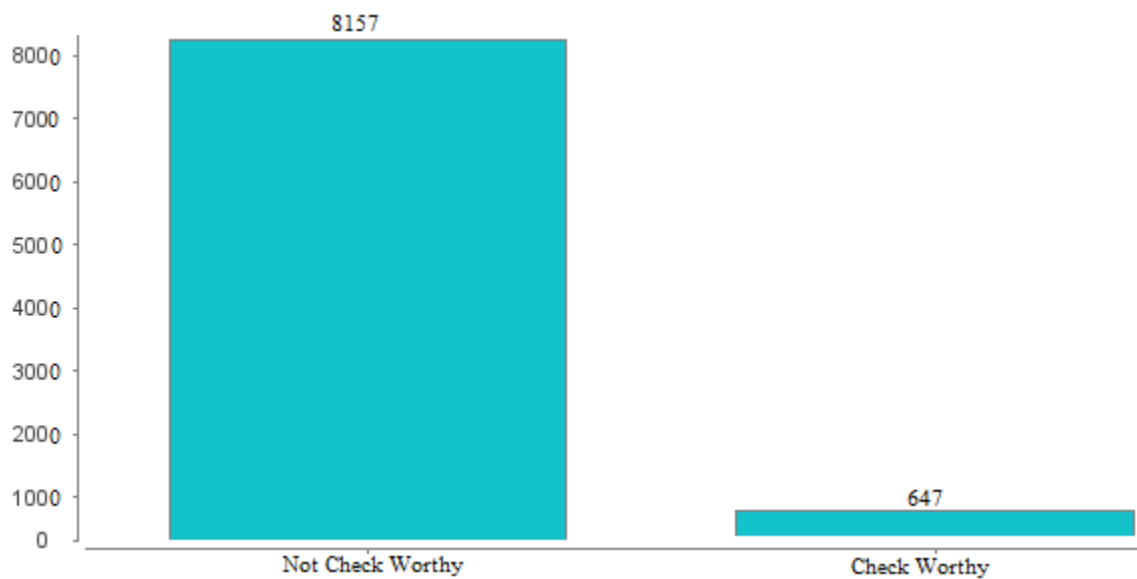


Figure 6.38: Class distribution of sentences in the dataset

The dataset is structured and each row in a file, depending on whether it is a speech or a debate, consists of the ID of a sentence, the speaker, the sentence text and the binary classification, of whether the sentence is check worthy or not. As can be seen in Figure 6.39, almost half of the sentences come from the same speaker as fact-checking organizations focus on individuals who have a greater public interest because they have fact-checked transcripts available for fact-checking.

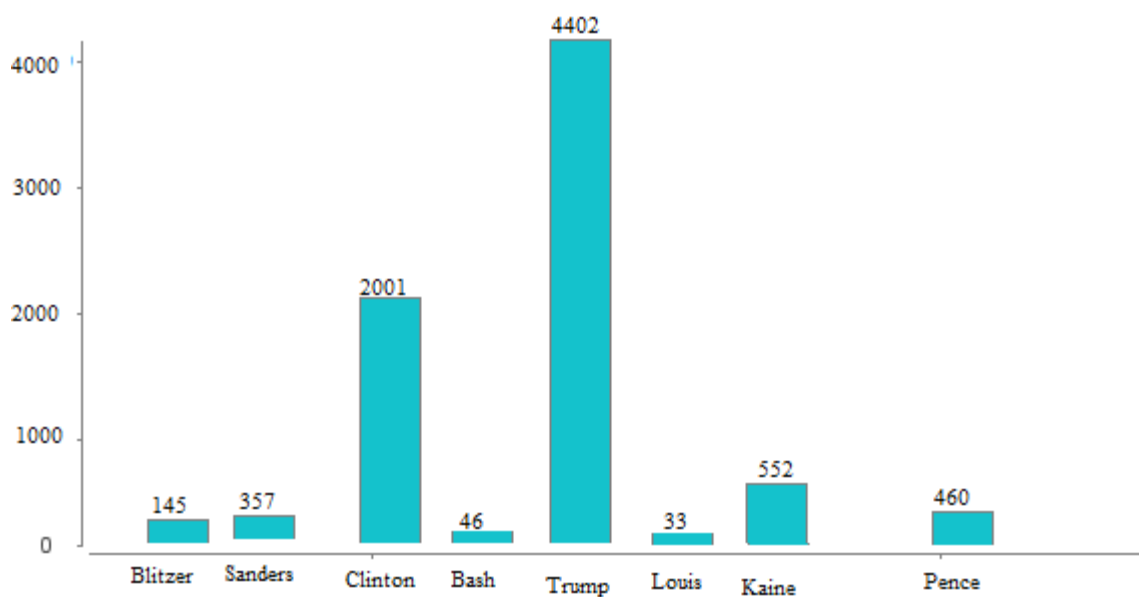


Figure 6.39: Number of sentences by each speaker



### 6.3.2 Feature Analysis

To identify the most effective feature for classification, several features were extracted from sentences. This is used to distinguish between sentences that are check worthy and non check worthy. Table 6.7 gives an overview of the features analyzed. The section summarizes the importance of the features in identifying check-worthy statements.

Table 6.7: Overview of extracted features from the target sentence (Total 1536)

Category	Number of features
Bag of Words	941
W2V Sentences Weighted Embeddings	300
Named Entities	172
Part of speech (POS) tags	45
Syntactic Dependency Parsing	45
Topic	30
Sentiment	1
Length	1
Speaker	1

- **Length:** Check-worthy sentences are longer than non-check-worthy sentences in terms of the both number of characters and the number of words.
- **Sentiment:** Based on the distribution of the sentiments of the sentences, it was analyzed that most of the check-worthy sentences have a negative sentiment compared to non-check-worthy sentences.
- **Named Entities:** Check-worthy claims contain named entities such as countries, organizations and individuals (Gencheva et al., 2017). When comparing the entity types, it was found that some entity types occur more frequently in check-worthy sentences than others (see Table 6.8)

Table 6.8: Entity types detected in check-worthy sentences and the whole dataset

Entity Type	Check Worthy	Total
Company	12 (16%)	75
Crime	6 (21%)	29
Drug	9 (60%)	15
Location	142 (9%)	1483
Country	81 (11%)	706
Organization	58 (12%)	485
Person	172 (11%)	1613
Quantity	101 (17%)	588

- **Part-of-Speech (POS):** Tags have already proven successful in similar tasks (Chenlo and Losada, 2014; Hassan, Li and Tremayne, 2015). POS labels assign each word its comparative linguistic category in a sentence. Words can have different meanings based on their usage in different parts of speech. This provides information about a word and the surrounding words. This can be used to analyze the context of a word to improve the identification of check-worthy statements (see Section 6.7).
- **Sentence Embedding:** Word embeddings were computed to represent the sentences in a low dimensional space where similar words are close to each other. To obtain the importance of a word in a sentence, TF-IDF (Ramos et al, 1999) was considered in combination with word2vec (Mikolov et al., 2013). TF-IDF reflects that the more documents contain the word, the less valuable it is to distinguish a particular document, or sentence in this work. An embedded word vector is created by word2vec (Mikolov et al. 2013) which uses either Continuous Bag-of-Words (CBOW) or the Continuous Skip-Gram model. CBOW predicts the current word based on the context window of the surrounding words, while the skip-gram model predicts the context window of the surrounding words based on the current word.

By combining TF-IDF with word2vec, a weighted vector representation is created for each sentence, which in this way also reduces the weight of the most frequently used

words and captures which words are most important in a sentence. In addition, the sentence was also represented as a Bag-of-Words (BoW), which has been shown to be helpful for classification in previous work (Hassan et al., 2015; Patwari et al., 2017). BoW representations do not retain any information about word order or grammar. Instead a sentence is represented as a bag of its words.

- **Syntactic Dependency Parsing:** Although similar to Part-of-Speech tags, syntactic dependency parsing can capture more complex phenomena in the speech. Syntactic dependency parsing has been used to encode the syntactic structure of a sentence. While POS cannot capture the grammatical relationships between words, syntactic dependency parsing can map the dependency of each word on the sentence structure, allowing the grammatical relationships to be determined for each word. The internal structure of dependency parsing consists of directed relations between lexical items in the sentence (see Figure 6.40).

*“She wants 550 percent more people than Barack Obama, and he has thousands and thousands of people.”*

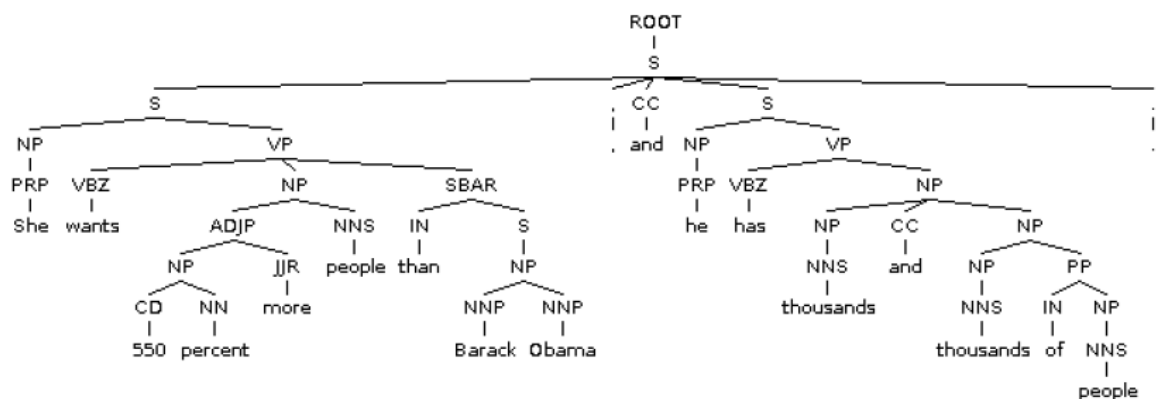


Figure 6.40: Syntactic dependence parse tree of the example sentence

- **Topic:** Considering that topics can give a clue in classifying the sentences, we extracted the topics of the sentences using the LDA topic model (Blei, Ng and Jordan, 2003) as presented by (Gencheva et al., 2017; Patwari et al., 2017). The topic of a sentence can indicate whether it is worth checking or not.
- **Speaker of Sentences:** The speaker of the sentence is another feature that is also extracted. The assumption is that if sentences from a particular speaker are often

considered to be check-worthy, this is an indicator that sentences from that speaker have a higher probability of being check worthy.

## 6.4 Using Context to identify Check-Worthy Statements

Many sentences are ambiguous and a decision can only be made after considering the context around them. By considering only the text of the targeted sentence, it is sometimes difficult for a human fact-checker to assign a label. Hassan *et al.* (2017) have recognized this need in their work, and their system has built functionality to preview sentences when needed. However, they do not model context in their work. The same logic was followed to model context in the form of features for this work. Apart from the features extracted for the target sentence, features of two previous and two following sentences are also extracted from the dataset. In this way, we can create a context window around the sentence. Figure 6.41 shows an example from the dataset. The target sentence is a check-worthy sentence.

If we look at the sentence in isolation, the context is not clear, but we can understand it better by looking at the sentence above it. Therefore, including features from this sentence would also help in the classification. As for the features, the sentence above has a negative sentiment and five named entities are found (two locations, one organization, two quantities) while the target sentence does not contain any named entity.

We have spent more than \$7 trillion in fighting wars in the Middle East.

As a candidate for President, I loudly pledged a new approach.

Great nations do not fight endless wars. Context Window

ISIS controlled more than 20,000 square miles in Iraq and Syria just two years ago.

**Today, we have liberated virtually all of the territory from...** ← Target Sentence

Now, as we work with our allies to destroy the remnants of ISIS.....

I have also accelerated our negotiations to reach...

The opposing side is also very happy to be negotiating.

Our troops have fought with unmatched valour.

Figure 6.41: Context window example from the dataset

Although features from surrounding sentences support the classification, the number of extracted features is kept small in order not to overwhelm the model. A total of 20 features from surrounding sentences are included in the context windows, five features for each sentence.

## 6.5 Learning from an Imbalanced Dataset

Learning from an imbalanced dataset is difficult because the classifier tends to favor the majority class, while often considering the minority class as noise in the data. As mentioned in Section 6.3.1, the dataset is very unbalanced, with the positive class, check-worthy sentences, accounting for 7% of the whole dataset.

- **Class Weight:** Class weights to classification models allow adding weights to the two classes proportional to the number of samples. Since the dataset is small in my work, it is crucial to see how the weight option allows us to create unbiased training data without losing training data. Resampling methods outperformed this approach.
- **Resampling the Dataset:** Another approach to dealing with imbalanced learning is either over-sampling minority class instances or under-sampling majority classes (Sun, Lim, & Liu, 2009). Even though this approach allows to obtaining a balanced dataset, there are certain drawbacks.
- **Model Overfitting:** Model overfitting leads to redundancy of sentences when selecting from a relatively small subset, with only 647 sentences compared to 8157 sentences of the other class.
- **Losing Informative Sentences:** Random undersampling can result in the loss of valuable information that contains differences in the two classes.

These drawbacks can be overcome by advanced resampling methods, as explained below:

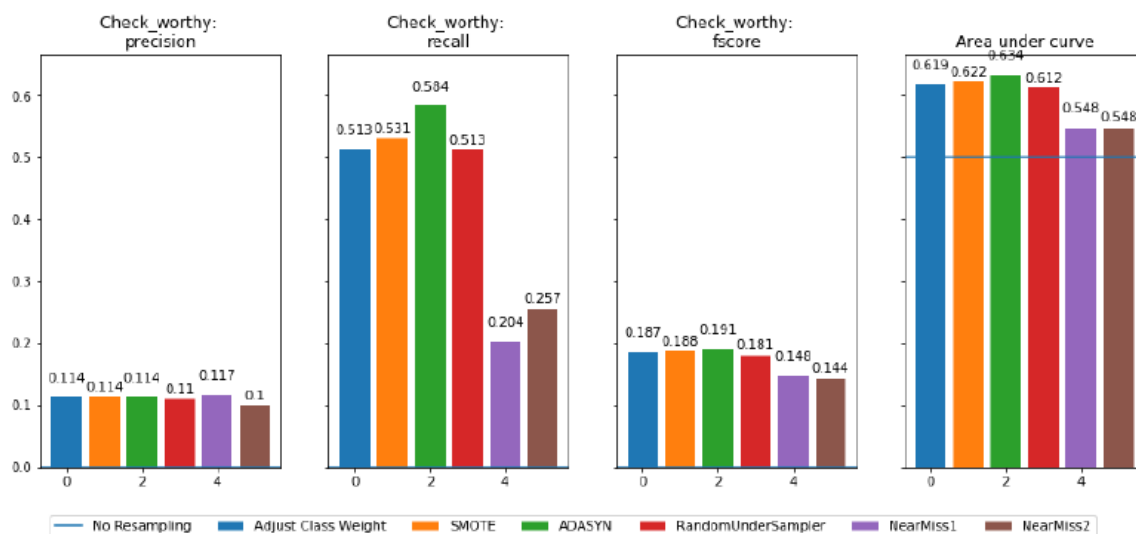


Figure 6.42: Initial experiments with resampling methods (Logistic Regression)

Since we knew the dataset was small, we need to oversample instead of under-sampling the data. Oversampling methods showed the best results and are more suitable for this particular case. In an already small dataset, under-sampling could lead to the loss of important non-check-worthy sentences that could give important clues about the differences between check-worthy and not check-worthy sentences. A well-known example of oversampling is the Synthetic Minority Oversampling Technique, SMOTE (Chawla, Bowyer, Hall, & Kegelmeyer, 2002).

## 6.6 Models

In the present study, the problem was modeled as a binary text classification task, in which sentences are classified into check-worthy and not check-worthy sentences. To select the appropriate model and tune the parameters, several experiments were conducted with different models. We experimented with classification models that were found to be effective in related sentences classification tasks and gave satisfactory results. Some of the models did not give good results and were discarded (e.g. Logistic Regression). However, Support Vector Machine and Feed Forward Neural Network provided promising results and were continued as part of our experiment.

### 6.6.1 Logistic Regression and SVM with Linear Kernel

Logistic Regression (LR) models and Support Vector Machines (SVMs) models with the linear kernel are effectively used in linearly separable problems. Support vector machines (SVMs) are suitable for learning in text classification situations (Joachims, 1998). The SVM ensures that a hypothesis  $h$  is found for which the lowest true error can be guaranteed. The true error of  $h$  is the probability of making an error on an unseen and randomly selected test example. SVMs are independent of the dimensionality of the feature space (Joachims, 1998). Considering a dataset of two subsets,  $X$  and  $Y$  are said to be linearly separable if there is a hyperplane  $P$  separating the subsets, such that the elements of  $X$  and  $Y$  lie on opposite sides of it (Elizondo, 2006).

I decided to start with Logistic Regression as a first step. This, as a simpler model has fewer hyperparameters that need to be tuned to see how well this model would perform in the dataset and then decide the following actions. Apart from some important differences in the application aspects of the philosophy, the performance of LR and SVM with a linear kernel is similar. SVM tries to maximize the distance between the nearest support vectors and tends to maximize the probability that a data point is classified correctly.

SVM with a linear kernel was not suitable for this task so we continued to work on SVM with the non-linear kernel. It can handle nonlinear cases and map samples in a higher-dimensional space. SVM with RBF kernel had better overall results and outperformed those with LINEAR and SIGMOID kernel. Apart from choosing the right kernel, other important parameters to decide on were the  $C$  and  $\gamma$  values for the SVM with RBF kernel. The  $C$  parameter in an SMV model tells the SVM optimization how much to avoid misclassification of each training example, so it controls the misclassification cost. If can lead to underfitting, if it is large, it can lead to overfitting, so it is important to choose an appropriate  $C$  value for the specific case. The parameter  $\gamma$  is used as a similarity measure between two points. A small  $\gamma$  parameter can cause the model to be very constrained as two points are considered similar if they are close to each other.

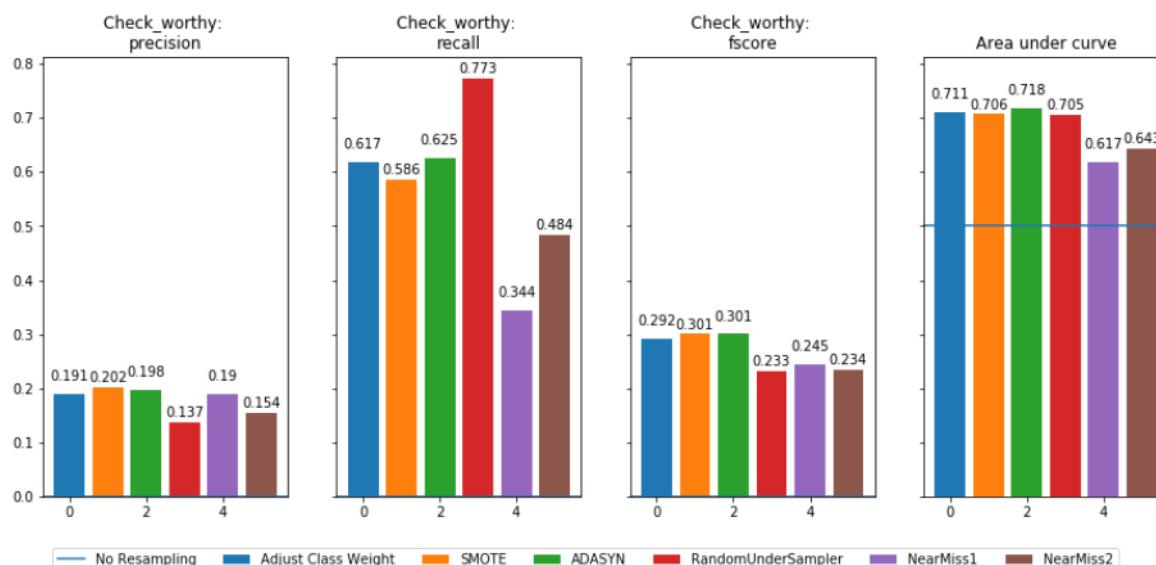


Figure 6.43: Initial SVM experiments metrics score for check-worthy claims

On the other hand, a high  $\gamma$  value would lead to overfitting despite the chosen  $C$  value. SVM showed the best results in cross-validation with  $C=0.6$  and  $\gamma=0.001$  as parameters.

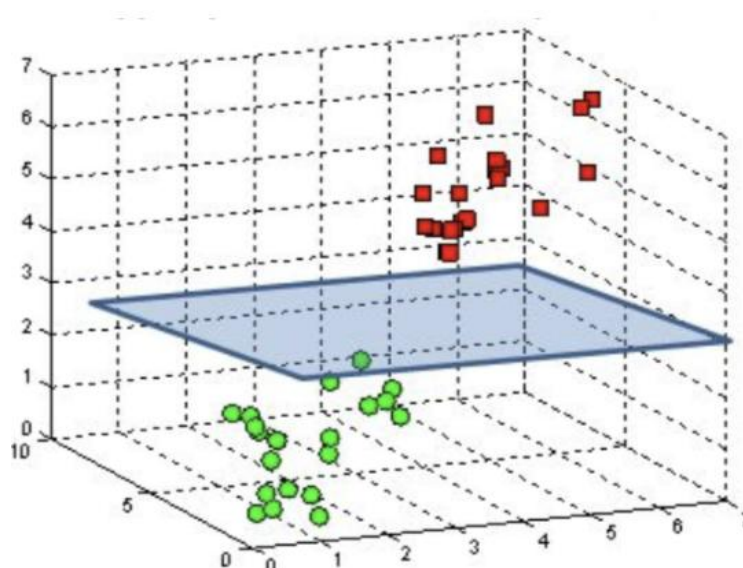


Figure 6.44: Initial SVM experiments on Hyperlane

## 6.6.2 Feed Forward Neural Network

Feed Forward Neural Networks take a fixed input and feed it forward through the network to produce an output without generating cycles. Feed Forward Neural Network (FNN) is commonly used in text classification tasks. Bengio et al. (2003) presented a Neural Network



Language Model (NNLM) to deal with the challenges of n-gram language models. Each neuron computes the weighted sum of all its inputs and applies the  $f$  activation function to it. For the activation function, we considered Rectified Linear Units (ReLU) (Nair and Hinton, 2010) and Hyperbolic Tangent function (Anastassiou, 2011), where ReLU had the better performance.

It does not involve expensive operations such as divisions and exponentials, as is the case with the Tanh function. We used ReLU for the activation function and stochastic optimization with Stochastic Gradient Descent (SGD) (Bottou, 2010) which performed better than the initial experiments with Adam (Kingma & Ba, 2015).

## 6.7 Results

Support Vector Machine (SVM) and Feed Forward Neural Networks (FNN) give better results as compared to the other classification methods. We trained the models on 4 debates and 8 speeches with a total of 7073 sentences. These trained models were used to generate predictions for the test set, consisting of 1 debate and 2 speeches, with 1731 sentences, which is about 20% of the dataset.

The proportion of check-worthy sentences is about 7% of the total number of sentences in both the training and test sets, which is the same as the proportion in the total dataset. During the training of the Feed Forward Neural Network (FNN) model, loss and accuracy values were obtained which can be seen in the following figure.

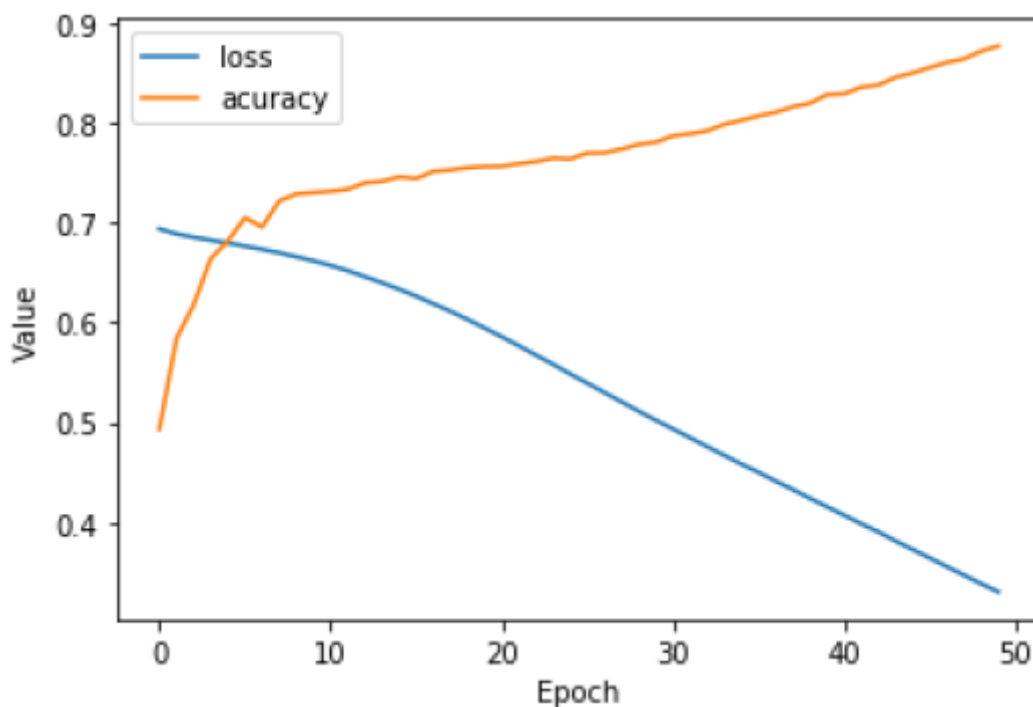


Figure 6.45: Loss and accuracy values during the training of the final FNN model

Support Vector Machine (SVM) and Feed Forward Neural Network (FNN) performed better in predicting not check-worthy sentences in all metrics.

For all experiments we reported the models with the best performance on the test set, using the F1 score as the primary evaluation metric, but also presenting results on the previously mentioned metrics. In cross-validation, we found that improved recall was accompanied by a decrease in precision. Although recall is important, increasing precision increases the likelihood of positive which is undesirable.

### 6.7.1 Final Test Set Results

In Table 6.9 we can see the results for the SVM and FNN models. The sentences are considered check worthy only if they achieve a score of 0.5 or more. The FNN has a slightly higher F1 score compared to SVM, with only 1.2%. The recall score of the SVM model was the highest, 13% higher than FNN, but with 7.1% lower precision than FNN.

Table 6.9: Results for SVM and FNN model

	Precision	Recall	F1	Accuracy
SVM	0.220	0.461	0.298	0.807
FNN	0.291	0.331	0.310	0.869
FNN (only embeddings)	0.163	0.591	0.256	0.694

Although, as mentioned earlier, recall is very important, it is not desirable to increase it at the expense of very low precision. The Feed Forward Neural Network (FNN), on the other hand, also achieved a higher recall than precision, but with a small difference from the latter. These values make the Feed Forward Neural Network (FNN) the better model overall. In Table 6.10, I have presented the scores from the classification of non-check-worthy sentences.

Table 6.10: Classification results for non-check-worthy sentences

	Precision-NCW	Recall-NCW	F1-NCW
SVM	0.941	0.841	0.887
FNN	0.934	0.921	0.928

As mentioned before, the results of this class are very high, since it represents more than 90% of the dataset, which is the reason for the high accuracy achieved by the classifiers.

Despite the results showing good generalization for the test set, I observed a high number of misclassifications in both models, as shown by the confusion metrics in Figure 6.46.

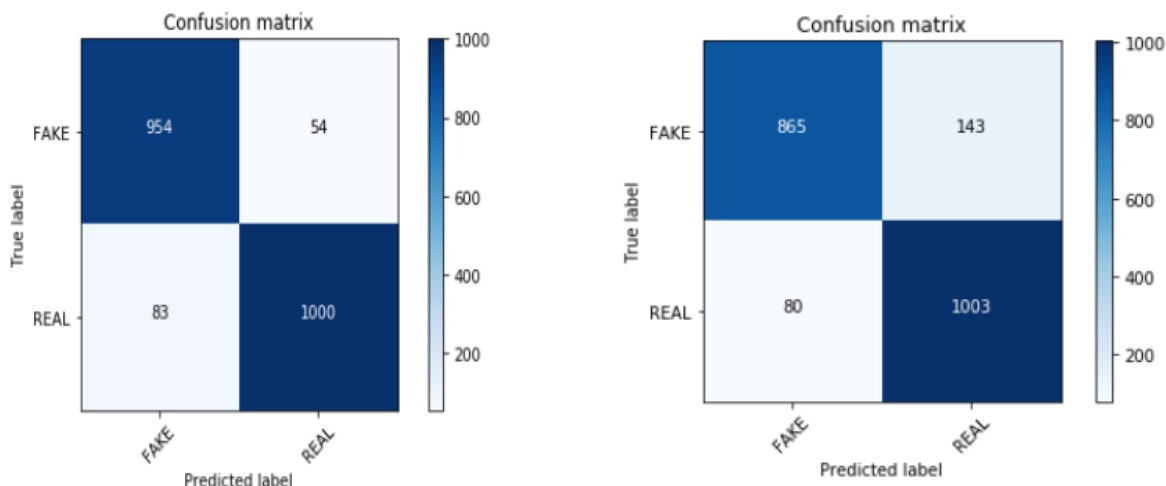


Figure 6.46: Confusion matrix for FNN and SVM respectively

The SVM model had a higher number of correctly classified check-worthy sentences, but on the other hand, has almost twice as many false positives sentences as the FNN model. To better understand this behaviour, we reviewed the sentences that were misclassified by the FNN model. It was observed that subsets of the misclassifications were caused by inconsistencies originating from fact-checking organizations.

Fact-checking organizations check statements based on a variety of factors, so even if a statement is check worthy it may be skipped by the fact-checking process. Example sentences for this case are shown below: Similar sentences are annotated in the dataset as check worthy. However, these sentences are annotated as not check-worthy but were predicted by the classifier to be check worthy. This type of misclassifications results from the lack of a formal definition regarding check-worthy claims from fact-checking organizations. Their decision in these cases is subjective or related to the editorial line of the organization. The results shown in Table 6.10 are for all sentences in the test set, and Table 6.11 shows the results for each of the speeches and the debate. One of the speeches scores higher compared to other debates and speeches. It is believed that the reason for this is that about half of the total sentences in the dataset are from the same speaker, so the classifier can predict these sentences more correctly. However, both the debate and the other speech also achieve a good result, showing that the model can capture information from the trained sentences and generalize well for new unseen sentences.

Table 6.11: Metrics scores for each test file separately

	Precision	Recall	F1	Accuracy
Presidential Speech	0.255	0.385	0.307	0.873
Clinton Speech	0.417	0.222	0.291	0.877
Trump Speech	0.429	0.334	0.375	0.770

Note that the dataset only contains sentences from the political domain, so the ability of the models is only tested on this domain. Figure 6.47 shows the results for all speeches.

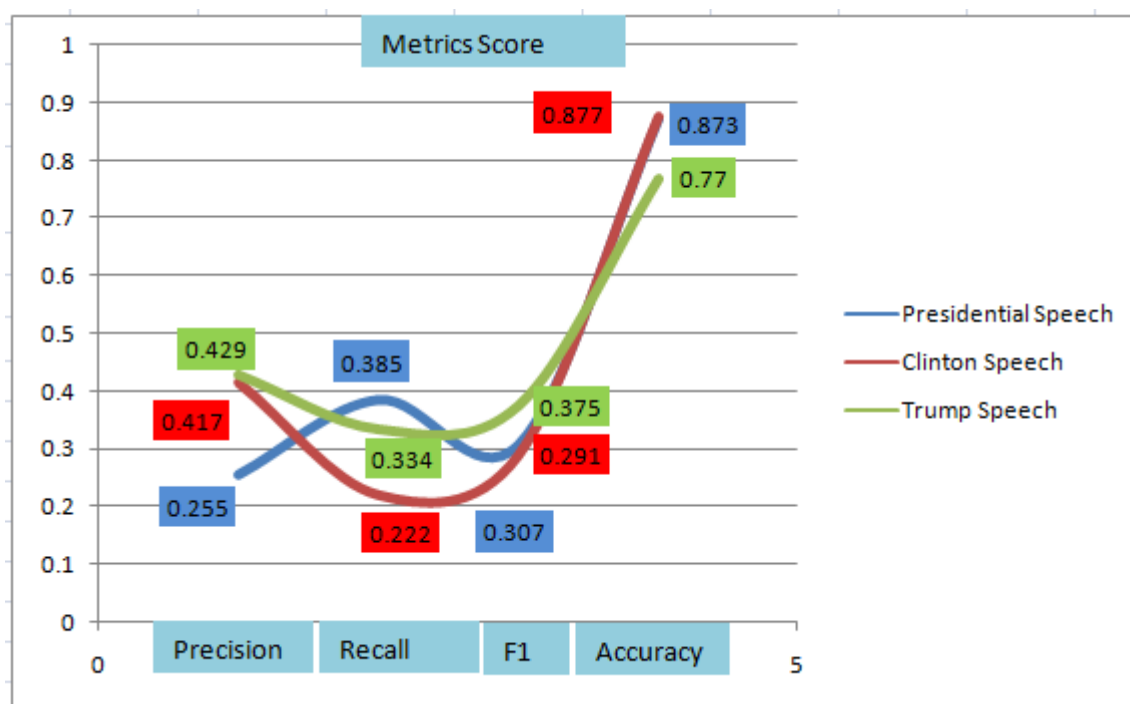


Figure 6.47: All speech results

## 6.7.2 Best Performing Features

By extracting features with the FNN model, we obtain the performance of each feature, to have a better overview which features contribute more to the classification. Table 6.12 shows the results of the FNN model with each feature when separated. Each feature type also includes the features extracted from the sentences in the context window, and not only for the target sentence. Overall, the weighted sentence embedding set of features performs better.

They achieve an F1 score 1.4 % higher than the second-best performing feature, POS tags, and 3.6 % higher than NE in third place. The good performance of embeddings, POS tags, and NE is to be expected, as these features have proven useful in text classification in works dealing with previews. Since it has been observed in the dataset that sentence length and sentiment can help to capture the differences between classes, these features also perform well, achieving F1 scores of 0.216 and 0.199, respectively. Since more than half of the sentences in the dataset are from the same speaker, this feature contributes less to the classification.

Table 6.12: Separate features scores in Feed Forward Neural Network. Ordered by the highest F1 score

Features	Precision	Recall	F1	Accuracy
Weighted embedding	0.163	0.591	0.256	0.694
Part-of-speech tags	0.162	0.481	0.242	0.733
Named Entities	0.150	0.408	0.220	0.788
Length	0.139	0.493	0.216	0.740
Sentiment	0.118	0.648	0.199	0.620
Bag-of-words	0.122	0.273	0.169	0.761
Dependency Parsing	0.191	0.18	0.167	0.892
Topic	0.095	0.662	0.167	0.517
Speaker	0.087	0.831	0.158	0.352

### 6.7.3 Context Features

The results show that the features extracted from the surrounding sentences are useful for classification, which as expected contributes significantly to identifying check-worthy statements. Table 6.13 shows the metrics for the FNN model when only features extracted from the target sentence are used, without including contexts.

Table 6.13: Performance of the FNN model without context features

	Precision	Recall	F1
FNN no context	0.239	0.310	0.271
<b>FNN+ all</b>	<b>0.291</b>	<b>0.331</b>	<b>0.310</b>

As can be seen, the FNN model using all features achieves a higher F1 score with 3.9%, but also outperforms the FNN using only features extracted from the target sentences in both precision and recall with 5.2 % and 2.1% respectively.

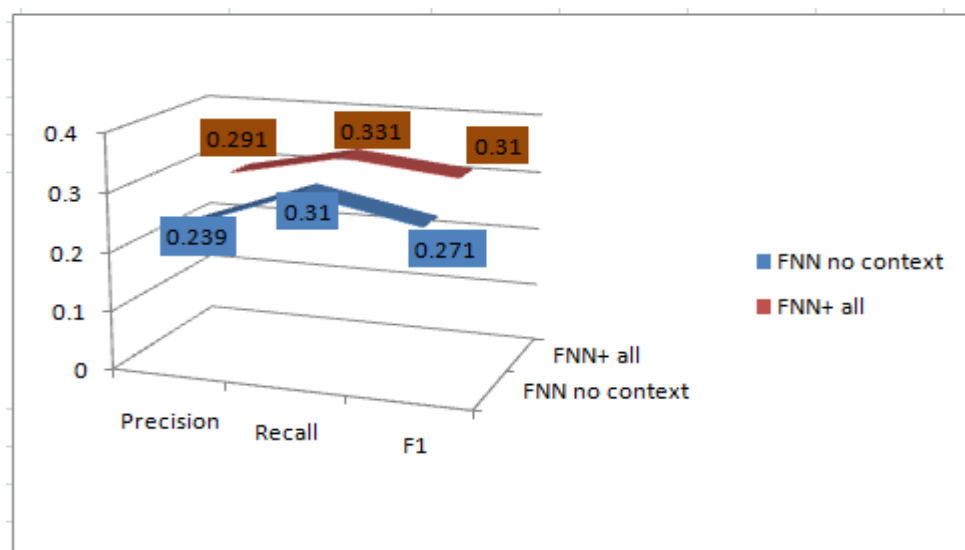


Figure 6.48: Context feature metrics results

## 6.8 Discussion

The research question “*How can check-worthy statements for fact-checking be automatically identified*” is answered. This chapter presented and evaluated an approach to detect check-worthy statements driven by the need to automate the fact-checking process given today’s large amount of misinformation. Relevant dataset were collected from different fact-checking organizations. A sentence is considered check worthy if it has been fact-checked by at least one of the fact-checking organizations. The dataset contains political debates and speeches. After collecting the dataset, different experiments were conducted to decide which features

should be extracted and how they should be combined to represent the sentences and help in their classification. For each sentence, in addition to the features extracted at the sentence level, additional features were extracted from two previous and two subsequent sentences, to form a context window around a target sentence. As expected the implementation of context features proved to be very useful in identifying check-worthy sentences. Experiments with different classification models showed that a Feed Forward Neural Network (FNN) with two hidden layers was better suited for this approach, compared to Support Vector Machines (SVMs). The hyperparameters of the model were each tuned using 4-fold cross-validation, but tested with a final unseen set of sentences. Overall both classification models showed good generalization on the test set, suggesting that the approach is reliable, but with room for improvement. Weighted sentence embeddings contribute more to the classification followed by Part-of-Speech tags and Named Entities. Classification showed significant improvement when all feature sets, were used compared to excluding context features. Considering the small dataset the number of context features was kept low to avoid overfitting, but in a larger dataset; it would be interesting to include more context features and see how the approach and classification models perform. The same feature set was extracted from all sentences without distinguishing whether they were from a debate or a speech, in order to create a general approach, that is not only based on the spoken language present in the dataset. This suggests that this feature set is also suitable for other types of datasets, even those containing text from written speech. A high number of misclassifications were observed in both models, Support Vectors Machines (SVM) and Feed Forward Neural Network (FNN). Error analysis showed that some of these misclassifications came from inconsistencies in the dataset, as different fact-checking organizations had different criteria for selecting a statement for fact-checking. These misclassifications could be avoided if the sentence is considered as check worthy or not based on a formal definition, which is yet not available. The results could be further improved if we used these check-worthy statements for sentence-level fact-checking and compared them to the known facts that are present in these statements.

In this chapter, my aim was to develop a method that could help automating both fact-checking as well as investigating the check-worthy statements. The results show that the proposed approach gives promising results. It could further improve and automate fact-checking of check-worthy claims to reduce the effort and time for fact-checkers.

The next chapter describes the investigation of the answers of research question 4.



## 7 Automated Fact-checking for Fake News Detection

This chapter addresses the problem of automatic sentence-level claim identification (fact-checking). Chapter 7 answers research question 4, examining the dataset and the proposed methodology for applying fact-checking. The development and implementation results of this chapter are presented in Chapter 8. This is in contrast to the previous modules, text classification (Chapter 5) and identification of check-worthy claims (Chapter 6), which dealt primarily with the document level rather than the sentence level. It corresponds to the demonstration phase of the DSRM (Peffer, 2006). I have already discussed the background knowledge of fact-checking and the existing fact-checking organizations in detail (see Chapter 2). The literature presented in Chapter 2 is to determine the *research problem* but here in this chapter the literature is reviewed that is relevant to deriving the *solution*. This chapter is dedicated to solving the automation challenge of this thesis, i.e., an automated fact-checking application that is also capable of searching Wikipedia and mainstream media sources on the web to fact-check a given claim. The results of this chapter and Chapter 8 provide an answer to the following research question:

***RQ 4: How can it be checked whether a statement is fact or fake?***

An important motivation for my research is to automate fact-checking. There are different ways to check the credibility of news that is fake or not. To tackle automated fact-checking, some researchers use source reliability and network structure. The major challenge in these cases is to train the model, which is impossible due to the unavailability of corpora (Hassan et al., 2015). Fake news contains information that may be false or inaccurate (Zannettou et al., 2019), and separating false from true text is a challenging and difficult task (Lazer et al., 2018b). In addition to Wikipedia, the news aggregation site Reddit.com (Mieghem, 2011) is another example of a news aggregation site that can be used as a basis for fact-checking. Rather than using a collection of known facts, crowdsourcing is an alternative approach to fact-checking where many contributors assess whether a news item is a fact or not

(Chatzimilioudis, Konstantinidis, Laoudias, & Yazti, 2012). Chatzimilioudis et al. (2012) have shown that disagreement is not noise but a signal, indicating that crowd sourcing can not only be cheaper and scalable, but also of higher quality with more information.

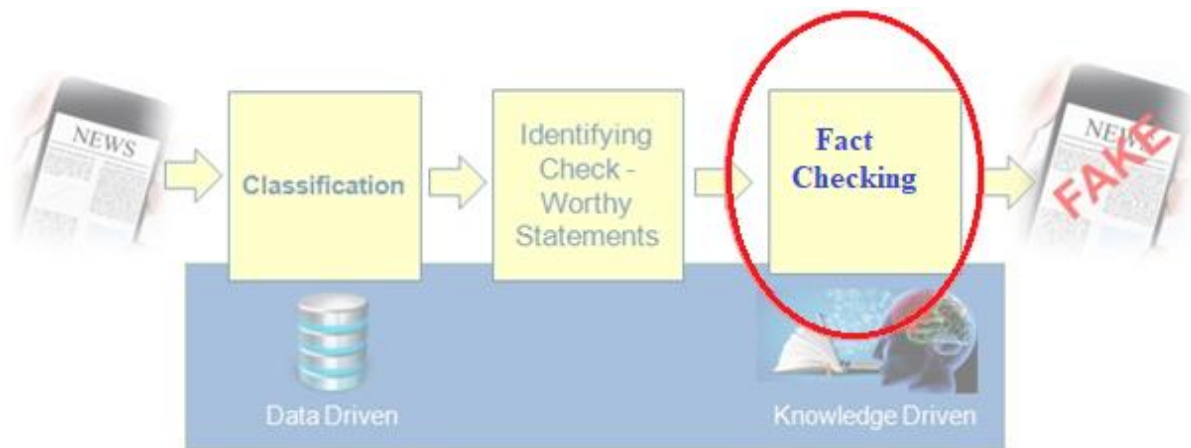


Figure 7.49: Proposed diagram for fact-checking (General View)

Most of the automated methods were based on supervised learning. In order to verify the veracity of news through fact-checking (Rashkin et al., 2017; Shu et al., 2017), the major limitation of the text classification approach is that fact-checking the claim requires world knowledge (Nakashole & Mitchell, 2014).

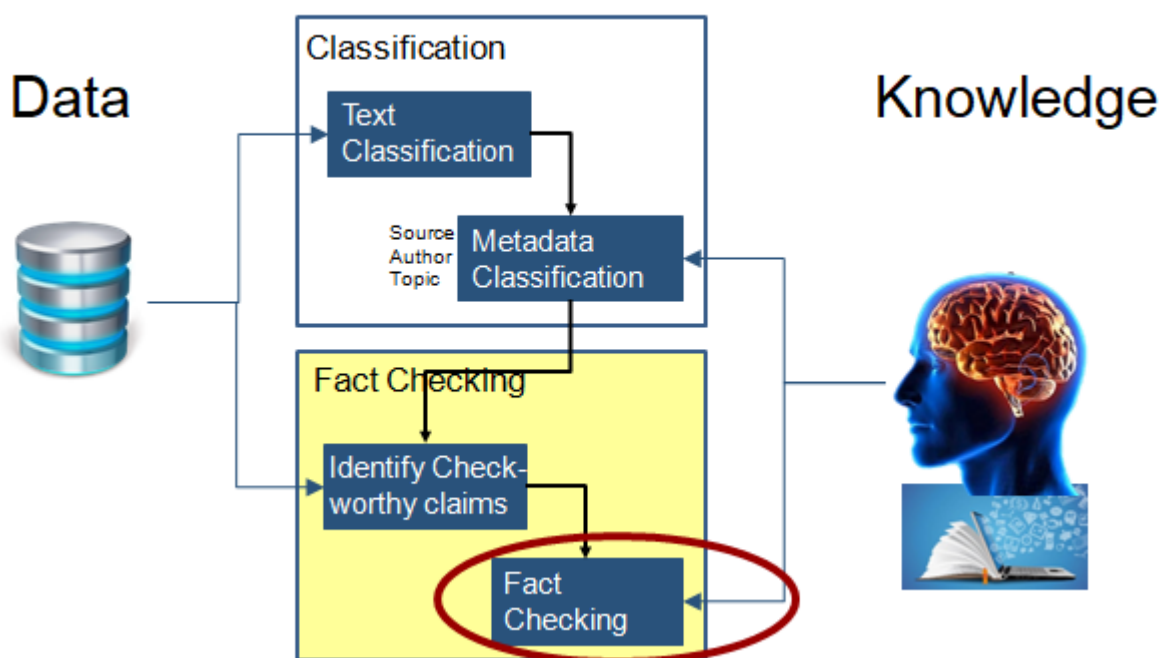


Figure 7.50: Proposed diagram for fact-checking (Inner View)

Popat et al (2020) proposed an approach to check the fact of the claim using a credibility check; where the credibility is checked from the social media sites/news and then given to a classifier for a credibility check. They conducted various experiments with fact-checking websites, e.g. snopes.com and politifact.com.

## 7.1 Problem Statement

Driven by the need to automate the fact-checking process in today's large amount of misinformation, in this chapter I give a contribution to automated fact-checking. There are different ways and methods for the detection of fake news, but I choose fact-checking as an interesting approach to tackle the problem. An important motivation for my research are efforts to automate fact-checking (Wu et al., 2014; Hassan et al., 2015). One side classification approaches are supervised, so we need a prior dataset to train our model but as mentioned earlier, obtaining a reliable fake news dataset is a very time-consuming process (see Chapter 5). On the other hand, despite great efforts by researchers we still do not have automated and context-aware fact-checking engines that are trustworthy enough to replace human fact-checkers. The challenge is to develop an automated application that takes claims directly from mainstream news media websites and fact-checks the news after applying classification and fact-checking components. Fact-checking is a challenging and time consuming process and with today's vast amounts of information, manual fact-checking is not feasible (Wu, Cheng, & Chai, 2018). When news is identified as fake, the existing techniques block it immediately due to its functionality as we cannot replace it; but when news is identified as fake we need at least an expert opinion or verification before blocking that particular news. The existing fake news systems are based on predictive models that simply classify whether the news is fake or not. The major challenge in these cases is to train the model, but this is impossible due to the unavailability of corpora. An alternative approach is needed that combines knowledge with data and fact-checking of check-worthy claims that look deeply at the content of the news with expert opinions, and at the same time can detect the fake news. My research started with identifying the problem that comes from the huge amount of time that human fact-checkers need to check a claim. To reduce some time, the first task of fact-checking, identification of check-worthy claims can be automated (see Chapter 6). In this chapter, I focus on political news which are annotated by fact-checking organizations and try to give a contribution to automate the fact-checking process. In section 7.4 I propose a

methodology for automation which comprises of the proposed approach and dataset explanation for better understanding.

## 7.2 Fact-checking

Fact-checking by humans can take as little as 15 to 30 minutes for a simple fact-check; a full day for a more typical one, to two or more days for complicated fact-checks (Hassan et al., 2015). A fact is something that has taken place and is also correct. In the context of news articles, events that have taken place and statements that claim to be true are factual, while opinions and interpretations are not. Before proceeding with the explanation of fact-checking and fact-checking algorithms I give a definition of fact-checking. According to Cambridge Dictionary<sup>63</sup> fact-checking is the process of verifying that all facts in a text, news article, speech, etc are correct. Manual fact checking is a time-consuming process, but automated fact checking can help to reduce the time and burden on humans. Fact checking is often considered a multi-step process (Riedel, 2014). Barron-Cedeno et al., 2018 describe the fact-checking pipeline which starts with monitoring different media sources; published online or even social media. From these sources, the first step is to identify articles that may contain interesting information to check. This is typically done by humans. The classification described in this chapter is supporting the human fact checker in identifying news items that might contain fake news. These articles are analyzed and then only the check-worthy statements are extracted (Barron et al., 2018). In Section 6.3, I presented a methodology for identification of check-worthy statements. The identified check-worthy statements are normalized where appropriate and then fact-checked. Finally, the results and verdicts from fact checking are published for the general public. End-to-end fact checking systems cannot be trusted without human intervention. I decided to make a contribution in the fact checking process that could help humans and reduce the burden of fact checking.

Most existing work on fake news detection is based on linguistic approaches (Jeffrey T. Hancock, Jennifer Thom-Santelli, 2004), but linguistic analysis alone has a major drawback. It is limited because it does not take into account useful contextual information around a claim. Combining linguistic approaches with additional analysis such as semantic analysis (Feng & Hirst, 2013) is useful and improves classification performance. Lexical and syntactic

---

<sup>63</sup> <https://dictionary.cambridge.org/dictionary/english/fact-check>

features detect writing styles commonly found in fake news content. Other work combines linguistic analysis with metadata attached to news stories. In a social network, metadata is used to analyze behaviors and patterns, that are often repeated in the spread of fake news (Cook et al., 2014). Social media-based methods combine features from user profiles (Castillo et al., 2011b), post content and news propagation (Wu & Liu, 2018b). Despite promising results, this approach is only applicable in the social media context, where the timeline of information dissemination can be easily followed.

There are two types of fact checking: *manual fact checking and automated fact checking*.

### 7.2.1 Manual Fact Checking

Manual fact checking is a disadvantage today, but automated fact checking can help to reduce the human burden. While end-to-end fact-checking solutions are not yet trusted to replace human fact-checkers, automating fact checking subtasks can support human fact checkers and save time. Fact checking is often considered as a multistep process, that includes the extraction of check-worthy statements (Vlachos and Riedel, 2014), the task on which my work is focuses. There are different ways to fact check news using the Internet: check the sources, check the URL, check the images and videos, pay attention to layout and text style, usage data protection and more<sup>64</sup>.

### 7.2.2 Automated Fact Checking

Fact checking is an intellectually demanding and time-consuming process, and with today's vast amount of information, manual fact checking cannot keep up (Wu et al., 2014). On the other hand, despite great efforts from researchers, there are still no automated and context-aware fact-checking engines that are trusted to replace human fact checkers (Sarr and Sall, 2017). The existing fake news systems based on the predictive models simply classify whether the news is fake or not fake (see also Chapter 5). Some models use source reliability and network structure, so the big challenge in these cases is to train the model, but this is impossible due to the unavailability of corpora (Ferreira et al. 2016). Zhang et al. (2020) have presented a comprehensive ecosystem that includes a detection system, an alert system, and an intervention system according to user behavior.

---

<sup>64</sup> <https://web.fhnw.ch/plattformen/blogs/wirtschaft/2018/10/31/10-tipps-zum-umgang-mit-fake-news/>

Automated fact checking encompasses many different methods, ranging from theoretical to practical approaches. Popat et al. (2016) proposed a model to support or refute claims from snopes.com and Wikipedia by considering supporting information from the web. They consider an open-domain setting without assuming any particular properties or structures in the input data. The solution automatically finds sources in news and social media and feeds them into a supervised classifier to evaluate the credibility of a claim. The approach presented by Wu et al. (2017) is based on structured data, which is increasingly common as more structured datasets become available either directly or through information extraction. They considered claims as queries with parameters, which allow the queries to be tested not only for correctness but also for more subtle quality measures by perturbing their parameters. Ciampaglia et al. (2015) also use publicly available databases, but they do not use structured datasets. Instead, they create knowledge graphs extracted from Wikipedia and use network analysis to predict whether an unobserved triple is likely to appear in a graph. They show that any collection of human factual knowledge can be used for automatic fact checking.

Thorne et al. (2018) created claim verification dataset consisting of 185,445 claims verified against Wikipedia pages. They label claims as SUPPORTED, REFUTED or NOTENOUGHINFO. The accuracy they achieve is 31.87% when the claim is accompanied by the evidence and 50.91% when the evidence is ignored. Regardless of their work, I did not use any external sources when classifying sentences as check-worthy or not but made the decision based on the dataset alone. Another approach used is to match a claim with an existing one, previously fact checked by fact-checking organizations. This reduces the task to sentence-level text similarity (Hassan et al., 2017; Riedel, 2014). Sentence level text classification is used to find check-worthy claims.

### 7.3 Role of Knowledge Engineering in Fact Checking

Knowledge engineering is appropriate for representing expert knowledge that is useful for fact checking. In Chapter 5, I explained that machine learning is appropriate for building AI-based systems but in some cases, knowledge-based systems can also be helpful. In the context of fake news, it can be said that knowledge is an important issue in distinguishing between fake and non-fake. The existing language-based and the feature-based content are not sufficient due to the distribution patterns of fake news (Zhou, Cao, Jin, Xie, Su, Zhang, et al., 2015b) but auxiliary features such as author credibility, source, and spreading pattern can

play a more important role in detecting fake news. If a news item is detected as fake, we need at least an expert opinion or verification before blocking that particular news. Xichen et al. (2020) suggested that the social context of the news content should also be examined, as news disseminators may target a wide audience that is not considered in data-driven approaches. Therefore, fact checking is required. Knowledge-based systems can be helpful in the future if we have a dataset of credible authors. With the help of the author's credibility check, we can detect the verdict of the news. First, detect the check-worthy statements which can help in fact checking and save time in fact checking. For this purpose, I have applied different experiments and found that Support Vector Machine (SVM) and Feed Forward Neural Network (FNN) give better results in checking the credibility of the statements.

In some cases, it is not possible to know whether a piece of information is a fact or not. In this case, we can compare it with known facts. Knowledge Linker (Ciampaglia et al., 2015c), PRA (Lao & Cohen, 2010), and PredPath (Shi & Weninger, 2016) are fact-checking approaches that compare a piece of news with known facts. There are also prediction algorithms that use knowledge for fact checking such as Degree Product (Shi & Weninger, 2016), (Adamic & Adar, 2003) and (Kyle Julian, 2016).

When comparing information extracted from news articles with known facts, one of the main problems is the credibility of the sources of the facts. With limited time and delicate skills, it is difficult for media and specialists to collect different facts from all of the sources. Shortly after the occurrence of an event, fake news starts to spread around the world; therefore, in this case, early detection is important to avoid worsening the situation. One of the possible solutions to prevent the spread of fake news is to the identification of check-worthy statements from potential fake articles, including causal relationships, and compares them with a dynamically updated knowledge graph for news facts. This technique has also been proposed by Pan et al. (2018). In a knowledge graph, different entities are defined as nodes and different relationships between them are defined as edges (Jia, Wang, Lin, Jin, & Cheng, 2016). An example of this is WordNet (Miller, 1995) and OpenKN (Liu, Wang, Jia, Li, & Yu, 2014) and realistic applications include document understanding (Wu et al., 2012) and link prediction (Liu et al., 2014). Google also uses a knowledge graph to improve the results of its search engine by collecting information from a variety of sources. All extracted information is presented to users in an information box next to the search results. Figure 7.51 shows an example of the Google knowledge graph.

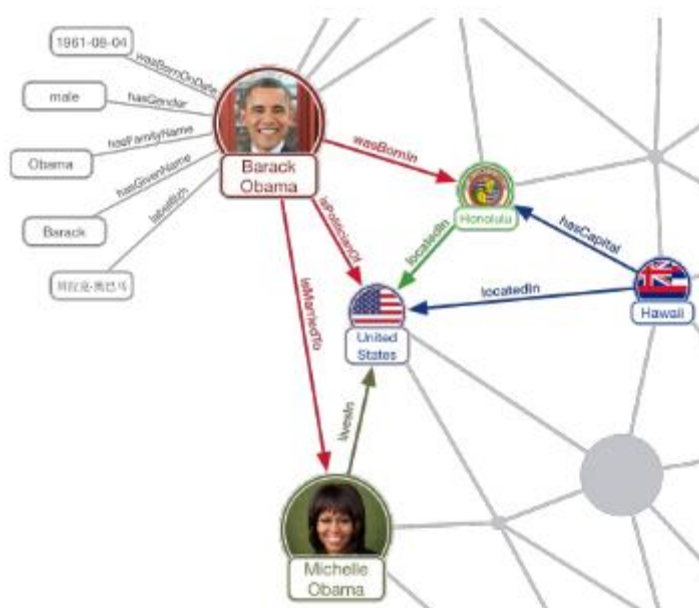


Figure 7.51: Example of knowledge graph (Zhou et al., 2019)

In the context of news propagation, a crowdsourcing model with knowledge graphs can be applied. Fact-checking sites, such as TruthSetter<sup>65</sup>, now also provide more scalable, peer-based assessments. People who hear about the events faster and more accurately can check the facts that they are sure of without much professional expertise. While doing the fact checking they can use a structured visualized interface for building and editing knowledge graphs by filling in the “subject”, “action”, “object”, “time” and “location” entities. The design of the model, in this case, could be visually similar to the Google knowledge graph as shown in Figure 7.51. Along with a working feature of being crowd-sourced, this model is user friendly to non- experts as well. Due to the dynamic updating of the knowledge graph, the timely fact information can be utilized in this model to detect fact tampering attacks in news articles. In the next steps, I highlight the problem in fact checking and then the methods used for evaluation.

### 7.3.1 Meta-Data

We can analyze fake news using various similarity measures, such as location, time, author, and quality. We can detect whether the same news has been published by other media agencies

<sup>65</sup> <https://truthsetter.com>



or not. We can check the location of the news: maybe the news has a higher probability of being fake if it was generated somewhere else and not in the place it deals with e.g. Trump writes about China or the Arab states or news about Hillary Clinton originates in Russia<sup>66</sup>. We can check the quality of the news; it is more likely that fake news does not cite sources and simply claims something, whereas real news cites the source (Zhou et al., 2018). We can check the timing of the news to see if the same news appears in other media or if it is repeated more often at the beginning because it is interesting and over time it is recognized as fake, which reduces the repetition.

### 7.3.2 News Content Models

Words in news media and political discourse have considerable power in shaping people's beliefs and opinions (Rashkin et al., 2017). A content model is a formal representation of structured content as a collection of content types and the relationships<sup>67</sup> among them. Content Modeling is the process of creating content models that describe structured content<sup>68</sup>. News content models are based on the characteristics of news content features. News content modeling involves identifying requirements, developing a taxonomy that satisfies those requirements, and considering where metadata should be allowed or required. Figure 7.52 shows news content models.

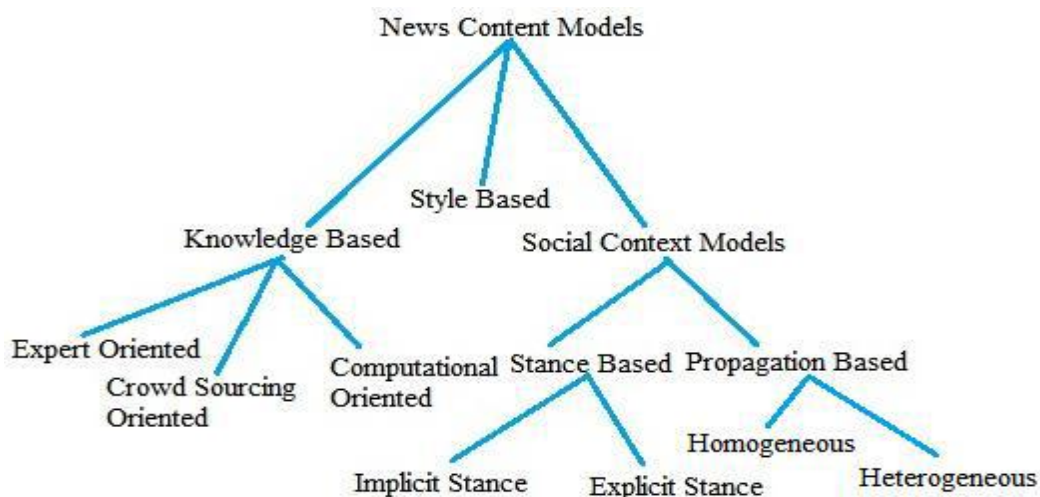


Figure 7.52: News content models

<sup>66</sup><https://theconversation.com/how-media-outlets-from-around-the-world-are-reacting-to-the-presidential-campaign-66263>

<http://www.clevegibbon.com/content-modeling/elements-of-a-content-model/>

<sup>68</sup><https://www.cmswire.com/content-strategy/content-modeling-what-it-is-and-how-to-get-started/#:~:text=The%20Definition%20of%20Content%20Modeling&text=%22Content%20modeling%20is%20the%20process,a%20design%20and%20technology%20agency.>

News content models can be categorized into knowledge based and style based, but due to the expansion in social media, another type is included which is a social context model. The main focus of news content modeling is on news content features and especially factual sources to detect fake and real texts (Shu et al., 2017). In the next sections, I will explain knowledge-based, style-based and social context models individually with examples. My focus is only on the knowledge-based approaches and existing applications in this area with examples as it relates to fact checking.

### 7.3.2.1 Knowledge-Based Content Models

The goal of a knowledge-based content model is to use external sources to fact check news content, and the goal of fact checking is to assign a truth value to a particular claim (Riedel, 2014). We can categorize knowledge-based fact-checking applications into three parts: expert oriented, crowdsourcing oriented and computational oriented.

- **Expert Oriented:** In expert oriented fact checking, we need domain experts who can examine data and documents to verify claims. Some notable fact-checking applications are Snopes<sup>69</sup> and PolitiFact<sup>70</sup>. Expert-oriented fact checking is not only very challenging but also time-consuming. Once a new claim is made, fact checkers consult domain experts, journals or statistics already available in that particular domain. This can take a lot of time, so the classification approach presented in Chapter 5 which can help identify potential fake news, together with the identification of check-worthy claims (see Chapter 6) enables efficient and timely fact checking. These mechanisms help and support the reader after critically evaluating the news before forming a judgment through fact checking. The aim of this work is not to provide results on whether the content is fake or not, but to provide a mechanism for critical evaluation during the news reading process. The reader starts reading the news and a fact-checking technique provides the reader with an opportunity to have any related or linked stories displayed for critical evaluation before rating. A formula for a rating measure is used and if the rating measure is below a threshold, the story is not displayed on the related fact check page (Guha, 2017).

There are three commonly accepted characteristics of fake news: the text of an article, the user response and the source, which must be included at one point. Ruchansky et al.

---

<sup>69</sup>[www.snopes.com](http://www.snopes.com)

<sup>70</sup>[www.politifact.com](http://www.politifact.com)

(2017) proposed a hybrid model that captures users' temporal behavior from published articles and measures text response. The second component score then estimates the score for each user and then combines it with the first module (Ruchansky et al., 2017).

- **Crowdsourcing Oriented:** A crowdsourcing approach allows a group of people to discuss and annotate the veracity of a particular claim. So, in other words, we can say that it completely relies on the wisdom of the crowd to provide fact checking based on their knowledge. Fiskkit<sup>71</sup> is an example of this type of fact checking as it allows people to discuss and annotate the accuracy of the news article at hand (Potthast et al., 2016). Another fake news detection application provides the ability to detect fake articles and allows users to report suspicious news content for editors to review further. Following the Facebook flag method of involving the public and using crowd signals to detect fake content, I applied the labeling technique (see Section 7.4.2). An algorithm called detective (Andrea et al., 2014) was developed to check the accuracy of labeling at runtime using the Bayesian inference method. This algorithm selects small subsets of each news, which are then sent back to the expert, who then determines if the news is fake. If the news is fake it is automatically stopped.
- **Computational Oriented:** Computational fact checking aims to provide users with an automatic system that can classify true and false content. Computational fact checking works on two points that identify check-worthy claims and then distinguish the truthfulness of factual claims. It works on the important basis and users viewpoints on the specific content (Houvardas & Stamatatos, 2006). Open web and structured knowledge graphs are examples of this type of computational-oriented fact checking. Open web sources are used that can differentiate news into true and false (Banko et al., 2007; Magdy & Wanas, 2010). Differentiating fake content can be divided into three categories: serious fabrication, large-scale hoaxes and humorous fake. Conroy et al. (2015) provide a way to filter, vet and verify news and discuss the pros and cons of these news in detail (Conroy, Rubin, & Chen, 2015b).

Bajaj (2017) developed a data-oriented application, that uses an existing dataset and then applies a deep learning method that proposes a new text classifier capable of predicting whether a news is fake or not. Traditionally, all rumor detection techniques are based on message level detection and analyze credibility based on data but in real-time detection

---

<sup>71</sup>[www.fiskkit.com](http://www.fiskkit.com)

based on keywords, the system then collects related microblogs using data collection. The proposed model combines user-based, propagation-based and content-based models and checks the credibility in real-time and returns the response within thirty-five seconds (Zhou, Cao, Jin, Xie, Su, Zhang, et al., 2015b). I have discussed different approaches that have been defined in recent years to address the problem of detecting fake news in social and news media. Most of these approaches are based on supervised or unsupervised methods (Chaovalit et al., 2005). These approaches do not give good results because there is no gold standard dataset available to train and evaluate the classifier to give good results. Subhabrata et al. (2015) explain the classification methods that are not specialized in detecting fake news. The motivation and psychological state of people may be different from those of professionals in the real world. Unlike my work, their focus was on political debates, which have different discourse characteristics than speeches which were also included in my dataset.

### 7.3.2.2 Style-Based Content Models

The style-based approach assumes that fake news editors use a particular writing style to appeal to a broader audience. This type of writing style is not evident in articles with real news content. The purpose of this activity is to mislead, distort or influence a large population. Social media provides researchers with additional resources to supplement and enhance news context models.

### 7.3.2.3 Social Context Content Models

Social context models are the engagement with the process of analysis and the capture of information in different forms from a different perspective. The existing approaches can be categorized as **stance based** and **propagation based**. An important point to highlight here is the existing approaches to social context models are used for detecting fake news.

- **Stance-based approach:** This method determines whether the reader of a particular news source is in favor of, against or neutral about that particular news. User stances can be categorized into explicit stances or implicit stances. In explicit stances, readers make direct expressions, such as thumbs up or thumbs down. For implicit stances, the results are extracted from social media posts, automatically determining from user posts whether the majority of users are in favor or against (Mohammad et al., 2017; Qazvinian, Rosengren, Radev, & Mei, 2011).

- **Propagation-based approaches:** These approaches examine the context of relevant events on in social media posts to identify the fake news and credibility of the particular news. Zhou, Cao, Jin, Xie, Su, Zhang, et al. (2015a) proposed a method to build a three layer network to include only the sub-events; after that they can check the credibility of news using a graph optimization framework. Nidhi & Gupta. (2011) proposed a propagation-based algorithm for users coding, credibility checking and tweets. Propagation based approaches are divided into **homogeneous and heterogeneous** parts. Homogeneous propagation contains single entities such as a post or an event (Zhiwei Jin et al., 2016; Zhiwei Jin et al., 2014; Manish et al., 2012). A heterogeneous credibility network contains multiple entities such as posts, events and sub-events.

### 7.3.3 Drawbacks with Existing Fact-Checking Applications

Existing fact-checking applications use digital tools to identify, verify and respond to misleading claims. The following are some challenges for existing applications.

- Once the claim is received it is forwarded to domain experts for annotation. Therefore the existing fact-checking websites are time-consuming.
- The growth of fact checking has been hampered by the nature of the work. It is time-consuming to find claims to fact-check. Journalists have to spend hours going through transcripts of speeches, debates and interviews to identify claims to research (Hassan et al., 2015)
- For fact checking the claims are passed to human editors, so there is a possibility of bias due to human nature e.g. like/dislike (Shu et al., 2017)
- Credibility related issues:

Only 42.67% of websites covered the knowledge base for credibility assessment, so most website domains have low credibility (Liu, Nielek, Adamska, Wierzbicki, & Aberer, 2015). Manual (human) credibility indicators for a set of websites are costly and search engines provide few information cues e.g. title and URL (Erkan & Radev, 2004).

Automation can help in the dissemination of fact checks. The technique I propose, which is a combination of text classification and fact checking of check-worthy statements, may perform better compared to existing applications.

## 7.4 Methodology

Features such as the size of the datasets and the length of the texts will also be discussed as part of the analysis. To classify the texts as real or fake news, each text was pre-processed and ‘cleaned’. Feature extraction techniques are then used before classification is performed. The framework then integrates various components of the fact-checking process; extracting check-worthy statements from mainstream news media sites, text searching for related stories from knowledge sources such as Wikipedia, fact-checking claims after linguistic analysis and aggregation. This process is outlined in Figure 7.53 and explained in the next sections.

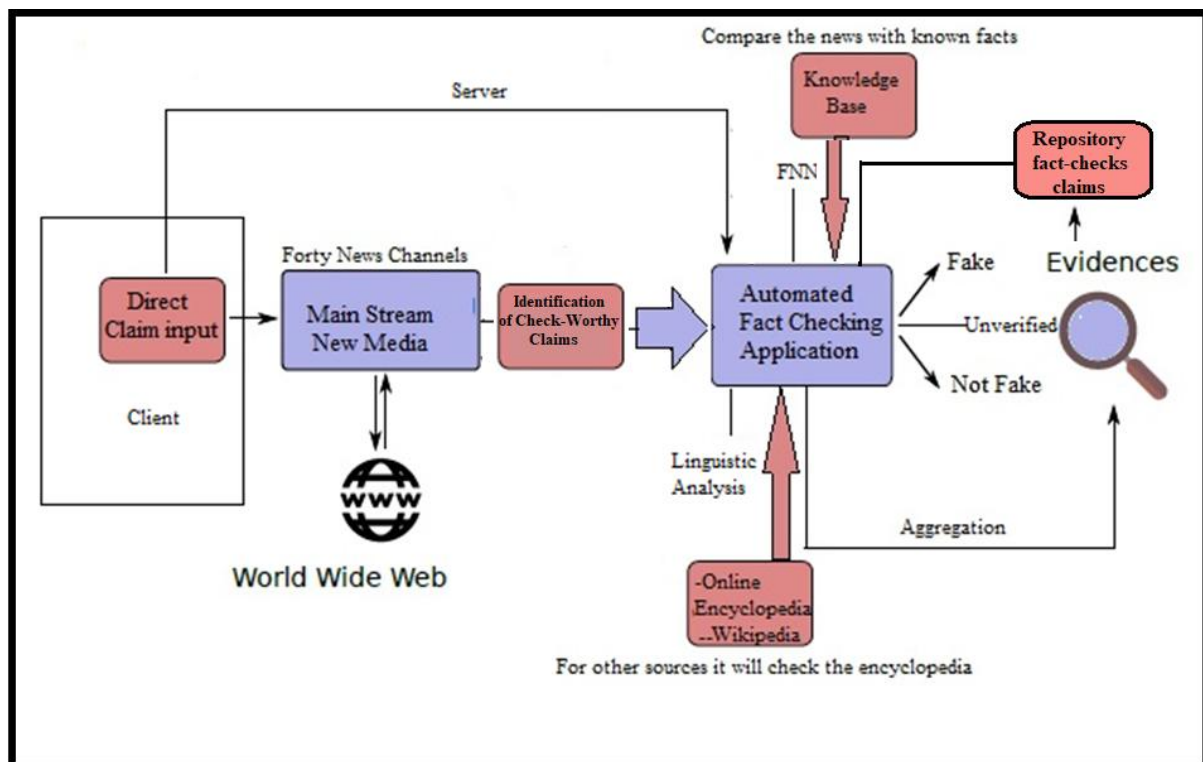


Figure 7.53: System Framework for automated fact checking

### 7.4.1 Proposed Approach for Automation

Manual classification of millions of news published online is a time-consuming and expensive task (Nidhi & Gupta, 2011). Learning from data and engineering knowledge could be helpful to solve the problem of fake news in news media. Some claims contain facts but they are irrelevant as the general public is not interested in knowing these claims. Some other

claims contain facts that the general public wants to know about. These facts could be helpful for fact checking. Hassan et al. (2015) proposed that political claims can be classified into three categories depending on the information they contain (Table 7.14). After separating irrelevant statements, we can then focus only on potentially fake statements, which we can then tag with relevant features and pass to the model for review and verification. This could be useful for identifying fake news.

Table7.14: Categorization of claims on the basis of facts Hassan et al. (2015)

Non Factual Sentences (Example)	Unimportant Factual Sentences (Example)	Check Worthy Factual Sentences (Example)
But I think it's time to talk about the future.	Next <u>Tuesday</u> is <u>Election Day</u>	He <u>voted against</u> the first <u>Gulf War</u>
You remember the last time you said that?	<u>Two days</u> ago we ate lunch at a <u>restaurant</u>	Over a million and a quarter <u>Americans</u> are <u>HIV-positive</u> .

To achieve this goal, a new combination algorithm approach was developed that classifies the text as soon as the news is published online. The main hypothesis behind this work is that each news article differs in context, making it difficult to detect fake news; especially when one part of the news is fake and another part is not. By labelling these known facts (true, false and unverified) available with each claim (news headline) in my proposed approach. I explain the dataset in Section 7.4.2.

#### 7.4.1.1 Automation Challenges

The goal is to develop an automated application that combines text classification (as described in Chapter 5) and identification of check-worthy statements (see Chapter 6) with knowledge-based fact checking to detect fake news. Chapters 5 (Fake news detection through classification) and 6 (Fact checking: identification of check-worthy statements) but for a web-based application, we have other fundamental challenges. Hassan et al. (2015) highlighted

two challenges for fact-checking applications. My task is not only to model the automated fact-checking application, but also to address the challenge of understanding what others are saying about the same claim that will be fact checking. Secondly, my proposed application should be able to distinguish between credible and non-credible sources. Third, the fact-checked news should receive evidence with the degree of representation.

#### 7.4.1.2 Linguistic Analysis

Linguistic features such as grammar features, word patterns, term count, and the occurrence of certain expressions are the main focus. Possible methods for automatic clickbait detection were discussed by (Conroy et al., 2015b). A review of methods for detecting both textual clickbait cues and non-textual cues including image and user behavior. A system was presented by Bourgonje et al. (2017) for detecting the stance of headlines in relation to their corresponding article bodies. The approach could be applied in fake news, especially clickbait detection scenarios. The spread and persuasion of fake news has been explained by the theory of Elaboration Likelihood Model. Rashkin et al. (2017) compared the language of real news with that of satire, hoaxes, and propaganda to find the linguistic characteristics of the untrustworthy text. Stylistic cues were used in their experiments to determine the truthfulness of the text. I have been concerned with language testing because I hope that it can also be helpful for fact checking in some context.

#### 7.4.2 Dataset Exploration and Analysis

For this task, I collected news articles from different websites. The organizations considered are Politifact<sup>72</sup>, Emergent<sup>73</sup>, daily mail<sup>74</sup>. The dataset separated different attributes like web page, claim, description, label, tags, domain, and date. Then I analyzed the dataset and checked how the articles differ from each other, both in terms of content and attribute. I sorted the data with different result indicators such as how often they were shared. All check-worthy claims were labelled as fake, true, and unverified (unverified claims are those that are not ambiguous). The corpus contained 2146 check-worthy claims, from which 731 were true claims, 793 were unverified claims, and 551 were false claims. The identification of check-

---

<sup>72</sup><https://www.politifact.com/>

<sup>73</sup><https://www.emergent.info>

<sup>74</sup><https://www.dailymail.co.uk/home/index.html>



worthy claims has already been explained in Chapter 6. In the next step, I identified the features that could help distinguish the claims as fake or not fake compared to the known facts. For each claim, I tagged known entities, such as name, location, country, organization name and any other information that could help us during fact check. Figure 7.54 shows the distribution of the sentences. I used RapidMiner, a powerful machine learning tool for data exploration. The discussion of data exploration and the machine learning tool I used is covered in Section 5.2.1.

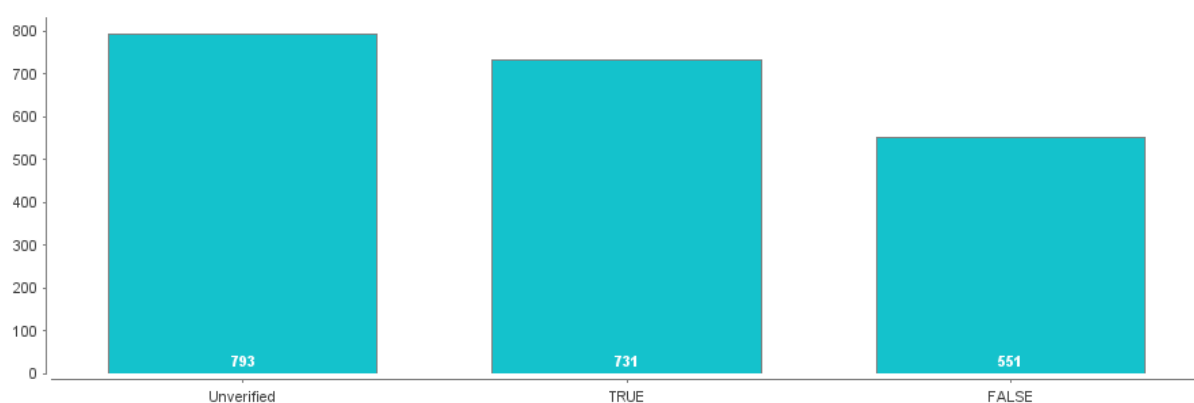


Figure 7.54: Class distribution of sentences

Table 7.15 shows an example of how a row in the data set is constructed. Each line in a file consists of the claim, the source, tags and the claim label.

Table 7.15: Dataset Row Structure Example Set

Sr. No	Claim	Source	Tags	Claim Label
1	An oil pipeline exploded in Saudi Arabia	Dailymail.co.uk	Pipeline, Saudi + Arabia	Fake
2	Microsoft is going to acquire Mojang AB	Avsforum.com	Microsoft, Mojang	Non-Fake
3	A fourth-grade student from Texas was suspended after threatening another student with magic	Dailymail.co.uk	Magic, Texas, Hobit, Lord + of + the + rings	Unverified

The class labeling chart represents the data set labeling procedure for each class of data. In Figure 7.55 below, can see the three classes False, True, and Unverified labeled claims for next step.

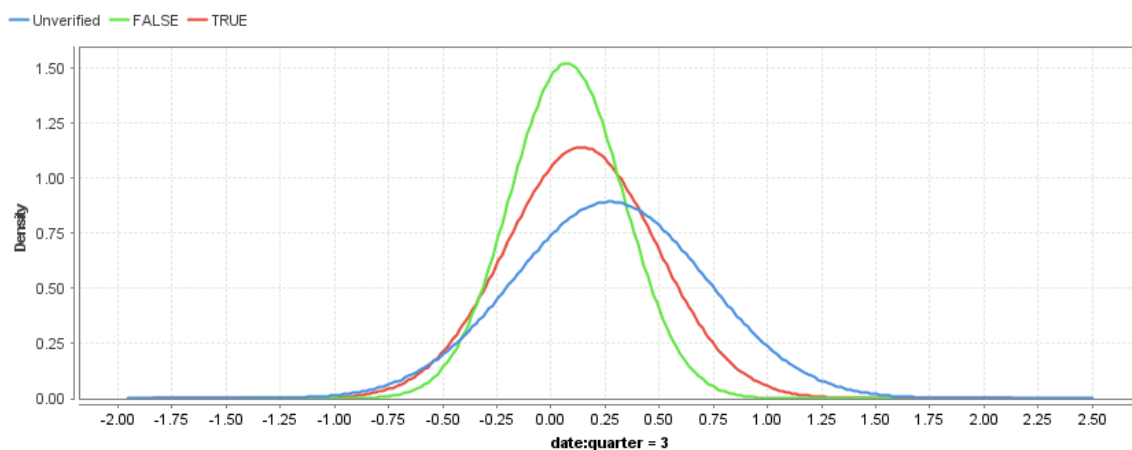


Figure 7.55: Dataset class labelling chart

As can be seen in Figure 7.56, almost half of the claims are true; this is because of the correct sources and evidence. I have tagged these claims as fake, non-fake and unverified. These check-worthy claims have already been examined by one of the fact-checking organizations as I discussed above in Section 7.4.1.

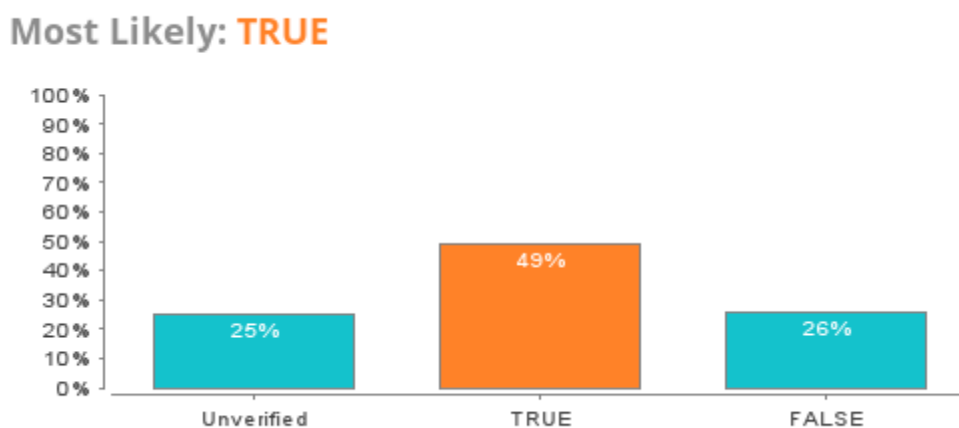


Figure 7.56: Claim label

I labeled 2146 check-worthy claims as true, false, unverified and then tagged the known facts present in these claims, e.g., location, place, event, time, name for experimentation (see Table 7.15). I considered 80-20 split of the data for the training and test sets.

As shown in Figure 7.57, I examined the labels and the percentage of the three categories of labels. I find that 38% are unverified, 35% are true.

Value	Count	Percentage
Unverified	793	38.22%
TRUE	731	35.23%
FALSE	551	26.55%

Figure 7.57: Claim labelling percentage

Figure 7.58 shows the positive factors in the dataset that can help us to model the design.

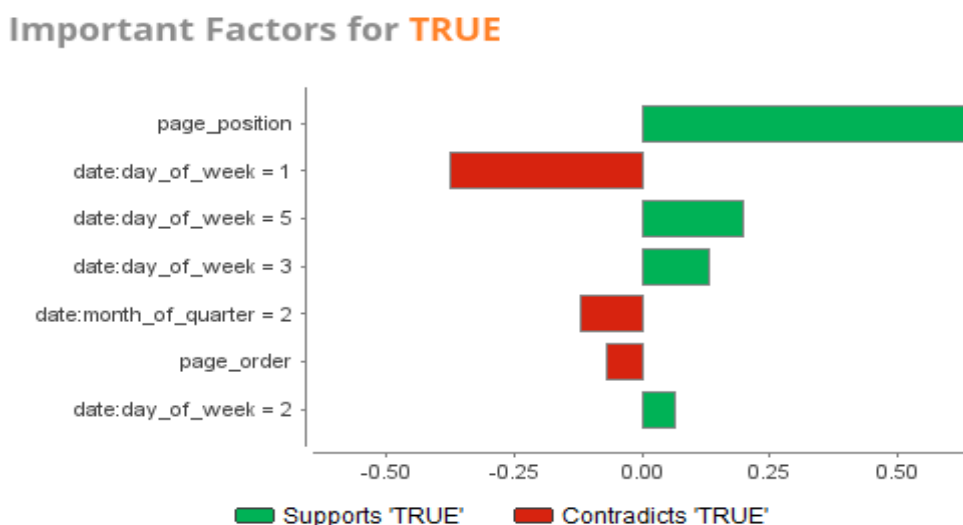


Figure 7.58: important factors that involved in dataset features

## 7.5 Discussion

To determine which features are effective for fact checking, different tagging features were analyzed. I found that while some claims contained facts (names, time etc.) they were unimportant and in some cases not helpful in identifying fake, so I did not tag them. Some

claims contained other facts that I highlighted above that could be helpful for both fact checking and general audience interest. Hassan et al. (2015) suggested that these claims could be classified into three categories depending on the information they contained. The corpus was labeled with location, author information, date, organization, headline, news text and tags. Then, I used RapidMiner for exploring the dataset and classifying the classes respectively fake, non-fake and unverified. Based on the annotations, I highlighted the percentage of labeling. My proposed approach contained three parts: classification, identification of check-worthy statements and fact checking. The data side contained the text classification (Chapter 5) and the identification of check-worthy statements (Chapter 6) while the knowledge side contained fact checking (Chapters 7 and 8), all of which help refine our results. I presented the general framework of my developed tool in this chapter. The development and evaluation results are presented in the next chapter.

## 8 Development and Evaluation

This chapter describes the implementation of the findings from Chapter 7 into a prototype (artefact), which consists of four components. It corresponds to the demonstration stage of the DSRM of Peffers, (2007) discussed in Section 3.3. I discuss the competing approaches with which I compare my results and conclude by describing the experimental settings for implementing my results. Looking at the implementation, firstly a web application is presented that takes the claim as input and verifies the facts from the news after collecting relevant sources from the mainstream news media. Automatic fact checking is based on several factors including extraction for given claims, reliability evaluation of media sources, stance detection of documents with respect to claims and fact checking of claims (Xu et al., 2018; Baly et al., 2018; Mohtarami et al., 2018; Baly et al., 2018; Mihaylova et al., 2018). These factors correspond to Natural Language Processing (NLP) and Information Retrieval (IR) tasks which also include information extraction and question answering (Shiralkar et al., 2017). Text classification problem has been addressed using the Veracity inference approach and this problem is tackled by developing linguistic, stylistic, and semantic features (Rashkin et al., 2017; Mihaylova et al., 2018; Nakov et al., 2017). Additionally, information from external sources has also been used (Mihaylova et al., 2018; Karadzhov et al., 2017). These steps are typically conducted in isolation.

In the work of author Wang and O'Brien (Wang, 2017; O'Brien et al., 2018), an algorithm has been proposed to predict the factuality of claims with a focus specifically on the input claims and their metadata information (e.g., the speaker of the claim). Thorne et al., 2018 proposed that the Fact Extraction and Verification (FEVER) focus has been driven towards a specific domain (e.g., Wikipedia). To address these gaps, the developed tool can be used to cover all fact-checking steps and can be used to search across different sources, predict a claim's sentence-level factuality, and can finally be used to present a set of evidence.

### 8.1 Web Application Development Task

I present a fact-checking system that combines text classification and fact checking of check-worthy statements to detect fake news. My developed model includes various components

such as document retrieving documents from mainstream media sources with different types of reliability, classification, evidence extraction, linguistic analysis and aggregation. Several organizations are performing manual fact checking over suspicious claims due to the rapid increase of fake news across social media and their negative impact on people (Mihaylov et al., 2015; Mihaylov and Nakov, 2016; Vosoughi et al., 2018). Manual fact checking is a challenging and time-consuming task, researchers are driving their focus toward automatic fact-checking methods. Automatic fact checking is a multi-step process and includes, checking the reliability of the media sources from which documents are retrieved, retrieving potentially relevant documents for a given claim, predicting the factuality of given claims, (Mihaylova et al., 2018; Karadzhov et al., 2017; Mohtarami et al., 2018; Xu et al., 2018; Mihaylova et al., 2018). The general architecture of the proposed fact checking is shown in Figure 7.53. For dual verification of the results, linguistic analysis checks are performed before it comes to the assessment and aggregation of the given claim. The text is cleaned in the same way in the training and testing phases. Stop words are removed along with punctuation before lemmatization is performed. The output is converted into TF-IDF values which are fed into a pre-trained Feed Forward Neural Network (FNN) model trained on the given dataset. A prediction is made along with aggregation that the prediction is correct based on the model. The Python micro framework Flask<sup>75</sup> was used to build the application using HTML<sup>76</sup> and CSS<sup>77</sup>. Flask was chosen due to its lightweight nature which was suitable for this application. The user interface consists of a text area for the text input and a button to run the search. I used the Python soup library<sup>78</sup> which makes it easy to scrape information from web pages. It also provides support for iterating, searching and modifying the data from the dataset between HTML and XML parsers. The developed application code is available at the end of this document in the appendix . This system is accessible through a web browser and has two sides: Client and Server. The first step in this process is that the user on the client side sends a request to the server in form of textual claim. The below figure shows the code settings for the claim input panel where the user will enter text for claim search.

---

<sup>75</sup><https://pypi.org/project/Flask/>

<sup>76</sup><https://html.com/>

<sup>77</sup><https://getbootstrap.com/>

<sup>78</sup><https://pypi.org/project/beautifulsoup4/>

```

180
181 <br><br>
182 <div class="row">
183 <div class="col-md-3"></div>
184 <div class="col-md-6">
185 <div class="alert alert-success"><strong>Searched Term:</strong> <?php echo $searchterm; ?></div>
186
187 <br><br>
188 <div class="box">
189 <div id="chartContainer" style="height: 370px; width: 100%;"></div>
190 <button class="btn invisible" id="backButton">< Back</button>
191
192 </div>
193 </div>
194
195 </div>
196 </body>
197

```

Figure 8.59: Search panel

This request is processed by the server, which forwards the request data to the document retrieval component, which then retrieves a list of relevant documents (see Section 8.3.) from three different sources: Wikipedia, mainstream news media (forty news organizations) and open search (see Section 8.4). The retrieved result is further refined by bypassing the retrieved document (see Section 8.3-8.4). The perspective of each relevant document with respect to the claim is detected by the fact-checking component, which is typically modeled by using tags and comparing these tags with a claim in the news. Further explanation about the model predictions is rationalized at the sentence level using the same component. A linguistic comparison also takes place in the fact-checking component to analyze the language of each document after it has been passed through the linguistic component (see Section 7.4.1.2). Finally, the aggregation component makes the final decision on the factuality of the claim by aggregating the classification and fact-checking predictions about the claim (see Section 8.7.1.1).

```

42 <script>
43 window.onload = function () {
44
45 var totalVisitors = <?php echo $TotalStatements; ?>;
46 var visitorsData = {
47   "New vs Returning Visitors": [{
48     cursor: "pointer",
49     explodeOnClick: false,
50     innerRadius: "75%",
51     legendMarkerType: "square",
52     name: "",
53     radius: "100%",
54     showInLegend: true,
55     startAngle: 90,
56     type: "doughnut",
57     dataPoints: [
58       { y: <?php echo $UnverifiedStatements; ?>, name: "Unverified", color: "#E7823A" },
59       { y: <?php echo $FalseStatements; ?>, name: "Non fake", color: "#546BC1" },
60       { y: <?php echo $TrueStatements ; ?>, name: "Fake", color: "#b22222" }
61     ]
62   }],
63 },
64 "Unverified": [{
65   color: "#E7823A",
66   name: "Unverified",
67   type: "column",
68   dataPoints: [
69     { x: new Date("1 Jan 2015"), y: 65 }
70   ]
71 }],
72 "Fake": [{
73   color: "#b22222",
74   name: "False",
75   type: "column",
76   dataPoints: [
77     { x: new Date("1 Jan 2015"), y: 33000 },
78     { x: new Date("1 Feb 2015"), y: 35960 },
79     { x: new Date("1 Mar 2015"), y: 42160 },
80     { x: new Date("1 Apr 2015"), y: 42240 },
81     { x: new Date("1 May 2015"), y: 43200 },
82     { x: new Date("1 Jun 2015"), y: 40600 },
83     { x: new Date("1 Jul 2015"), y: 42560 },

```

Figure 8.60: False, True and Unverified Statements Percentage

It can predict the factuality of a given claim with appropriate sentence-level evidence to support its prediction. The above figure shows the prediction criteria. The full code and configuration is available in the appendix (see Appendix A-C) at the end of this document.

## 8.2 Front End Display for our Fact-Checking System

The front end comprises of three views:

- **Claim Entry View:** Figure 8.61 shows how to enter a claim to be checked for factuality.



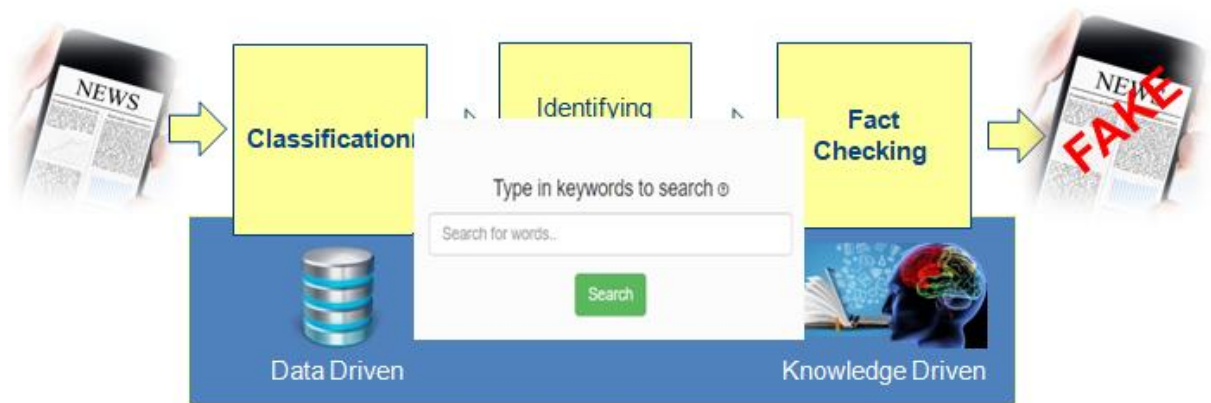


Figure 8.61: Claim input panel for users

- **Output view:** This includes lists of documents from factual types of sources: Wikipedia, Open browser search and mainstream news media (forty organizations) (Section 8.3). The final score for the input claim is shown in the next sections (Figure 8.68), and the fact check score appears next to it for each document.
- **Retrieved document view:** When retrieving a document, the proposed system displays not only the text of the document but also the important sentences, based on their score regarding the claim in highlighted form (Figure 8.65).

### 8.2.1 Aggregation

The linguistic analysis and fact checking by the Feed Forward Neural Network (FFN) are performed in parallel on the given claims and the retrieved documents based on the claim from all sources. After fact checking, an average score is assigned to each claim and then an aggregate score is compiled in the list of retrieved documents with the highest rank. A higher agreement score means the claim is true and a higher disagreement score means false.

### 8.2.2 Key Points for the Fact-Checking System

The fact-checking application which I have developed consists of the following three approaches.

- The evidence extraction phase takes place based on the fact checked given claims through the user's text input window.

- To check the reliability of the given claims and retrieved media sources (Baly et al., 2018).
- The fact-checking module takes place which checks the claim through Feed Forward Neural Network (FNN) algorithms and also verifies the results through linguistic checking.

The above three steps correspond to Natural Language Processing (NLP) and information retrieval (IR) tasks, that involve information extraction. Existing approaches were mostly used for text classification problems and utilized different linguistic, stylistic, and semantic features (Karadzhov, Nakov, Màrquez, Cedeño, & Koychev, 2017b) and few of them used information from external sources (Mihaylova et al., 2018). For example, looking at recent work on Fact Extraction and Verification (FEVER) (Thorne, Vlachos, Christodoulopoulos, & Mittal, 2018), the focus is on a specific domain (e.g., Wikipedia) and according to (Alsmadi & O'Brien, 2020; Shu et al., 2017) algorithms have been proposed to predict the factuality of claims by focusing mainly on the input claims and their metadata information. I have tried to fill these gaps and designed the proposed fact-checking system, which consists of fact-checking steps (Figure 7.53) and is not able to search across different sources but also predict the factuality of claims and present a set of evidence with explanations to support the prediction. There are the results based on fake, non-fake and unverified claims with the aggregation of the factuality. An example is shown in Figure 8.68, where the claim factuality of 90% is labeled as “Fake”. I present the proposed fact-checking system as an online application for automatic fact checking of claims. My developed system is helpful for individuals and professionals to check the facts of claims in one place as it not only has the ability to check the factuality of a claim with aggregation after multiple checks but also presents relevant documents as evidence to support its prediction for a particular claim. In the future, I plan to continue to expand the system and make it even more advanced and user-friendly by focusing on the further development of the underlying components such as stance detection, topic detection, credibility comparison, and source-wise, author-based cross-linguistic settings (see Sections 7.4.1).

## 8.3 Text Retrieval

This step is feasible because we only need to retrieve data using different APIs from different news agencies. The tool I developed offers both the ability to enter keywords and to select claims from mainstream media that have been fact checked by existing fact-checking organizations. The first step is to convert an entered claim directly into a query by considering its verbs, nouns, and adjectives (Potthast et al., 2013). I used the Natural Language toolkit (NLTK)<sup>79</sup> which is suitable for linguistically related tasks, to extract relevant documents from mainstream new media sites and also from open search.

Below figure shows the verification of our proposed results with Wikipedia for dual verification. It also checks the relevant documents from Wikipedia. For full code details which include other media sites check Appendix B.

```

51 """ wikipedia """
52 wikiLinks = []
53 wikiSubjects = []
54 searchterm = search
55 wikiSearch = searchterm.replace(" ", "+")
56 urlwiki = "https://en.wikipedia.org/w/index.php?cirrusUserTesting=glent_m0&sort=relevance&search="+wikiSearch+"&title=Special%3ASearch&profile"
57 r = requests.get(urlwiki)
58 content = r.content
59
60 soup = BeautifulSoup(content, 'html.parser')  Define a constant instead of duplicating this literal 'html.parser' 4 times. [+3 locations]
61 for ul in soup.findAll("ul", {'class': 'mw-search-results'}):
62     for li in ul.findAll("li", {'class': 'mw-search-result'}):
63         for a in li.findAll("a"):
64             wikiLinks.append("https://en.wikipedia.org/"+a["href"])
65             sob = a["href"].replace("/wiki/", "").replace("_", " ")
66             wikiSubjects.append(sob)
67             break
68
69 date = ul.find("div", {'class': 'mw-search-result-data'})
70 dateEx = date.text
71 myArray = dateEx.split(" - ")

```

Figure 8.62: Results verification phase with Wikipedia

Finally, the forty links with the highest match to the given claim were determined. My proposed approach stands out well from existing approaches where human fact checkers mainly focus on multiple sources rather than relying on one source (like Wikipedia). The user view or claim input window is shown in Figure 8.63.

<sup>79</sup><https://www.nltk.org/>

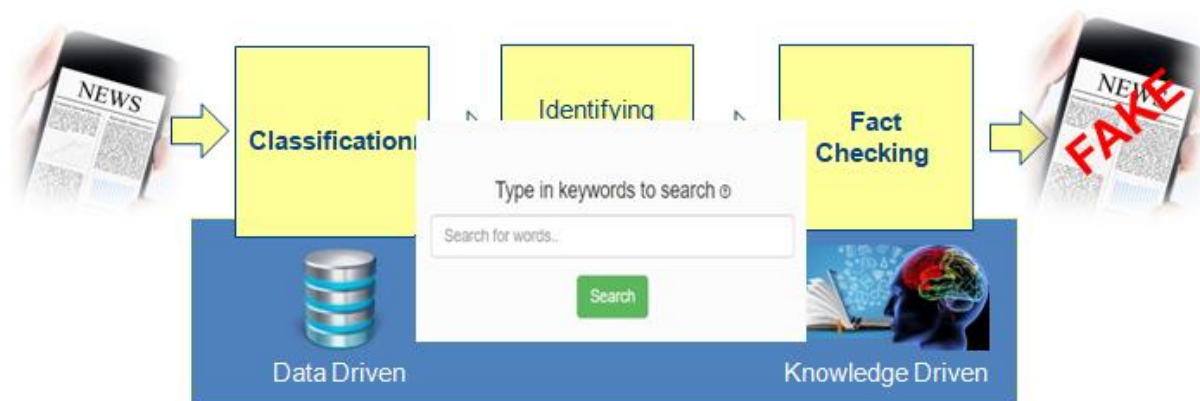


Figure 8.63: The Web-Application main Interface

Some researchers address this problem with text-based processing and separate fake and non-fake text based on classification. On the other hand, some previous researchers have separately studied components of this multi-step process, which include:

- (i) Retrieving potentially relevant documents for a given claim (Karadzhev et al., 2017a; Mihaylova et al., 2018)
- (ii) Verifying the reliability of media sources from which documents are retrieved (Popat et al., 2017)
- (iii) Predicting the stance of each document according to the given claim (Baly et al., 2018; Du, Xu, He, & Gui, 2017), and then predicting the factuality of claims (Mihaylova et al., 2018)

In my work, I present an automated web-based fact-checking tool that combines all its four components into one framework and has the potential to predict the factuality of a given claim along with evidence for its sentence-level predictions. In Chapters 5 and 6 developed classification system predictions were based on document level, but here we further analyzed and verified the factuality of the claim through sentence level.

## 8.4 Source Collection

Currently, the relevant text to a given query is collected from any media sources using search engines (e.g. Google, Bing and Yahoo). Four types of sources are used to retrieve relevant documents: Wikipedia, high factual content media, mixed and low factual content media. Usually, journalists spend a considerable amount of time verifying their information sources

(Nguyen, Kharosekar, Lease, & Wallace, 2018; Popat et al., 2016). Sometimes, a list of unreliable online news sources was also provided by the journalists of some fact-checking organizations. The below figure shows the comparison of total statements collected from different sources and then from those false and true statements highlighted. For complete source code please look at Appendix A.

```

cgi-bin > viewdata_php
1  <?php
2
3  $TotalStatements = htmlspecialchars($_POST["TotalStatements"]);
4  $FalseStatements = htmlspecialchars($_POST["FalseStatements"]);
5  $TrueStatements = htmlspecialchars($_POST["TrueStatements"]);
6  $UnverifiedStatements = htmlspecialchars($_POST["UnverifiedStatements"]);
7  $searchterm = htmlspecialchars($_POST["searchterm"]);
8  ?>
9
10 <head>
11 <style>
12 .box{
13     position:relative;
14     text-align:center;
15     height: auto;
16     border-radius:15px;
17     padding:10px 20px 65px;
18     background-color:#fcfcfc;
19     -webkit-transition: all 1000ms ease;
20     -moz-transition: all 100ms ease;
21     -ms-transition: all 1000ms ease;
22     -o-transition: all 1000ms ease;
23     transition: all 1000ms ease;
24     box-shadow:0px 0px 30px rgba(0,0,0,0.15);
25 }
26 #backButton {
27     border-radius: 4px;
28     padding: 8px;
29     border: none;
30     font-size: 16px;
31     background-color: #2eacd1;
32     color: white;
33     position: absolute;
34     top: 10px;
35     right: 10px;
36     cursor: pointer;
37 }
38 .invisible {
39     display: none;
40 }
41 </style>

```

Figure 8.64: Comparison of True and False Statements

I extracted the information from news sources with high accuracy using available libraries that provide parsers for information extraction (Stanford NLP)<sup>80</sup>. In my work, I have used the above three categories of media sources to retrieve documents using the document retrieval

<sup>80</sup> <https://nlp.stanford.edu/software/lex-parser.shtml>

component. In addition to the forty mainstream media sources and open web search documents, I used Wikipedia, which contains accurate information. Figure 8.65 shows the top search documents collected based on targeted mainstream media sources for the given claim.

Pentagon outlines withdrawal of troops from nation's capital after ...	Haberman
Putin Rejects Trump's Request for Ten Thousand Russian Troops to ...	Donald Trump, Life of a Zombie PartyAnd an army is blindly following.By Charles M. Blow
U.S. Army: Esper reverses plan to send active-duty troops home ...	Why The Times Calls Trump 'Mr.' (No, We're Not Being Rude)Here's why The Times refers to President Trump as Mr. Trump and calls some retired military officials "Mr."By Philip B. CorbettPRINT EDITIONDecember 11, 2017, Page A2
White House wanted 10,000 active duty troops to quell protesters ...	The Soldier Donald Trump Called a TraitorThe president-elect's campaign crusade against Sgt. Bowe Bergdahl makes it unlikely that the former Taliban hostage can get a fair trial.By The Editorial BoardPRINT EDITIONThe Soldier Mr. Trump Called a Traitor November 27, 2016, Page SR10
Trump calls DC mayor 'incompetent' after senator says she evicted ...wjla.com › news › local › president-donald-trump-utah-senator-mike-lee-n...	Trump's 2nd Nominee for Army Secretary WithdrawsMark E. Green, a Tennessee state senator and former Army surgeon, said his nomination had become a distraction because of "false and misleading attacks."By Helene CooperPRINT EDITIONTrump's 2nd Nominee for Army Secretary Withdraws, Citing 'Misleading' Attacks May 6, 2017, Page A11
Trump Says National Guard Troops Will Begin Withdrawing From ...	A Bad Call on the Bergdahl Court-MartialThe political frenzy prompted by the case may have influenced the Army's decision to try a soldier who endured five years as a Taliban prisoner.By The Editorial BoardPRINT EDITIONA Bad Call on the Bergdahl Court-Martial  December 16, 2015, Page A34
Trump 'will not even consider' renaming Army bases named for ...	'Disgusted': Sen. Duckworth on unanswered questions about D.C....
Trump says he 'will not even consider' stripping Army posts of ...	Rep. Max Rose: Trump admin plan for Natl. Guard 'killing morale'
Trump dismisses possibility of renaming Army posts named for ...	Calling All Corpsmen-In this Time of Pandemic
Trump says admin won't consider renaming Army bases named after ...	Despite Trump's rhetoric, more troops diagnosed with brain injuries
Than 280 Former Military Officials, Diplomats Call on Donald Trump ...	Team Trump eyes dubious Iowa road trip for caucus members
Trump says his administration 'will not even consider' renaming ...	Why Republicans are taking aim at a war hero to defend Trump
Gavin Newsom sidesteps Trump's call for governors to 'dominate ...	Second chances at the intersection of sports and politics in Louisiana
	Tuesday's Mini-Report, 1.7.20
	Trump highlights accused war criminals at Republican fundraiser

Figure 8.65: Source collection from mainstream media and top search results

## 8.5 Fact Checking Module

For fact checking, I used Feed Forward Neural Network (FNN) model for classification as proposed in the paper (Xu et al., 2018). My developed model is a combination of Bag of Words (BOW) and constructed in a two-level hierarchy scheme. First, the tags (Name, Location, Event, etc.) are checked, and the system matches these tags with the claim and then the body of the claim for further verification. If the selected tags match, the model will segregate them and create an array into which all matching documents are inserted, with priority based on the best match. The related documents are then passed to the fact-checking module for comparison based on known facts. For this purpose, I have already sorted the dataset which was explained in the previous chapter in the data exploration section. Each claim was labeled with its own category and the facts were separated for comparison with the claim and the main body of the claim to get the status. For double checking, the documents

were indexed and retrieved using Apache Lucene<sup>81</sup>. It helped us to link the system with Wikipedia for comparing the statements and extracting the cleaned results. This step aims to overcome the limited size of labeled data at the time of training by using different domains.

The fact checking and linguistic analysis components are run simultaneously against all documents originally retrieved by the document retrieval component from any type of source. This component further rationalizes in depth to sentence level for further prediction of the developed model. The average of all the scores over these documents is computed and the aggregate scores for each matching best matching and less matching category are displayed at the top of the ranked list of retrieved documents. Finally, the factuality of the claim is determined based on the algorithm scores: the higher the score, the greater the claim is factually true and the lower the disagree score; the more false the claim is false.

## 8.6 Repository of Fact-Checked Claims

In fact checking claim matching is an important task in the fact-checking process (Majithia et al., 2019). This step aims to find the identical or similar claims from the repository of existing fact checks.

Once the claim is identified as fake, non fake or unverified, the fact-checking process stores it in the repository for future reference. To this end, each claim has a markup that stores it and retrieves it when the existing claim is requested for fact-checking. The fact-checking repository is composed of the fact-checked claims collected from different fact checking organizations as discussed in previous sections. The system compares the similarity between given claim and existing facts based on sentence level similarity of the basis of markup. The goal of this task is to check if they match something we have fact checked before.

## 8.7 Results

The web application was created for demonstration purposes. The web application takes a text as an input claim from the user and classifies the text as Fake, Non-fake or Unverified based on a pre-trained model in the fact-checking module. Below I show the results of all

---

<sup>81</sup><https://lucene.apache.org>



three categories with the full description of the claim and the results obtained by my developed application.

### 8.7.1 Example 1: Fake

To evaluate my developed application, I applied it to the dataset discussed in Section 7.4.2. The corpus contains 2146 news articles, of which 551 are fake claims, 793 are unverified claims and 731 are not fake claims. For each claim, I tag known entities such as name, location, country, organization name and other items that may help us help us in fact checking. The organizations considered are Politifact<sup>82</sup>, Emergent<sup>83</sup>, and Daily Mail<sup>84</sup>. These organizations have already fact checked all these claims. In the following table I show the claim and all the details of that particular claim which makes us check the fact of that claim.

Table: 8.16 A fake claim sample with assessment and explanation

<b>Claim:</b> KFC restaurants in Colorado will start selling marijuana
<b>Headline:</b> KFC restaurants in Colorado will start selling marijuana.
<b>Date:</b> 03/04/2017
<b>Description:</b> KFC Gets Occupational Business License To Sell Marijuana In Colorado Restaurants KFC Gets Occupational Business License To Sell Marijuana In Colorado Restaurants.
<b>Tags:</b> KFC, Marijuana, Hoaxes, Fake+ News, Colorado
<b>Evidence:</b> The Racket Report is an unreliable source, and this was a fake news article. Snopes provided a debunking
<b>Source:</b> Emergent
<b>Label:</b> Fake



Figure 8.66: Claim input panel for users

<sup>82</sup><https://www.politifact.com/>

<sup>83</sup><https://www.emergent.info>

<sup>84</sup><https://www.dailymail.co.uk/home/index.html>



### 8.7.1.1 Overall Result

As Figure 8.67 shows the factuality of the claim overall result is fake because different media channels have reported on this claim, so the initial response of the system is fake as per the sources available.

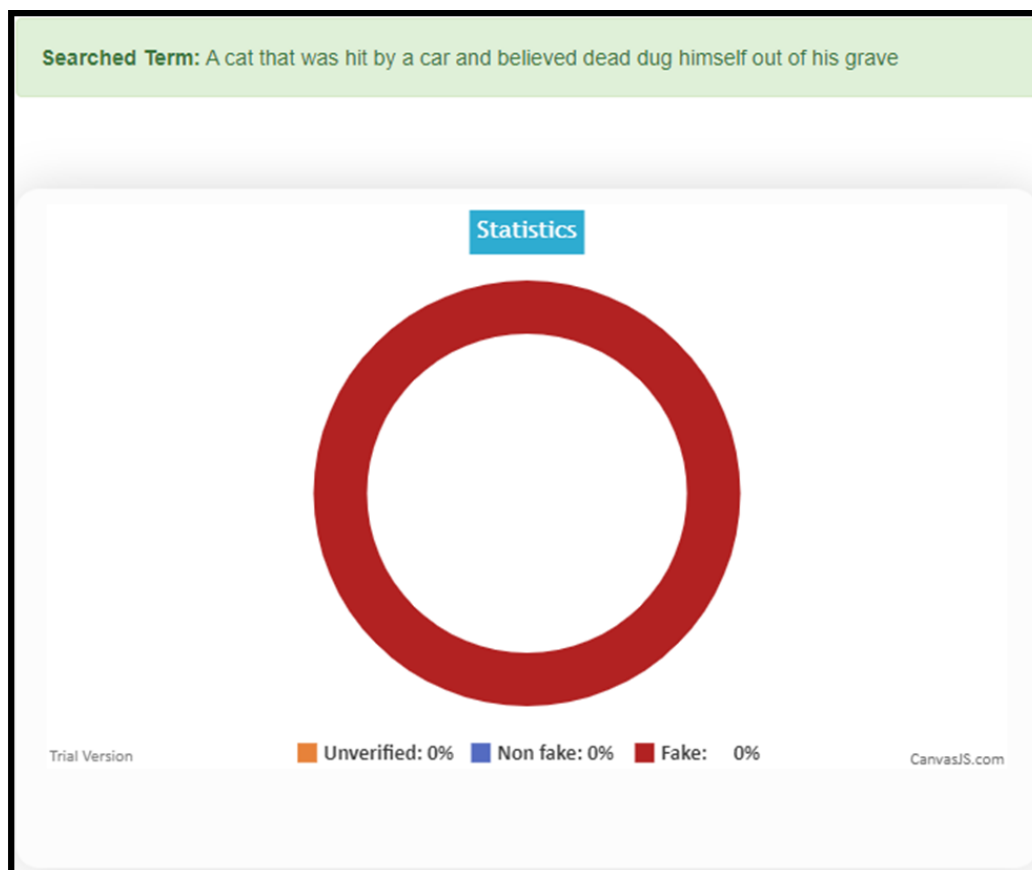


Figure 8.67: An Example of a fake prediction (General)

As I explained earlier, my developed system tests the factuality of the claim on sentence level after comparing of the claim with different checks highlighted in Figure 7.68.

### 8.7.1.2 After Sentence Level Comparison

Finally, we verify the claim by the fact-checking module, which can be seen in Table 8.16 in the tags section. In the table, we can also see that the racket report is an unreliable source, and it was a fake news claim. My system suggested a 90% fake factuality on a comparison with mainstream news media organizations and Wikipedia.

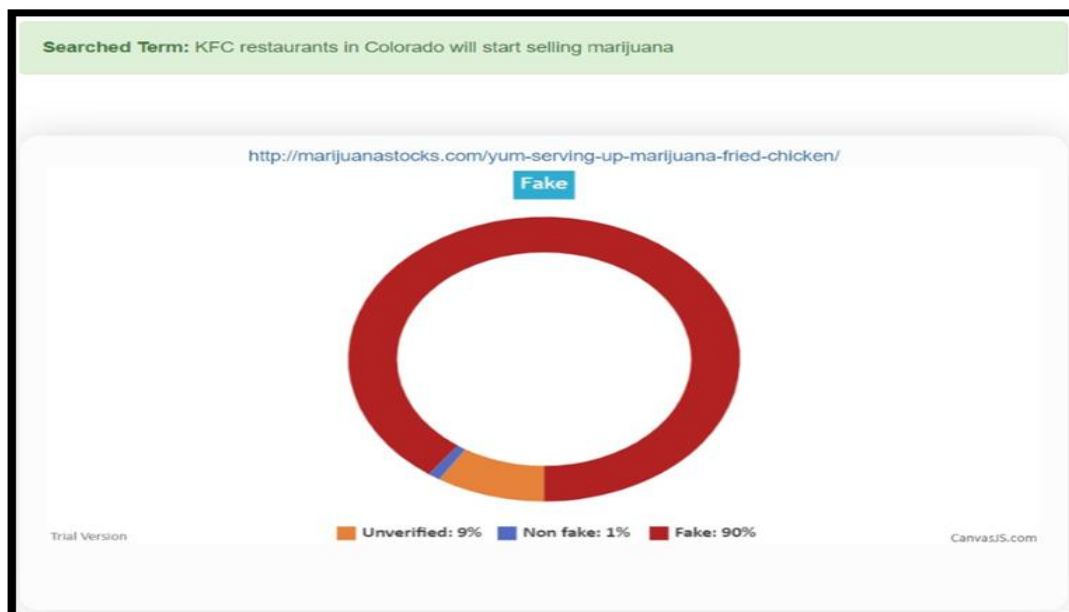


Figure 8.68: An Example of a fake Prediction with claim and evidence

### 8.7.2 Example 2: Non-Fake

Another claim published by emergent with the headline “cat claws out of grave five days later” was exactly true. The details of the news item are listed in the following table.

Table: 8.17 A Non-fake claim with assessment and explanation

<b>Claim:</b> A cat that was hit by a car and believed dead dug himself out of his grave
<b>Headline:</b> Cat claws out of grave 5 days later.
<b>Date:</b> 1/26/2017
<b>Description:</b> Bart the cat showed up in his neighbour’s yard five days after being buried. He should make a full recovery, according to the Humane Society.
<b>Tags:</b> Cat, Animals, Florida, Zombies
<b>Evidence:</b> The Humane Society in Tampa provided images and background on the cat and believes the cat's injuries are consistent with the story. Bart's owner, Ellis Hutson, said that one neighbour helped him bury the cat, and another neighbour found Bart. "I open the door and my neighbour’s standing there with the cat in her hand," Hutson told ABC. "She said, 'Bart is not dead.' I said, 'That impossible. We buried Bart.'" The involvement of the humane society combined with the other people in this story leads us to consider it true.
<b>Source:</b> Emergent
<b>Label:</b> Non-Fake

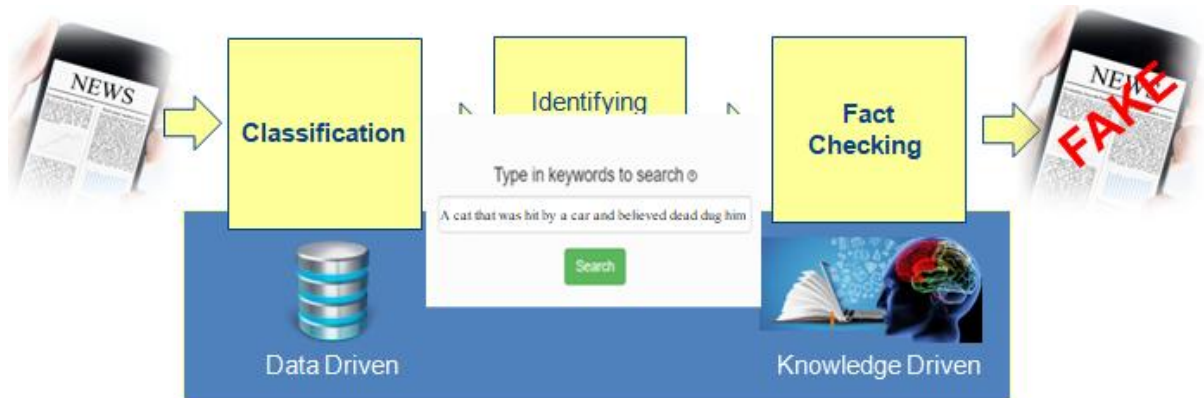


Figure 8.69: Claim input panel for users

### 8.7.2.1 Overall Result

If we verify the claim factuality of the claim, the initial findings based on the other media sources are bogus, but we need further sentence level investigation to verify the fact of the news. We compare it with known facts and the initial results are shown in Figure 8.70.

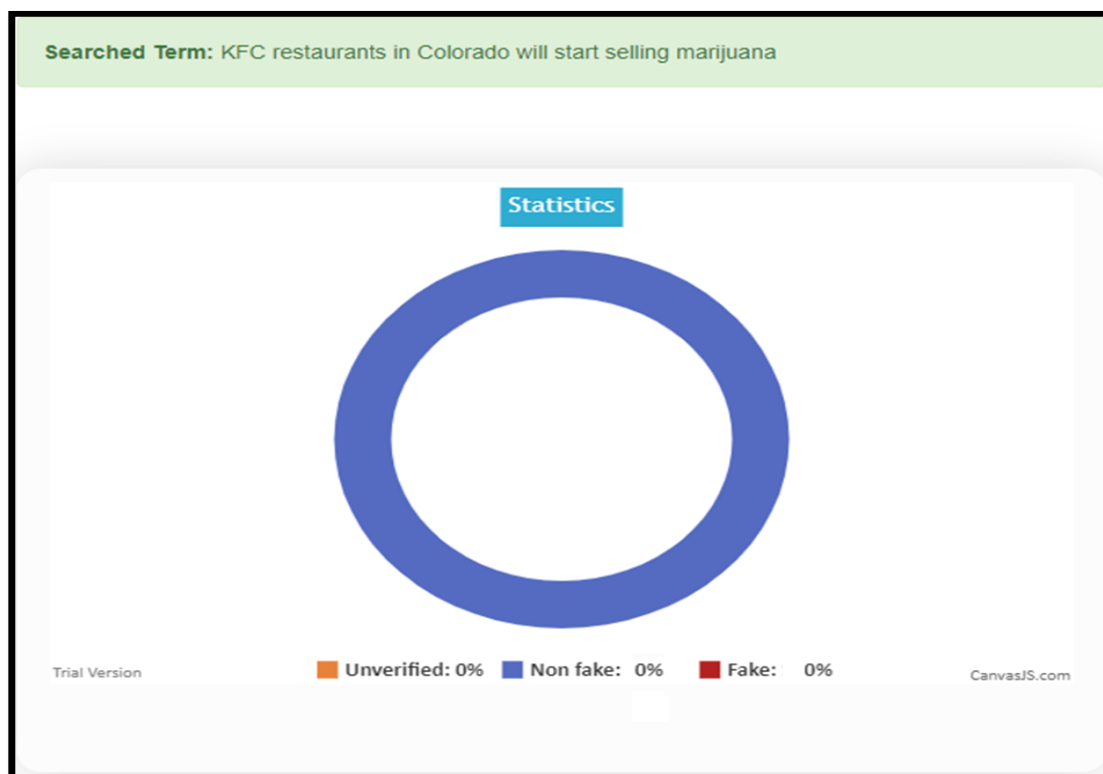


Figure 8.70: An Example of a non-fake prediction (General)

### 8.7.2.2 After Sentence Level Comparison

After comparison with different facts obtaining from the Tampa human society, the system has concluded that the claim is 80% non fake, 7% unverified, and 13% fake. So based on the majority, the system predicts that the overall result of the claim is non-fake. The overall results can be seen in Figure 8.71.

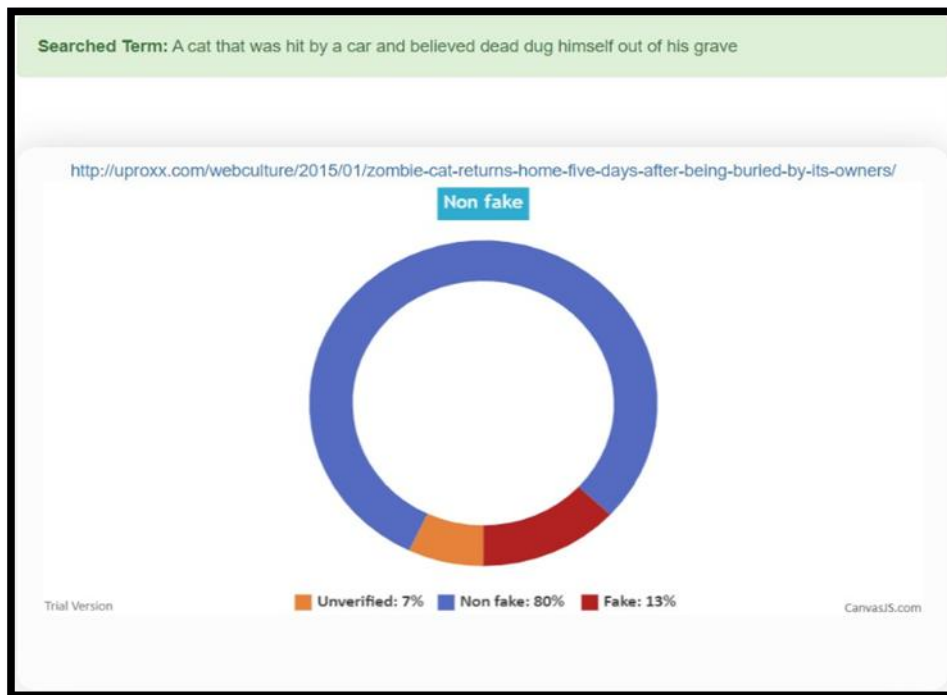


Figure 8.71: An Example of a Non-fake Prediction with claim and evidence

### 8.7.3 Example 3: Unverified Claim

Our final claim is the headline of a fourth grade student who was suspended from school after threatening his classmate. The status of this claim is unclear due to the lack of comment from school officials. The full story of the news can be seen in the table below.

Table: 8.18 Unverified claim with assessment and explanation

<b>Claim:</b> A fourth-grade student from Texas was suspended after threatening another student with magic
<b>Headline:</b> Parent: Fourth-grader suspended after using magic from 'The Hobbit'. Interview.
<b>Date:</b> 2/2/2017
<b>Description:</b> Allegedly, the 9-year-old told a classmate his magic ring would make them disappear. The boy had recently seen "The Hobbit" with his family and was supposedly inspired by that and the powerful ring in "The Lord of the Rings"
<b>Tags:</b> Magic, Texas, Hobit, Lord + of + the + rings
<b>Evidence:</b> The Odessa American was the first with the story Jan. 30, interviewing the boy's father, Jason Steward. They reported the child was suspended "for allegedly making a terroristic threat," though Kermit Elementary School Principal Roxanne Greer declined to comment. Until the school confirms the incident, we will keep this as Unverified.
<b>Source:</b> Daily mail
<b>Label:</b> Unverified



Figure 8.72: Claim input panel for users

### 8.7.3.1 Overall Result

Initial findings were based on the other media sources, which only reported the student father's point of view, not the other side's point of view. The system shows this claim in 100% unverified.

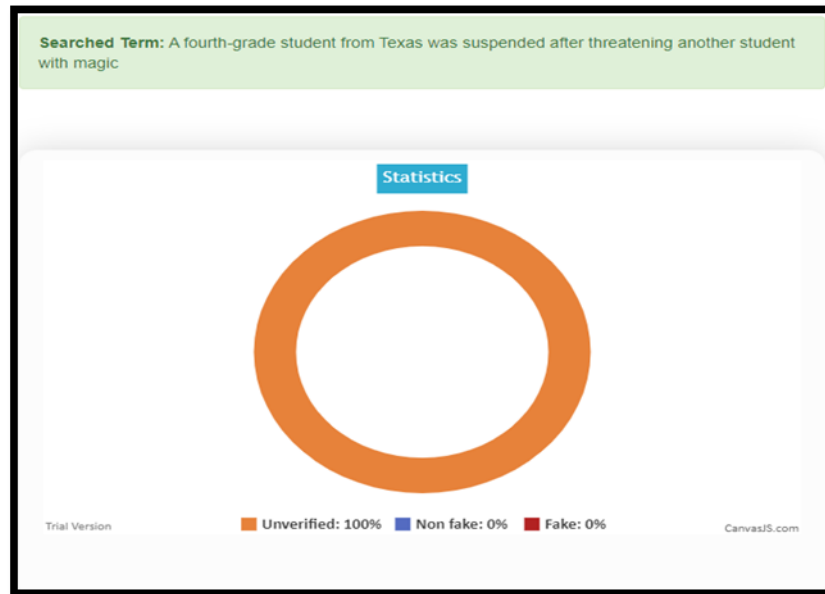


Figure 8.73: An Example of an unverified prediction (General)

### 8.7.3.2 After Sentence Level Comparison

After reviewing the factuality at the sentence level, which includes comparing text with different known facts such as the location of the incident, the father stance on media etc. We found that 65% of the claim status is unverified, but on the other hand such as the location of the school and the student's father's stance so the system predicts 13% true and 20% false status.



Figure 8.74: An Example of an Unverified result Prediction with claim and evidence

## 8.8 Conclusion from the Evaluation

Important insights can be drawn from the results derived from a dataset discussed in previous chapter (see Sections 7.4.2). While previous works separately investigated individual components of the fact-checking process, in this work, we present a unified framework which combines classification (Chapter 5), Identification of check-worthy statements (Chapter 6) and automation (Chapters 7 and 8). In this chapter I presented the results that integrate these components to not only predict the factuality of given claims but also provide evidence at the document and sentence level to explain its predictions. The primary focus of this research is driven towards fake news detection with the approach towards classification and fact checking. Here classification addresses the fake news, which are further analysed for fact checking. Fact checking consists of two parts, one being the check-worthy statements which will reduce the time and burden of fact-checking process and the other comparison with known facts. Besides knowing that text classification is data-driven, additional knowledge is required about fact checking.

The dataset used in Chapter 5 was taken from public domain. Experimental analysis on two publicly available datasets demonstrated interesting and improved performance. The initial results after applying this method gave an accuracy of 93%, 85% and 84% with the algorithms PA, NB and SVM respectively. The developed system with accuracy up to 93% proved the importance of classification in detecting fake news. In Chapter 6 the dataset was created using the information from the 2016 U.S. Presidential Election and the following year's election. The approach to the creation of the dataset was similar to that of the work of Patwari et al. (2017).

At the sentence level review, transcripts must be broken down into sentences. The sentences in the transcripts of the debates were considerably small and contained ill-defined sentences (see example in Table 6.6), which were manually deleted. Even though the number of sentences was decreased from 9187 to 8804, these sentences were not all check worthy, so there was no reduction in the number of check-worthy sentences. Based on the results drawn from Chapter 6, we proposed an approach of classifying statements into check-worthy and non-check-worthy, thereby taking into account the context around a statement. The initial approach to this step was initiated by extraction of sentences and context features from the sentences, followed by classifying the sentences based on these features. Based on the well-differentiating capability of the check-worthy statements, the feature set and the context

features were selected after several experiments. This work demonstrates that a significant contribution towards classification was made due to the inclusion of context in the approach. The results of the same were further analyzed by examining all the features used and which specific features contributed more towards classification.

In chapters 7 the dataset was created by collecting news articles from different websites. The organizations used for data collection were Politifact, Emergent, daily mail. The dataset separated different attributes such as web page, claim, description, label, tags, domain, and date (see 7.4.2). Further, I analyzed the dataset and determined how the articles differ from each other, both in terms of content and attribute. The data was sorted with different result indicators such as how often they were shared. All check-worthy claims were labeled as fake, true, and unverified (unverified claims are those that are not ambiguous). The corpus contained 2146 check-worthy claims, out of which 731 were true claims, 793 were unverified claims, and 551 were false claims. The identification of check-worthy claims has already been explained in Chapter 6. For each claim, known entities were tagged, such as name, location, country, organization name and any other relevant information that could contribute towards fact checking. The class distribution of the sentences is shown in figure 7.54. Further, a powerful machine learning tool for data exploration called Rapid Miner was used. The discussion of data exploration and the machine learning tool has been discussed under Section 5.2.1. With the goal in mind, I have developed applications that directly integrate various components of fact checking starting from the collection of check-worthy statements from mainstream news media sources, through information retrieval from credible sources. The proposed system compares the statements and predicts the fact of the news and shows the aggregation of fake and not fake news. Example 1 shows the initial prediction of the system was fake but when our developed system further investigated and compared with known facts we have come to know that this claim was 90% fake, 9% unverified and 1% non fake. In the non-fake example we can see that the initial response was non-fake but when we compare sentence level of fact-checking it was 80% non fake, 13% fake and 7% unverified. Similarly, in our last example the system's initial response was 100% unverified but after sentence-level prediction 65% unverified, 15% true and 20% fake. Finally, the conclusions were drawn that our developed system performed well when we combined classification and fact checking in identification of fake news.



## 8.9 Discussion

Fake news detection is a real problem for different sectors of society, which I have discussed in detail in various sections of this draft. The developed system can assist individuals and fact checking organizations in verifying the factuality of claims by presenting relevant documents. It provides evidence with a prediction explanation after integrating various components of the fact-checking process. The objectives of the project have all been satisfied. A framework has been developed to allow the evaluation of different classification and feature extraction techniques, as well as the creation of a simple web application that can classify a user submitted text as False, True or Unverified after combining machine (text based) and human based fact checking . The results are limited due to the small size of the dataset, i.e., there were not enough texts to both effectively train and test the model. Ultimately, the classification techniques analyzed in this project are not substantial enough to effectively combat fake news; however, the results have provided valuable insight into the potential of fact checking by incorporating knowledge engineering, which uses knowledge of previously verified facts.

The work described in this chapter, namely the development of an automated fact-checking tool, has met the requirements. I proved that we can detect fake news with the integration of text classification and fact checking of check-worthy claims. The chapter provided an answer to the research question “How can it be checked whether a statement is fact or fake?”

In the future, the tool will be further enhanced based on the future research directions discuss in Chapter 9.

## 9 Conclusion

This thesis studies the research problem of combining classification and fact checking of check-worthy statements, which allows detecting fake news in news media. In particular, it addresses the problems of text-based classification for fake news (Chapter 5), identifying check-worthy statements as input for fact checking (Chapter 6), automated fact checking (Chapter 7), and development and evaluation (Chapter 8).

Although the conclusions of individual chapters have already been presented, this chapter summarizes my main contributions and explains the main findings that contribute to answering my research questions.

### 9.1 Contributions

My key contributions are the following:

- I proposed an overall approach for fake news detection as a combination of classification and fact checking, where classification identifies potential fake news which are then further analyzed by fact checking.
- I have proposed a fact-checking approach combined with identification of check-worthy statements which is the subtask in fact checking and reduces the effort of fact checking.
- I automated an approach that considers the context around a statement to identify a check-worthy claim that can mimic a human fact checker's intuition in decision making.
- I proposed a framework that facilitates the evaluation and comparison of the accuracy of the best classifiers.
- I developed an application that can classify text and compare the claims with other media sources and known facts, and then present the veracity of the claims as real, fake or unverified news.

- I obtained good results in two different tasks: fake news detection through classification, and distinguishing between fake and non-fake news articles through fact checking. The automated fact-checking application will be freely available to the general public.

The main scientific contribution is the identification of a combined approach and the development of a computational model to detect fake news in news media, which is published in (Ahmed et al., 2019).

In **Chapter 5**, I first reviewed the existing state-of-the-art methods for detecting fake news and then discussed the strengths and limitations of the proposed solutions in detail. Next, in the technical background, different natural language processing (NLP) techniques are presented in detail. In addition to the NLP techniques, the other features sentiment, topic, context and part of the speech are described, as well as the evaluation metrics used to determine the performance of the different models. I developed a fake news model using machine learning and natural language processing. The proposed classifier uses text-based processing and achieved the highest accuracy of 92% after comparison with other methods (Ahmed et al., 2020).

In **Chapter 6**, I presented a context-aware approach to identify check-worthy statements using Feed Forward Neural Network (FNN) and Support Vector Machine (SVM), which yielded good results. The dataset contains debates and speeches from the field of politics. The task works in a highly imbalanced dataset, where the check-worthy sentences accounted for only 7% of the total dataset – which is typical for fake news. After collecting the dataset, several experiments were conducted to decide which features to extract and how to combine them to represent the sentences. For each sentence additional features were extracted from two previous and two subsequent sentences, to form a context window around the target sentence. As expected, the implementation of context features proved to be very useful in identifying check-worthy statements.

Experiments with different classification models showed that the Feed Forward Neural Network (FNN) with two hidden layers was better suited for this approach compared to Support Vector Machines (SVM). Each of the hyperparameters of the model was tuned by 4-fold cross validation and tested in a final unseen set of sentences. Overall both classification models showed good generalization on the test set, suggesting that the approach is reliable for

identifying check-worthy statements, which is the subtask in fact checking. Weighted sentence embeddings contribute more to classification, followed by Part-of-Speech (POS) tags and Named Entities (NE). Classification showed significant improvement when using all feature sets, compared to excluding context features. Considering the small dataset, the number of context features was kept low to avoid overfitting. However, in a larger dataset, it would be interesting to include more context features and see how the approach and classification models perform. The same feature set was extracted from all sentences without distinguishing whether they were from a debate or a speech, in order to create a general approach, that is not only based on the spoken language present in the dataset. This suggests that this feature set would be suitable for other types of datasets, even those containing written text. A high number of misclassifications were observed in both models, Support Vectors Machines (SVMs) and Feed Forward Neural Network (FNN). Error analysis showed that some of these misclassifications stem from inconsistencies in the dataset, as different fact-checking organizations have different criteria for selecting a statement for fact checking. These misclassifications could be avoided if the sentence is classified as check worthy or not based on a formal definition of fake news that is not yet available (Ahmed et al., 2021).

In **Chapters 7-8**, I automated the proposed technique with the web-based fact-checking application. To evaluate the approach, I collected 2146 claims and labeled them as fake, non-fake and unverified. In the next step, I separated the name, location, event, place and many other tags that can help us compare the statements in a claim and then compare them with the full claim body. When the user enters the claim it is compared to known facts from trusted sources that are already checked facts from mainstream news media for dual verification. I trained Feed Forward Neural Network (FNN) algorithm for this task because after several experiments I found that feed forward performed well compared to the other algorithms. After this comparison, I have evidence for the claim verdict and a clear aggregation about the claim in the form of a pie chart which is published in (Ahmed et al., 2022).

## 9.2 Future Directions

Fake news detection is a very hot topic and accordingly there is a great desire for solutions that can accurately detect fake content. There are several ways to extend the work presented

in this thesis, some of which have already been mentioned in this document. Here I will discuss some directions for future work. These directions are:

- **Fact Checking Dashboard Enhancement**: I introduced a fact-checking dashboard after combining classification and fact checking of check-worthy statements. I discussed details in the last chapter but it can be further extended to introduce a user-friendly dashboard. In the next phase, I would like to extend my work to other media platforms. In this thesis, my focus is on fake news detection and a combination of classification and fact checking together. My focus has been on mainstream news media (e.g., BBC, CNN, etc) but in the future these similar techniques and algorithms could be applied to social media platforms (e.g., Facebook, Twitter, etc).
- **Emotion Aware Approaches**: Further research on how isolating specific emotions can help improve classification techniques. The possibility of fine-grained emotion analysis should be explored, particularly in relation to shorter texts such as those included in the PHEME dataset.
- **Extraction of Sentiment from Text**: Sentiment and emotion-aware model-specific hyperparameter optimisation should be explored. This could improve the solution from this project by analyzing how different hyperparameters can be optimized taking into account sentiment and emotion awareness. Further work should also include analysis of sentence and sub-sentence level approaches.
- **Fake News Impact Prediction**: Predicting the impact of news on different areas of society is indeed a very valuable insight. In the future, impact prediction can also be complemented with my proposed automatic fact-checking application which could be helpful in identifying the targeted domain.
- **Multimodal Approach**: Using only text is not enough to create practical solutions for fake news detection. A broader approach that incorporates modalities such as image, video, and context attributes such as author, location, platform, etc. would be a much more practical approach. Such a model would be able to integrate the findings of this project into the text-domain.

Some of the features discussed in this thesis could be integrated into the automated fact-checking system which we have developed in the future. This is because it looks like all future research will be based on these features:

- **Time**: Perhaps, news items have a higher probability of being fake if they are initially repeated more often, because they are interesting, and are detected as fake over time, which reduces repetition or they are deleted from some websites.
- **Location**: Where did the news originate? Perhaps news has a higher probability of being fake if it is generated somewhere else rather than the place it is about (e.g. Trump writes about China or the Arabian States, news about Clinton originates in Russia).
- **Detect**: When the same news appears in other media or sources, we refer to it as stance detection.
- **News about news**: It is more likely that a news item is fake, if many people or sources say it is fake.
- **Quality**: Maybe, it is more probable that fake news does not have mentioned its sources; simply claiming something, whereas with real news the source is mentioned.

### 9.3 Concluding Remarks

Fake news detection is a real-world problem for different sectors of society which I have discussed in detail in Sections 1.1 and 4.1. The first contribution of the research is an overall approach to fake news detection as a contribution of classification and fact checking, where classification identifies potential fake news which is then further analyzed for fact checking. My second contribution is an approach that focuses on classification of statements into check-worthy and non-check-worthy, taking into account the context around a statement. I have developed an application that directly integrates various components of fact checking starting from the collection of check-worthy statements from mainstream news media sources, through information retrieval with credible sources. The developed system compares the statements and predicts the fact of the news and shows the aggregation of fake and not fake news. The experimental analysis shows very encouraging and improved performance.

There are many other interesting features that I discussed in Section 9.2 that can be incorporated into our automated tool for further improvement. From a scientific and analytical point of view, the work done in this thesis has been fulfilling and has met the requirements. I hope this system will perform strongly and help individuals and society because of its potential.

“A successful book is not made of what is *in* it, but of what is left *out* of it”

—Letter to Henry H. Rogers, 26–28 April 1897

## Bibliography

- Adamic, L. A., & Adar, E. (2003). Friends and neighbors on the Web. *Social Networks*, 25(3), 211–230. [https://doi.org/10.1016/S0378-8733\(03\)00009-1](https://doi.org/10.1016/S0378-8733(03)00009-1).
- Abdelzaher, T. F., Shin, K. G., & Bhatti, N. (2002). Performance guarantees for web server end-systems: A control-theoretical approach. *IEEE transactions on parallel and distributed systems*, 13(1), 80-96.
- Ahmed, H. (2017). Detecting Opinion Spam and Fake News Using N-gram Analysis and Semantic Similarity By Master Thesis, University of Victoria. Retrieved from [https://dspace.library.uvic.ca/bitstream/handle/1828/8796/Ahmed\\_Hadeer\\_Masc\\_2017.pdf](https://dspace.library.uvic.ca/bitstream/handle/1828/8796/Ahmed_Hadeer_Masc_2017.pdf).
- Ahmed S., Balla K., Hinkelmann K., Corradini F. (2021) Fact Checking: Detection of Check Worthy Statements Through Support Vector Machine and Feed Forward Neural Network. In: Arai K. (eds) *Advances in Information and Communication.FICC 2021*. Springer, Cham. [https://doi.org/10.1007/978-3-030-73103-8\\_37](https://doi.org/10.1007/978-3-030-73103-8_37).
- Ahmed, S., Hinkelmann, K., & Corradini, F. (2020). Development of Fake News Model Using Machine Learning through Natural Language Processing. *International Journal of Computer and Information Engineering*, 14(12), 454-460.
- Ahmed, S., Balla, K., Hinkelmann, K., & Corradini, F. (2021, April). Fact Checking: Detection of Check Worthy Statements Through Support Vector Machine and Feed Forward Neural Network. In *Future of Information and Communication Conference* (pp. 520-535). Springer, Cham.
- Ahmed S., Hinkelmann K., Corradini F. (2022) Fact Checking: An Automatic End to End Fact Checking System. In: Lahby M., Pathan AS.K., Maleh Y., Yafooz W.M.S. (eds) *Combating Fake News with Computational Intelligence Techniques. Studies in Computational Intelligence*, vol 1001. Springer, Cham. [https://doi.org/10.1007/978-3-030-90087-8\\_17](https://doi.org/10.1007/978-3-030-90087-8_17)
- Allcott, H., & Gentzkow, M. (2017). Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives*, 31(2), 211–236. <https://doi.org/10.1257/jep.31.2.211>
- Allcott, H., Gentzkow, M., & Yu, C. (2019). Trends in the diffusion of misinformation on social media. *Research and Politics*, 6(2). <https://doi.org/10.1177/2053168019848554>
- Allison, P. D. (2012). *Logistic regression using SAS: Theory and application*. SAS institute.
- Alsmadi, I., & O'Brien, M. J. (2020). Rating news claims: Feature selection and evaluation. *Mathematical Biosciences and Engineering*, 17(3), 1922–1939. <https://doi.org/10.3934/mbe.2020101>.
- Anastassiou, G. A. (2011). Multivariate hyperbolic tangent neural network approximation. *Computers & Mathematics with Applications*, 61(4), 809-821.
- Bajaj, S. (2017). “ The Pope Has a New Baby !” Fake News Detection Using Deep Learning, *Stanford CS224N*,1–8.
- Baly, R., Mohtarami, M., Glass, J., Márquez, L., Moschitti, A., & Nakov, P. (2018). Integrating stance detection and fact checking in a unified corpus. *NAACL HLT 2018 - Conference of the North American Chapter of the Association for Computational*



- Linguistics: Human Language Technologies - Proceedings of the Conference*, 2, 21–27. <https://doi.org/10.18653/v1/n18-2004>.
- Balla, k. Extracting Check-Worthy Statements - Towards Automated Fact-Checking. Master Thesis, MSc in Computer Science UNICAM & MSc in Business Information Systems (FHNW), Double Degree. Tutors: Prof. Dr. F. Corradini, Prof. Dr. K. Hinkelmann, 2019.
- Banerjee, S., Chua, A. Y. K., & Kim, J.-J. (2015). Using supervised learning to classify authentic and fake online reviews. *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication - IMCOM '15*, 1–7. <https://doi.org/10.1145/2701126.2701130>
- Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). Open information extraction from the web. *IJCAI International Joint Conference on Artificial Intelligence*, 2670–2676.
- Bennett, K. P., & Campbell, C. (2000). Support vector machines: hype or hallelujah?. *ACM SIGKDD explorations newsletter*, 2(2), 1-13.
- Benner, P. (2021). Computing leapfrog regularization paths with applications to large-scale k-mer logistic regression. *Journal of Computational Biology*, 28(6), 560-569.
- Benamira, A., Devillers, B., Lesot, E., Ray, A. K., Saadi, M., & Malliaros, F. D. (2019). Semi-supervised learning and graph neural networks for fake news detection. *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2019*, 568–569. <https://doi.org/10.1145/3341161.3342958>
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. NY Springer.
- Burfoot, C., & Baldwin, T. (2009). Automatic satire detection: Are you having a laugh? *ACL-IJCNLP 2009 - Joint Conf. of the 47th Annual Meeting of the Association for Computational Linguistics and 4th Int. Joint Conf. on Natural Language Processing of the AFNLP, Proceedings of the Conf.*, (August), 161–164.
- Burkhardt, J. M. (2017). Can We Save Ourselves? *Library Technology Reports*, 53(8), 22–28.
- Castillo, C., Mendoza, M., & Poblete, B. (2011a). Information credibility on Twitter. *Proceedings of the 20th International Conference Companion on World Wide Web, WWW 2011*, 675–684. <https://doi.org/10.1145/1963405.1963500>
- Castillo, C., Mendoza, M., & Poblete, B. (2011b). Information credibility on Twitter. *Proceedings of the 20th International Conference Companion on World Wide Web, WWW 2011*, (May 2014), 675–684. <https://doi.org/10.1145/1963405.1963500>
- Chatzimilioudis, G., Konstantinidis, A., Laoudias, C., & Zeinalipour-Yazti, D. (2012). Crowdsourcing with smartphones. *IEEE Internet Computing*, 16(5), 36–44. <https://doi.org/10.1109/MIC.2012.70>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16(February 2017), 321–357. <https://doi.org/10.1613/jair.953>
- Chaovalit, P., & Zhou, L. (2005, January). Movie review mining: A comparison between supervised and unsupervised classification approaches. In *Proceedings of the 38th annual Hawaii international conference on system sciences* (pp. 112c-112c). IEEE.

- Cherubini, F., & Graves, L. (2016). The rise of fact-checking sites in Europe. Reuters Institute for the Study of Journalism, University of Oxford.
- Chen, Y., Conroy, N. J., & Rubin, V. L. (2015). Misleading online content: Recognizing clickbait as “false news.” *WMDD 2015 - Proceedings of the ACM Workshop on Multimodal Deception Detection, Co-Located with ICMI 2015*, 15–19. <https://doi.org/10.1145/2823465.2823467>.
- Chitrakar, R., & Huang, C. (2014). Selection of candidate support vectors in incremental SVM for network intrusion detection. *computers & security*, 45, 231-241.
- Ciampaglia, G. L., Shiralkar, P., Rocha, L. M., Bollen, J., Menczer, F., & Flammini, A. (2015a). Computational fact checking from knowledge networks. *PLoS ONE*, 10(6), 1–13. <https://doi.org/10.1371/journal.pone.0128193>
- Ciampaglia, G. L., Shiralkar, P., Rocha, L. M., Bollen, J., Menczer, F., & Flammini, A. (2015b). Computational fact checking from knowledge networks. *PLoS ONE*, 10(6), 1–20. <https://doi.org/10.1371/journal.pone.0128193>
- Ciampaglia, G. L., Shiralkar, P., Rocha, L. M., Bollen, J., Menczer, F., & Flammini, A. (2015c). Computational fact checking from knowledge networks. *PLoS ONE*, 10(6), 1–8. <https://doi.org/10.1371/journal.pone.0128193>
- Conroy, N. J., Rubin, V. L., & Chen, Y. (2015a). Automatic deception detection: Methods for finding fake news. *Proceedings of the Association for Information Science and Technology*, 52(1), 1–4. <https://doi.org/10.1002/pra2.2015.145052010082>
- Conroy, N. J., Rubin, V. L., & Chen, Y. (2015b). Automatic deception detection: Methods for finding fake news. *Proceedings of the Association for Information Science and Technology*, 52(1), 1–4. <https://doi.org/10.1002/pra2.2015.145052010082>
- Conroy, N. J., Rubin, V. L., & Chen, Y. (2015c). Automatic Deception Detection: Methods for Finding Fake News. *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*, (October), 82. <https://doi.org/10.1002/pra2.2015.145052010082>
- Cook, D., Waugh, B., Abdipanah, M., Hashemi, O., & Abdul Rahman, S. (2014). Twitter Deception And Influence: Issues Of Identity, Slacktivism, And Puppetry. *Journal of Information Warfare*, 13(1), 58–71.
- Davuth, N., & Kim, S. R. (2013). Classification of malicious domain names using support vector machine and bi-gram method. *International Journal of Security and Its Applications*, 7(1), 51-58.
- Del Vicario, M., Quattrociocchi, W., Scala, A., & Zollo, F. (2019). Polarization and fake news: Early warning of potential misinformation targets. *ACM Transactions on the Web*, 13(2). <https://doi.org/10.1145/3316809>
- Dey, K., Shrivastava, R., & Kaushik, S. (2017). Twitter Stance Detection - A Subjectivity and Sentiment Polarity Inspired Two-Phase Approach. *IEEE International Conference on Data Mining Workshops, ICDMW, 2017-Novem*, 365–372. <https://doi.org/10.1109/ICDMW.2017.53>
- Dikici, E., Semerci, M., Saraclar, M., & Alpaydin, E. (2013). Classification and ranking approaches to discriminative language modeling for ASR. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(2), 291-300.
- Dongare, A. D., Kharde, R. R., & Kachare, A. D. (2012). Introduction to artificial neural network. *International Journal of Engineering and Innovative Technology (IJEIT)*, 2(1),

189-194.

- Du, J., Xu, R., He, Y., & Gui, L. (2017). Stance classification with target-specific neural attention networks. *IJCAI International Joint Conference on Artificial Intelligence*, 3988–3994. <https://doi.org/10.24963/ijcai.2017/557>
- Elizondo, D. (2006). The linear separability problem: Some testing methods. *IEEE Transactions on Neural Networks*, 17(2), 330–344. <https://doi.org/10.1109/TNN.2005.860871>.
- Ellis, T. J., & Levy, Y. (2010). A Guide for Novice Researchers: Design and Development Research Methods. *Proceedings of the 2010 InSITE Conference*, (December), 107–118. <https://doi.org/10.28945/1237>
- Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22, 457–479. <https://doi.org/10.1613/jair.1523>
- Fan, C. (2017). Classifying Fake News, 1–14. Retrieved from <http://www.conniefan.com/wp-content/uploads/2017/03/classifying-fake-news.pdf>
- Feng, V. W., & Hirst, G. (2013). Detecting Deceptive Opinions with Profile Compatibility. *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, (October), 338–346. Retrieved from <http://www.aclweb.org/anthology/I13-1039>.
- Feng, G., Guo, J., Jing, B. Y., & Sun, T. (2015). Feature subset selection using naive Bayes for text classification. *Pattern Recognition Letters*, 65, 109-115.
- Figueira, Á., & Oliveira, L. (2017). The current state of fake news: Challenges and opportunities. *Procedia Computer Science*, 121(December), 817–825. <https://doi.org/10.1016/j.procs.2017.11.106>
- Flintham, M., Karner, C., Bachour, K., Creswick, H., Gupta, N., & Moran, S. (2018). Falling for Fake News : Investigating the Consumption of News via Social Media. <https://doi.org/10.1145/3173574.3173950>
- F. Sebastiani. Machine learning in automated text categorization. Technical Report Technical Report IEI-B4-31-1999, Consiglio Nazionale delle Ricerche, Pisa, Italy, 1999.
- Gencheva, P., Nakov, P., Marquez, L., Barrón-Cedeno, A., & Koychev, I. (2017). A context-aware approach for detecting worth-checking claims in political debates. *International Conference Recent Advances in Natural Language Processing, RANLP, 2017-Septe*, 267–276. <https://doi.org/10.26615/978-954-452-049-6-037>.
- Gilda, S. (2017, December). Evaluating machine learning algorithms for fake news detection. In 2017 IEEE 15th Student Conference on Research and Development (SCoReD) (pp. 110-115). IEEE.
- Goldani, M. H., Momtazi, S., & Safabakhsh, R. (2020). Detecting Fake News with Capsule Neural Networks. Retrieved from <http://arxiv.org/abs/2002.01030>.
- Gruppi, M., Horne, B. D., & Adalı, S. (2020). NELA-GT-2019: A Large Multi-Labelled News Dataset for the Study of Misinformation in News Articles. arXiv preprint arXiv:2003.08444.
- G. Salton and M. J. McGill. An Introduction to Modern Information Retrieval. McGraw-Hill, 1983.
- Guha, S. (2017). Related Fact Checks: a tool for combating fake news. Retrieved from

<https://arxiv.org/abs/1711.00715>

- Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *Annals of statistics*, 28(2), 337-407.
- Haigh, M., Haigh, T., & Kozak, N. I. (2017). Stopping Fake News: The work practices of peer-to-peer counter propaganda. *Journalism Studies*, (April), 1–26. <https://doi.org/10.1080/1461670X.2017.1316681>
- Hancock, J. T., Thom-Santelli, J., & Ritchie, T. (2004). Deception and design. *CHI 2004, April 24–29, 2004, Viena, Austria*, 129–134. <https://doi.org/10.1145/985692.985709>
- Handl, A. (2010). Multivariate Analysemethoden. *Multivariate Analysemethoden*, (December 2014). <https://doi.org/10.1007/978-3-642-14987-0>
- Hassan, N., Li, C., & Tremayne, M. (2015). Detecting check-worthy factual claims in presidential debates. *International Conference on Information and Knowledge Management, Proceedings, 19-23-Oct-*, 1835–1838. <https://doi.org/10.1145/2806416.2806652>
- Hassan, N., Zhang, G., Arslan, F., Caraballo, J., & Jimenez, D. (2017). ClaimBuster : The First-ever End-to-end Fact-checking System, 1945–1948.
- Heaton, J. (2018). Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning. *Genetic Programming and Evolvable Machines*, 19(1), 305–307. <https://doi.org/10.1007/s10710-017-9314-z>
- Heinrich, C. U., & Borkenau, P. (n.d.). Deception and Deception Detection : The Role of Cross-Modal Inconsistency, (October 1998).
- Hevner, A., & Chatterjee, S. (2010). Design Science Research in Information Systems. *Springer, Boston, MA*. <https://doi.org/10.1007/978-1-4419-5653-8>
- Hevner Alan, R. (2007). A Three Cycle View of Design Science Research. *Scandinavian Journal of Information Systems*, 19(2), 87–92.
- Himma-Kadakas, M. (2017). Alternative facts and fake news entering journalistic content production cycle. *Cosmopolitan Civil Societies*. <https://doi.org/10.5130/ccs.v9i2.5469>.
- Kwak, H., Lee, C., Park, H., & Moon, S. (2010, April). What is Twitter, a social network or a news media?. In Proceedings of the 19th international conference on World wide web (pp. 591-600).
- Houvardas, J., & Stamatatos, E. (2006). N-gram feature selection for authorship identification. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 4183 LNCS, 77–86. [https://doi.org/10.1007/11861461\\_10](https://doi.org/10.1007/11861461_10)
- Janze, C., & Risius, M. (2017). Automatic Detection of Fake News on Social Media Platforms, 261. Retrieved from <http://aisel.aisnet.org/pacis2017/261>
- Jaradat, I., Gencheva, P., Barrón-Cedeño, A., Márquez, L., & Nakov, P. (2018). ClaimRank: Detecting Check-Worthy Claims in Arabic and English, 26–30. <https://doi.org/10.18653/v1/n18-5006>
- Java, A., Song, X., Finin, T., & Tseng, B. (2007). Why we twitter: Understanding microblogging usage and communities. *Joint Ninth WebKDD and First SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis*, 56–65. <https://doi.org/10.1145/1348549.1348556>

- Hancock, J. T., Thom-Santelli, J., & Ritchie, T. (2004, April). Deception and design: The impact of communication technology on lying behavior. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 129-134).
- Jia, Y., Wang, Y., Lin, H., Jin, X., & Cheng, X. (2016). Locally adaptive translation for knowledge graph embedding. *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, 992–998.
- Jindal, N., & Liu, B. (2008, February). Opinion spam and analysis. In Proceedings of the 2008 international conference on web search and data mining (pp. 219-230).
- Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1398, 137–142. <https://doi.org/10.1007/s13928716>
- Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects, 349(6245).
- Julian, K., & Lu, W. (2016). Application of machine learning to link prediction 1-5.
- Kahneman, D. (2013). A perspective on judgment and choice: Mapping bounded rationality. *Progress in Psychological Science around the World. Volume 1 Neural, Cognitive and Developmental Issues: Congress Proceedings: XVIII International Congress of Psychology, Beijing, 2004*, 1–47. <https://doi.org/10.4324/9780203783122>
- Karadzhov, G., Nakov, P., Màrquez, L., Barrón-Cedeño, A., & Koychev, I. (2017a). Fully automated fact checking using external sources. *International Conference Recent Advances in Natural Language Processing, RANLP, 2017-Septe*(February 2018), 344–353. <https://doi.org/10.26615/978-954-452-049-6-046>
- Karadzhov, G., Nakov, P., Màrquez, L., Barrón-Cedeño, A., & Koychev, I. (2017b). Fully automated fact checking using external sources. *International Conference Recent Advances in Natural Language Processing, RANLP, 2017-Septe*, 344–353. <https://doi.org/10.26615/978-954-452-049-6-046>
- Katz, L (1953). A new status index derived from sociometric analysis. *Psychometrika* 18, 39–43. <https://doi.org/10.1007/BF02289026>
- Keskar, D., Palwe, S., & Gupta, A. (2020). Fake News Classification on Twitter Using Flume, N-Gram Analysis, and Decision Tree Machine Learning Technique. In Proceeding of International Conference on Computational Science and Applications (pp. 139-147). Springer, Singapore.
- Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–15.
- Kratzert, F., Klotz, D., Brenner, C., Schulz, K., & Herrnegger, M. (2018). Rainfall–runoff modelling using long short-term memory (LSTM) networks. *Hydrology and Earth System Sciences*, 22(11), 6005-6022.
- Kohavi, R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *International Joint Conference of Artificial Intelligence*, (June).
- Kostakos, P., Nykanen, M., Martinviita, M., Pandya, A., & Oussalah, M. (2018). Meta-terrorism: identifying linguistic patterns in public discourse after an attack. In 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) (pp. 1079-1083). IEEE.

- Kumar, S., West, R., & Leskovec, J. (2016). Disinformation on the web: Impact, characteristics, and detection of wikipedia hoaxes. *25th International World Wide Web Conference, WWW 2016*, 591–602. <https://doi.org/10.1145/2872427.2883085>
- Kwak, H., Lee, C., Park, H., & Moon, S. (2010). What is Twitter, a social network or a news media? *Proceedings of the 19th International Conference on World Wide Web, WWW '10*, 591–600. <https://doi.org/10.1145/1772690.1772751>
- Lao, N., & Cohen, W. W. (2010). Relational retrieval using a combination of path-constrained random walks. *Machine Learning*, *81*(1), 53–67. <https://doi.org/10.1007/s10994-010-5205-8>
- Lazer, D. M. J., Baum, M. A., Benkler, Y., Berinsky, A. J., Greenhill, K. M., Menczer, F., ... Zittrain, J. L. (2018a). The science of fake news. *Science*, *359*(6380), 1094–1096. <https://doi.org/10.1126/science.aao2998>
- Lazer, D. M. J., Baum, M. A., Benkler, Y., Berinsky, A. J., Greenhill, K. M., Menczer, F., ... Zittrain, J. L. (2018b). The science of fake news. *Science*, *359*(6380), 1094–1096. <https://doi.org/10.1126/science.aao2998>.
- Leonard, D., & Sensiper, S. (1998). The role of tacit knowledge in group innovation. *California management review*, *40*(3), 112–132.
- Le, Q. V. (2015). A tutorial on deep learning part 1: Nonlinear classifiers and the backpropagation algorithm. Google Inc., Mountain View, CA, 18.
- Li, Y., Gao, J., Meng, C., Li, Q., Su, L., Zhao, B., ... Han, J. (2016). A Survey on Truth Discovery. *ACM SIGKDD Explorations Newsletter*, *17*(2), 1–16. <https://doi.org/10.1145/2897350.2897352>
- Liu, D., Wang, Y., Jia, Y., Li, J., & Yu, Z. (2014). LSDH: A hashing approach for large-scale link prediction in microblogs. *Proceedings of the National Conference on Artificial Intelligence*, *4*, 3120–3121.
- Liu, X., Nielek, R., Adamska, P., Wierzbicki, A., & Aberer, K. (2015). Towards a highly effective and robust Web credibility evaluation system. *Decision Support Systems*, *79*, 99–108. <https://doi.org/10.1016/j.dss.2015.07.010>
- Lin, M., Chen, Q., & Yan, S. (2013). Network in network. arXiv preprint arXiv:1312.4400.
- Lorent, S., & Itoo, A. (2019). Fake News Detection Using Machine Learning Author : Simon Lorent Supervisor : Ashwin Itoo A thesis presented for the degree of Master in Data Science University of Liege, Belgium.
- Lukasik, M., Cohn, T., & Bontcheva, K. (2015). Classifying tweet level judgements of rumours in social media. *Conference Proceedings - EMNLP 2015: Conference on Empirical Methods in Natural Language Processing*, (September), 2590–2595. <https://doi.org/10.18653/v1/d15-1311>.
- Macías-Escrivá, F. D., Haber, R., Del Toro, R., & Hernandez, V. (2013). Self-adaptive systems: A survey of current approaches, research challenges and applications. *Expert Systems with Applications*, *40*(18), 7267–7279.
- Magdy, A., & Wanas, N. (2010). Web-based statistical fact checking of textual documents. *International Conference on Information and Knowledge Management, Proceedings*, 103–109. <https://doi.org/10.1145/1871985.1872002>.
- Majithia, S., Arslan, F., Lubal, S., Jimenez, D., Arora, P., Caraballo, J., & Li, C. (2019, July). ClaimPortal: Integrated monitoring, searching, checking, and analytics of factual claims

- on twitter. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (pp. 153-158).
- March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision support systems*, 15(4), 251-266.
- Masood, F., Ammad, G., Almogren, A., Abbas, A., Khattak, H. A., Ud Din, I., ... Zuair, M. (2019). Spammer Detection and Fake User Identification on Social Networks. *IEEE*, 68140–68152. <https://doi.org/10.1109/ACCESS.2019.2918196>.
- Mallya, A., Davis, D., & Lazebnik, S. (2018). Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 67-82).
- Meel, P., & Vishwakarma, D. K. (2020). Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. *Expert Systems with Applications*, 153, 112986.
- Mihaylova, T., Nakov, P., Márquez, L., Barrón-Cedeño, A., Mohtarami, M., Karadzhov, G., & Glass, J. (2018). Fact checking in community forums. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 5309–5316.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 1–9.
- Mohammad, S. M., Sobhani, P., & Kiritchenko, S. (2017). *Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA*, 17(3).
- Moran, R. E. (2018). Deciding what's true: The rise of political fact-checking in American journalism.
- Mazurowski, M. A., Habas, P. A., Zurada, J. M., Lo, J. Y., Baker, J. A., & Tourassi, G. D. (2008). Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural networks*, 21(2-3), 427-436.
- Naaman, M., Boase, J., & Lai, C. H. (2010). Is it really about me?: Message content in social awareness streams. *Proceedings of the ACM Conference on Computer Supported Cooperative Work, CSCW*, 189–192. <https://doi.org/10.1145/1718918.1718953>.
- Naseem, U., Razzak, I., Khan, S. K., & Prasad, M. (2021). A comprehensive survey on word representation models: From classical to state-of-the-art word representation language models. *Transactions on Asian and Low-Resource Language Information Processing*, 20(5), 1-35.
- Nakashole, N., & Mitchell, T. M. (2014). Language-aware truth assessment of fact candidates. *52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 - Proceedings of the Conference, 1*, 1009–1019. <https://doi.org/10.3115/v1/p14-1095>.
- Nguyen, A. T., Kharosekar, A., Lease, M., & Wallace, B. C. (2018). An interpretable joint graphical model for fact-checking from crowds. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, (3), 1511–1518.
- Ng, S. S., Xing, Y., & Tsui, K. L. (2014). A naive Bayes model for robust remaining useful life prediction of lithium-ion battery. *Applied Energy*, 118, 114-123.
- Nidhi, & Gupta, V. (2011). Recent Trends in Text Classification Techniques. *International Journal of Computer Applications*, 35(6), 45–51.

- Nørregaard, J., Horne, B. D., & Adalı, S. (2019). NELA-GT-2018: A large multi-labelled news dataset for the study of misinformation in news articles. *Proceedings of the 13th International Conference on Web and Social Media, ICWSM 2019*, 630–638.
- Noriega, L. (2005). Multilayer perceptron tutorial. School of Computing, Staffordshire University.
- Oshikawa, R., Qian, J., & Wang, W. Y. (2018). A Survey on Natural Language Processing for Fake News Detection. Retrieved from <http://arxiv.org/abs/1811.00770>.
- Ott, M., Choi, Y., Cardie, C., & Hancock, J. T. (2011, June). Finding deceptive opinion spam by any stretch of the imagination. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1* (pp. 309-319). Association for Computational Linguistics.
- Parikh, S. B., & Atrey, P. K. (2018). Media-Rich Fake News Detection: A Survey. *Proceedings - IEEE 1st Conference on Multimedia Information Processing and Retrieval, MIPR 2018*, (November), 436–441. <https://doi.org/10.1109/MIPR.2018.00093>
- Patwari, A., Goldwasser, D., & Bagchi, S. (2017). Tathya: A multi-classifier system for detecting check-worthy statements in political debates. *International Conference on Information and Knowledge Management, Proceedings, Part F1318*, 2259–2262. <https://doi.org/10.1145/3132847.3133150>.
- Pear. Analytics. Twitter study–august 2009. San Antonio, TX: Pear Analytics. Available at: [www.pearanalytics.com/blog/wp-content/uploads/2010/05/Twitter-Study-August-2009](http://www.pearanalytics.com/blog/wp-content/uploads/2010/05/Twitter-Study-August-2009).
- Pavleska, T., Školokay, A., Zankova, B., Ribeiro, N., & Bechmann, A. (2018). Performance analysis of fact-checking organizations and initiatives in Europe: a critical overview of online platforms fighting fake news. *Social media and convergence*, 29.
- Pfeffers, K., Tuunanen, T., Gengler, C. E., Rossi, M., Hui, W., Virtanen, V., & Bragge, J. (2006). The design science research process: A model for producing and presenting information systems research. In *Proceedings of the First International Conference on Design Science Research in Information Systems and Technology (DESRIST 2006)*, Claremont, CA, USA (pp. 83-106).
- Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 1532–1543. <https://doi.org/10.3115/v1/d14-1162>.
- Pennycook, G., & Rand, D. G. (2019). Fighting misinformation on social media using crowdsourced judgments of news source quality. *Proceedings of the National Academy of Sciences of the United States of America*, 116(7), 2521–2526. <https://doi.org/10.1073/pnas.1806781116>.
- Popat, K., Mukherjee, S., Strötgen, J., & Weikum, G. (2016). Credibility assessment of textual claims on the web. *International Conference on Information and Knowledge Management, Proceedings, 24-28-Octo*, 2173–2178. <https://doi.org/10.1145/2983323.2983661>
- Pothast, M., Hagen, M., Gollub, T., Tippmann, M., Kiesel, J., Rosso, P., ... Stein, B. (2013). *Overview of the 5th international competition on plagiarism detection. CEUR Workshop Proceedings* (Vol. 1179).
- Pothast, M., Köpsel, S., Stein, B., & Hagen, M. (2016, March). Clickbait detection. In *European Conference on Information Retrieval* (pp. 810-817). Springer, Cham.



- Pratiwi, I. Y. R., Asmara, R. A., & Rahutomo, F. (2017, October). Study of hoax news detection using naïve bayes classifier in Indonesian language. In 2017 11th International Conference on Information & Communication Technology and System (ICTS) (pp. 73-78). IEEE.
- Prashant Shiralkar, Alessandro Flammini, Filippo Menczer, and Giovanni Luca Ciampaglia. 2017. Finding streams in knowledge graphs to support fact checking. arXiv preprint arXiv:1708.07239.
- Qazvinian, V., Rosengren, E., Radev, D. R., & Mei, Q. (2011). Rumor has it: Identifying misinformation in microblogs. *EMNLP 2011 - Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 1589–1599.
- Qin, Y., Wurzer, D., & Tang, C. (2018). Predicting future rumours. *Chinese Journal of Electronics*, 27(3), 514-520.
- Rashkin, H., Choi, E., Jang, J. Y., Volkova, S., & Choi, Y. (2017). Truth of varying shades: Analyzing language in fake news and political fact-checking. *EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, 2931–2937. <https://doi.org/10.18653/v1/d17-1317>
- Rasmus. (2019). *allport.pdf*.
- Ramy Baly, Georgi Karadzhov, Dimitar Alexandrov, James Glass, and Preslav Nakov. 2018a. Predicting factuality of reporting and bias of news media sources. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 3528–3539. Association for Computational Linguistics.
- Raulji, J. K., & Saini, J. R. (2016). Stop-word removal algorithm and its implementation for Sanskrit language. *International Journal of Computer Applications*, 150(2), 15-17.
- Rao, G., Huang, W., Feng, Z., & Cong, Q. (2018). LSTM with sentence representations for document-level sentiment classification. *Neurocomputing*, 308, 49-57.
- Ripley, B. D. (1994). Neural networks and related methods for classification. *Journal of the Royal Statistical Society: Series B (Methodological)*, 56(3), 409-437.
- Rubin, V., Conroy, N., Chen, Y., & Cornwell, S. (2016a). Fake News or Truth? Using Satirical Cues to Detect Potentially Misleading News. *Proceedings of the Second Workshop on Computational Approaches to Deception Detection*, 7–17. <https://doi.org/10.18653/v1/W16-0802>.
- Rubin, V., Conroy, N., Chen, Y., & Cornwell, S. (2016b). Fake News or Truth? Using Satirical Cues to Detect Potentially Misleading News. In *Proceedings of NAACL-HLT 2016, pages 7–17, San Diego, California, June 12-17, 2016. c?2016 Association for Computational Linguistics* (pp. 7–17). <https://doi.org/10.18653/v1/w16-0802>.
- Rubin, V. L., Conroy, N. J., & Chen, Y. (2015, January). Towards news verification: Deception detection methods for news discourse. In *Hawaii International Conference on System Sciences* (pp. 5-8).
- Rubin, V. L., & Vashchilko, T. (2012, April). Identification of truth and deception in text: Application of vector space model to rhetorical structure theory. In *Proceedings of the Workshop on Computational Approaches to Deception Detection* (pp. 97-106).
- Ruchansky, N., Seo, S., & Liu, Y. (2017). CSI: A hybrid deep model for fake news detection. *International Conference on Information and Knowledge Management, Proceedings, Part F1318*, 797–806. <https://doi.org/10.1145/3132847.3132877>.
- Samonte, M. J. C. (2018). Polarity analysis of editorial articles towards fake news detection. *ACM International Conference Proceeding Series*, 108–112.

<https://doi.org/10.1145/3230348.3230354>

- Savalia, S., & Emamian, V. (2018). Cardiac arrhythmia classification by multi-layer perceptron and convolution neural networks. *Bioengineering*, 5(2), 35.
- Sharma, Y., Agrawal, G., Jain, P., & Kumar, T. (2017, December). Vector representation of words for sentiment analysis using GloVe. In 2017 international conference on intelligent communication and computational techniques (icct) (pp. 279-284). IEEE.
- Shi, B., & Weninger, T. (2016). Discriminative predicate path mining for fact checking in knowledge graphs. *Knowledge-Based Systems*, 104, 123–133.  
<https://doi.org/10.1016/j.knosys.2016.04.015>.
- Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake News Detection on Social Media. *ACM SIGKDD Explorations Newsletter*, 19(1), 22–36.  
<https://doi.org/10.1145/3137597.3137600>.
- Silva, R. M., Santos, R. L., Almeida, T. A., & Pardo, T. A. (2020). Towards automatically filtering fake news in Portuguese. *Expert Systems with Applications*, 146, 113199.
- Silverman, C. (2016). This analysis shows how viral fake election news stories outperformed real news on Facebook. *BuzzFeed news*, 16.
- Singh, V., Dasgupta, R., Sonagra, D., Raman, K., & Ghosh, I. (2017). Automated Fake News Detection Using Linguistic Analysis and Machine Learning. *Proceedings of the 2017 International Conference on Social Computing, Behavioural-Cultural Modeling* (July), 1–3. <https://doi.org/10.13140/RG.2.2.16825.67687>.
- Singhania, S., Fernandez, N., & Rao, S. (2017, November). 3han: A deep neural network for fake news detection. In *International conference on neural information processing* (pp. 572-581). Springer, Cham.
- Siponen, M., & Stucke, C. (2006). Effective anti-spam strategies in companies: An international study. *Proceedings of the Annual Hawaii International Conference on System Sciences*, 6(February). <https://doi.org/10.1109/HICSS.2006.140>
- Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science* 359(6380):1146–1151.
- Stassen, W. (2011). Your news in 140 characters: exploring the role of social media in journalism. *Global Media Journal African Edition*, 4(1), 116–131.  
<https://doi.org/10.5789/4-1-15>.
- Sternberg, R. J. (1986). *Critical Thinking: Its Nature, Measurement, and Improvement*, ERIC Number ED272882.
- Sterrett, D., Malato, D., Benz, J., Kantor, L., Tompson, T., Rosenstiel, T., ... Loker, K. (2019). Who Shared It?: Deciding What News to Trust on Social Media. *Digital Journalism*, 7(6), 783–801. <https://doi.org/10.1080/21670811.2019.1623702>.
- Sun, A., Lim, E. P., & Liu, Y. (2009). On strategies for imbalanced text classification using SVM: A comparative study. *Decision Support Systems*, 48(1), 191–201.  
<https://doi.org/10.1016/j.dss.2009.07.011>.
- Svozil, D., Kvasnicka, V., & Pospichal, J. (1997). Introduction to multi-layer feed-forward neural networks. *Chemometrics and intelligent laboratory systems*, 39(1), 43-62.
- Swol, L. M. Van, & Braun, M. T. (2014). Channel Choice , Justification of Deception , and Detection, *Journal of Communication*, 64(6), 1139-1159, 1–21.  
<https://doi.org/10.1111/jcom.12125>.

- Tacchini, E., Ballarin, G., Della Vedova, M. L., Moret, S., & de Alfaro, L. (2017). Some like it Hoax: Automated fake news detection in social networks. *CEUR Workshop Proceedings, 1960*, 1–12. <https://doi.org/10.1257/jep.31.2.211>.
- Tambuscio, M., Ruffo, G., Flammini, A., & Menczer, F. (n.d.). Fact-checking Effect on Viral Hoaxes: A Model of Misinformation Spread in Social Networks, *15*, 977–982. <https://doi.org/10.1145/2740908.2742572>.
- Tandoc, E. C., Lim, Z. W., & Ling, R. (2018). Defining “Fake News”: A typology of scholarly definitions. *Digital Journalism*, *6*(2), 137–153. <https://doi.org/10.1080/21670811.2017.1360143>.
- Tian, Y., & Pan, L. (2015, December). Predicting short-term traffic flow by long short-term memory recurrent neural network. In 2015 IEEE international conference on smart city/SocialCom/SustainCom (SmartCity) (pp. 153-158). IEEE.
- Thorne, J., Vlachos, A., Christodoulopoulos, C., & Mittal, A. (2018). FEVER: a Large-scale Dataset for Fact Extraction and Verification, 809–819. <https://doi.org/10.18653/v1/n18-1074>.
- Todor Mihaylov and Preslav Nakov. 2016. Hunting for troll comments in news community forums. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 399–405, Berlin, Germany
- Torunoğlu, D., Çakirman, E., Ganiz, M. C., Akyokuş, S., & Gürbüz, M. Z. (2011). Analysis of preprocessing methods on classification of Turkish texts. In 2011 International Symposium on Innovations in Intelligent Systems and Applications (pp. 112-117). IEEE.
- Tschiatschek, S., Singla, A., Gomez Rodriguez, M., Merchant, A., & Krause, A. (2018a). Fake News Detection in Social Networks via Crowd Signals, 517–524. <https://doi.org/10.1145/3184558.3188722>.
- Tschiatschek, S., Singla, A., Gomez Rodriguez, M., Merchant, A., & Krause, A. (2018b). Fake News Detection in Social Networks via Crowd Signals. (*International World Wide Web Conference Committee*, 517–524. <https://doi.org/10.1145/3184558.3188722>).
- Vedova, M. L. Della, Tacchini, E., Moret, S., Ballarin, G., & Dipierro, M. (2018). Content and Social Signals. *2018 22nd Conference of Open Innovations Association (FRUCT)*, 272–279.
- Vijayaraghavan, S., Wang, Y., Guo, Z., Voong, J., Xu, W., Nasser, A., ... Wadhwa, E. (2020). Fake News Detection with Different Models. arxiv preprint arxiv:2003.04978.
- Vlachos, A., & Riedel, S. (2014). Fact Checking: Task definition and dataset construction. *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, 18–22. <https://doi.org/10.3115/v1/W14-2508>.
- Vosoughi, S., Roy, D., & Aral, S. (2018). News On-line. *Science*, *1151*(March), 1146–1151.
- Vrij, A., & Granhag, P. A. (2012). Eliciting cues to deception and truth: What matters are the questions asked. *Journal of Applied Research in Memory and Cognition*, *1*(2), 110–117. <https://doi.org/10.1016/j.jarmac.2012.02.004>.
- Wang, W., Do, D. B., & Lin, X. (2005, July). Term graph model for text classification. In International Conference on Advanced Data Mining and Applications (pp. 19-30). Springer, Berlin, Heidelberg.
- Weber, K., & Alcock, L. (2004). Semantic and syntactic proof productions. Educational

- studies in mathematics, 56(2-3), 209-234.
- Wu, L., & Liu, H. (2018a). Tracing fake-news footprints: Characterizing social media messages by how they propagate. *WSDM 2018 - Proceedings of the 11th ACM International Conference on Web Search and Data Mining, 2018-February*, 637–645. <https://doi.org/10.1145/3159652.3159677>.
- Wu, L., & Liu, H. (2018b). Tracing fake-news footprints: Characterizing social media messages by how they propagate. *WSDM 2018-Proceedings of the 11th ACM International Conference on Web Search and Data Mining, 2018-February*(February), 637–645. <https://doi.org/10.1145/3159652.3159677>.
- Wu, Y., Agarwal, P. K., Li, C., Yang, J., & Yu, C. (2014). Toward computational fact-checking. *Proceedings of the VLDB Endowment*, 7(7), 589–600. <https://doi.org/10.14778/2732286.2732295>.
- Brian Xu, Mitra Mohtarami, and James Glass. 2018. Adversarial domain adaptation for stance detection. In *Proceedings of the Thirty-second Annual Conference on Neural Information Processing Systems (NeurIPS)–Continual Learning*
- Yazdi, K. M., Yazdi, A. M., Khodayi, S., Hou, J., Zhou, W., & Saedy, S. (2020). Improving Fake News Detection Using K-means and Support Vector Machine Approaches. *International Journal of Electronics and Communication Engineering*, 14(2), 38–42.
- Yan, H., Jiang, Y., Zheng, J., Peng, C., & Li, Q. (2006). A multilayer perceptron-based medical decision support system for heart disease diagnosis. *Expert Systems with Applications*, 30(2), 272-281.
- Zannettou, S., Sirivianos, M., Blackburn, J., & Kourtellis, N. (2019). The web of false information: Rumors, fake news, hoaxes, clickbait, and various other shenanigans. *Journal of Data and Information Quality*, 11(3). <https://doi.org/10.1145/3309699>.
- Zhang, J., Dong, B., & Yu, P. S. (2018). Fake Detector: Effective Fake News Detection with Deep Diffusive Neural Network, 1–13. Retrieved from <http://arxiv.org/abs/1805.08751>.
- Zeng, J., & Qiao, W. (2013). Short-term solar power prediction using a support vector machine. *Renewable Energy*, Volume 52, 118-127. <https://doi.org/10.1016/j.renene.2012.10.009>
- Zou, J., Han, Y., & So, S. S. (2008). Overview of artificial neural networks. *Artificial Neural Networks*, 14-22.
- Zhou, Xing, Cao, J., Jin, Z., Xie, F., Su, Y., Chu, D., ... Zhang, J. (2015). Real-Time News Certification System on Sina Weibo. *Proceedings of the 24th International Conference on World Wide Web - WWW '15 Companion*, 983–988. <https://doi.org/10.1145/2740908.2742571>.
- Zhou, X., & Zafarani, R. (2018). Fake news: A survey of research, detection methods, and opportunities. *arXiv preprint arXiv:1812.00315*, 2.
- Zhou, Xing, Cao, J., Jin, Z., Xie, F., Su, Y., Zhang, J., ... Cao, X. (2015a). Real-time news certification system on sina weibo. *WWW 2015 Companion - Proceedings of the 24th International Conference on World Wide Web*, 983–988. <https://doi.org/10.1145/2740908.2742571>.
- Zhou, Xinyi, & Zafarani, R. (2018). Fake News: A Survey of Research, Detection Methods, and Opportunities. *ACM*. Retrieved from <http://arxiv.org/abs/1812.00315>.
- Zubiaga, A., Aker, A., Bontcheva, K., Liakata, M., & Procter, R. (2018). Detection and

resolution of rumours in social media: A survey. *ACM Computing Surveys*, 51(2), 1–36.  
<https://doi.org/10.1145/3161603>.

## Appendix-A: Configuration of Fact Checking Query Submission

```

<style>
.content {
  width: 400px;
  height: 200px;

  position:absolute; /*it can be fixed too*/
  left:0; right:0;
  top:0; bottom:0;
  margin:auto;

  max-width:100%;
  max-height:100%;
  overflow:auto;
}
.box{
  position:relative;
  text-align:center;
  height: 200px;
  border-radius:15px;
  padding:10px20px65px;
  background-color:#fcfcfc;
  -webkit-transition: all1000msease;
  -moz-transition: all100msease;
  -ms-transition: all1000msease;
  -o-transition: all1000msease;
  transition: all1000msease;
  box-shadow:0px0px30pxrgba(0,0,0,0.15);
}

</style>
<html>
<head>
<!-- Latest compiled and minified CSS -->
<linkrel="stylesheet"href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

<!-- jQuery library -->
<scriptsrc="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js">
</script>

<!-- Latest compiled JavaScript -->
<scriptsrc="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
<title>Search with a keyword...</title>
</head>

```

```
<bodystyle="background-image:url(main.jpg); background-repeat:no-repeat;
background-size:cover">
<divclass="content">
  <divclass="box">
    <br><br>
    <divstyle="text-align:center"><h4>Type in keywords to search
<imgwidth="10"height="10"src="question.png"></h4></div>
    <formname="search"action="cgi-bin/search.py"method="get">
      <divclass="form-group">
        <inputtype="text"name="searchbox"class="form-
control"placeholder="Search for words.."/>
      </div>
      <divclass="form-group">
        <divstyle="text-
align:center"><buttontype="submit"value="Submit"class="btnbtn-
success">Search</button></div>
      </div>
    </form>
  </div>
</div>
</body>
</html>
```

## Appendix-B: Configuration of Fact checking application with Wikipedia and other news media organizations

```

#!C:\Users\ilear\AppData\Local\Programs\Python\Python37\python.exe
print("content-type: text/html\n\n" )

import requests
from bs4 import BeautifulSoup
from selenium import webdriver
from selenium.common.exceptions import TimeoutException
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.by import By
from lxml import html
import re
import os
import httpplib2
import cgi

linkArray= []
subjectArray= []

form =cgi.FieldStorage()
http =httpplib2.Http()
search = form["searchbox"].value
os.environ['NO_PROXY'] = '127.0.0.1'

results =40# valid options 10, 20, 30, 40, 50, and 100
page =requests.get(f"https://www.google.com/search?q={search}&num={results}")
soup =BeautifulSoup(page.content, "html.parser")
links =soup.findAll("a")
for link inlinks :
    link_href=link.get('href')
    if"url?q="inlink_hrefandnot"webcache"inlink_href:

        subject =link.text
        sub =subject.split("http")
        subjectArray.append(sub[0])
        linkArray.append(link.get('href').split("?q=")[1].split("&sa=U")[0])

data =""
subjectData=""
for link inlinkArray:
    data += link+";"

for sub insubjectArray:

```



```

    subjectData+= sub+";"

data =data[:-1]
subjectData=subjectData[:-1]

""" wikipedia """
wikiLinks= []
wikiSubjects= []
searchterm= search
wikiSearch=searchterm.replace(" ", "+")
urlwiki="https://en.wikipedia.org/w/index.php?cirrusUserTesting=glent_m0&sort=
relevance&search="+wikiSearch+"&title=Special%3ASearch&profile=advanced&fullte
xt=1&advancedSearch-current=%7B%7D&ns0=1"
r =requests.get(urlwiki)
content =r.content

soup =BeautifulSoup(content, 'html.parser')
for ul insoup.findAll("ul", {'class': 'mw-search-results'}):
    for li inul.findAll("li", {'class': 'mw-search-result'}):
        for a inli.findAll("a"):
            wikiLinks.append("https://en.wikipedia.org/"+a["href"])
            sob = a["href"].replace("/wiki/", "").replace("_", " ")
            wikiSubjects.append(sob)
            break

    date =ul.find("div", {'class': 'mw-search-result-data'})
    dateEx=date.text
    myArray=dateEx.split(" - ")

wikiLinksData=""
for link inwikiLinks:
    wikiLinksData+= link+";"

wikiSubjectsData=""
for link inwikiSubjects:
    wikiSubjectsData+= link+";"

""" CNN """
cnnLinks= []
cnnSubjects= []

cnnSearch=searchterm.replace(" ", "+")
urlcnn="https://edition.cnn.com/search?q="+cnnSearch

```

```

driver
=webdriver.Chrome(executable_path=r'C:\Users\ilear\.wdm\drivers\chromedriver\8
0.0.3987.106\win32\chromedriver.exe')
driver.get(urlcnn)
try:
    element_present=EC.presence_of_element_located((By.CLASS_NAME, 'cnn-
search__resultcnn-search__result--article'))
    WebDriverWait(driver, 1).until(element_present)
exceptTimeoutException:
    pass

content =driver.page_source

soup =BeautifulSoup(content, 'html.parser')
for div2 insoup.findAll("div", {'class': 'cnn-search__resultcnn-
search__result--article'}):
    for h3 in div2.findAll("h3", {'class': 'cnn-search__result-headline'}):
        link = h3.find("a")
        final_link= link["href"]
        cnnLinks.append(final_link[2:])
        cnnSubjects.append(link.text)
        break

cnnLinksData=""
for link incnnLinks:
    cnnLinksData+= link+";"

cnnSubjectsData=""
for link incnnSubjects:
    cnnSubjectsData+= link+";"

""" msnbc """
msnbcLinks= []
msnbcSubjects= []

msnbcSearch=searchterm.replace(" ", "%20")
urlmsnbc="http://www.msnbc.com/search/"+cnnSearch
driver.get(urlmsnbc)
try:
    element_present=EC.presence_of_element_located((By.CLASS_NAME, 'search-
result__teaser'))
    WebDriverWait(driver, 1).until(element_present)
exceptTimeoutException:
    pass

content =driver.page_source

```

```

dateEx= []
soup =BeautifulSoup(content, 'html.parser')
for div3 in soup.findAll("div", {'class': 'search-result__teaser'}):
    link = div3.find("a", {'class': 'search-result__teaser__title__link'})
    msnbcLinks.append("https://www.msnbc.com/"+link["href"])
    msnbcSubjects.append(link.text)

msnbcLinksData=""
for link in msnbcLinks:
    msnbcLinksData+= link+";"

msnbcSubjectsData=""
for link in msnbcSubjects:
    msnbcSubjectsData+= link+";"

""" nytimes """
nyLinks= []
nySubjects= []
nySearch=searchterm.replace(" ", "+")
urlny="https://www.nytimes.com/search?query="+cnnSearch
driver.get(urlny)
try:
    element_present=EC.presence_of_element_located((By.CLASS_NAME, 'css-
e1lvw9'))
    WebDriverWait(driver, 1).until(element_present)
except TimeoutException:
    pass

content =driver.page_source
soup =BeautifulSoup(content, 'html.parser')
for li in soup.findAll("div", {'class': 'css-e1lvw9'}):
    link =li.find("a")
    nyLinks.append("https://www.nytimes.com/"+link["href"])
    nySubjects.append(link.text)

nyLinksData=""
for link in nyLinks:
    nyLinksData+= link+";"

nySubjectsData=""
for link in nySubjects:
    nySubjectsData+= link+";"

finaldata= {"nyLinksData": nyLinksData, "nySubjectsData": nySubjectsData,
"data": data, "subject": subjectData, "wikiLinks": wikiLinksData,

```

```
"wikiSubjects": wikiSubjectsData, "cnnLinksData": cnnLinksData,  
"cnnSubjectsData": cnnSubjectsData, "msnbcLinksData": msnbcLinksData,  
"msnbcSubjectsData": msnbcSubjectsData}  
r =requests.post("http://localhost/pySearch/cgi-bin/viewdata.php",  
data=finaldata)  
print(r.text)
```

## Appendix-C: Search other news media sites for comparison

```

<?php

$totalStatements=htmlspecialchars($_POST["TotalStatements"]);
>falseStatements=htmlspecialchars($_POST["FalseStatements"]);
>trueStatements=htmlspecialchars($_POST["TrueStatements"]);
>unverifiedStatements=htmlspecialchars($_POST["UnverifiedStatements"]);
$searchterm=htmlspecialchars($_POST["searchterm"]);
?>

<head>
<style>
.box{
    position:relative;
    text-align:center;
    height: auto;
    border-radius:15px;
    padding:10px20px65px;
    background-color:#fcfcfc;
    -webkit-transition: all1000mtease;
    -moz-transition: all100mtease;
    -ms-transition: all1000mtease;
    -o-transition: all1000mtease;
    transition: all1000mtease;
    box-shadow:0px0px30pxrgba(0,0,0,0.15);
}
#backButton {
    border-radius: 4px;
    padding: 8px;
    border: none;
    font-size: 16px;
    background-color: #2eacd1;
    color: white;
    position: absolute;
    top: 10px;
    right: 10px;
    cursor: pointer;
}
.invisible {
    display: none;
}
</style>
<script>
window.onload=function () {

var totalVisitors=<?php echo $TotalStatements; ?>;
var visitorsData= {

```

```

"New vs Returning Visitors": [{
  cursor: "pointer",
  explodeOnClick: false,
  innerRadius: "75%",
  legendMarkerType: "square",
  name: "",
  radius: "100%",
  showInLegend: true,
  startAngle: 90,
  type: "doughnut",
  dataPoints: [
    { y: <?php echo $UnverifiedStatements; ?>, name: "Unverified", color:
"#E7823A" },
    { y: <?php echo $FalseStatements; ?>, name: "Non fake", color: "#546BC1"
},
    { y: <?php echo $TrueStatements ; ?>, name: "Fake", color:
"#b22222" }
  ]
}],
"Unverified": [{
  color: "#E7823A",
  name: "Unverified",
  type: "column",
  dataPoints: [
    { x: newDate("1 Jan 2015"), y: 65 }
  ]
}],
"Fake": [{
  color: "#b22222",
  name: "False",
  type: "column",
  dataPoints: [
    { x: newDate("1 Jan 2015"), y: 33000 },
    { x: newDate("1 Feb 2015"), y: 35960 },
    { x: newDate("1 Mar 2015"), y: 42160 },
    { x: newDate("1 Apr 2015"), y: 42240 },
    { x: newDate("1 May 2015"), y: 43200 },
    { x: newDate("1 Jun 2015"), y: 40600 },
    { x: newDate("1 Jul 2015"), y: 42560 },
    { x: newDate("1 Aug 2015"), y: 44280 },
    { x: newDate("1 Sep 2015"), y: 44800 },
    { x: newDate("1 Oct 2015"), y: 48720 },
    { x: newDate("1 Nov 2015"), y: 50840 },
    { x: newDate("1 Dec 2015"), y: 51600 }
  ]
}],
"Non fake": [{

```

```

    color: "#546BC1",
    name: "True",
    type: "column",
    dataPoints: [
      { x: newDate("1 Jan 2015"), y: 22000 },
      { x: newDate("1 Feb 2015"), y: 26040 },
      { x: newDate("1 Mar 2015"), y: 25840 },
      { x: newDate("1 Apr 2015"), y: 23760 },
      { x: newDate("1 May 2015"), y: 28800 },
      { x: newDate("1 Jun 2015"), y: 29400 },
      { x: newDate("1 Jul 2015"), y: 33440 },
      { x: newDate("1 Aug 2015"), y: 37720 },
      { x: newDate("1 Sep 2015"), y: 35200 },
      { x: newDate("1 Oct 2015"), y: 35280 },
      { x: newDate("1 Nov 2015"), y: 31160 },
      { x: newDate("1 Dec 2015"), y: 34400 }
    ]
  }
}
];

var newVSReturningVisitorsOptions = {
  animationEnabled: true,
  theme: "light2",
  title: {
    text: ""
  },
  subtitles: [{
    text: "Statistics",
    backgroundColor: "#2eacd1",
    fontSize: 16,
    fontColor: "white",
    padding: 5
  }],
  legend: {
    fontFamily: "calibri",
    fontSize: 14,
    itemTextFormatter: function (e) {
      return e.dataPoint.name + ": " + Math.round(e.dataPoint.y/totalVisitors*100)
      + "%";
    }
  },
  data: []
};

var visitorsDrilldownChartOptions = {
  animationEnabled: true,
  theme: "light2",
  axisX: {

```

```

    labelFontColor: "#717171",
    lineColor: "#a2a2a2",
    tickColor: "#a2a2a2"
  },
  axisY: {
    gridThickness: 0,
    includeZero: false,
    labelFontColor: "#717171",
    lineColor: "#a2a2a2",
    tickColor: "#a2a2a2",
    lineThickness: 1
  },
  data: []
};

var chart = new CanvasJS.Chart("chartContainer", new VSReturningVisitorsOptions);
chart.options.data=visitorsData["New vs Returning Visitors"];
chart.render();

$("#backButton").click(function() {
  $(this).toggleClass("invisible");
  chart = new CanvasJS.Chart("chartContainer", new VSReturningVisitorsOptions);
  chart.options.data=visitorsData["New vs Returning Visitors"];
  chart.render();
});

}
</script>
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

<!-- jQuery library -->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js">
</script>

<!-- Latest compiled JavaScript -->
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
<title>Search with a keyword...</title>
<script src="https://canvasjs.com/assets/script/jquery-1.11.1.min.js"></script>
<script src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>
</head>
<body>

<br><br>
  <div class="row">
    <div class="col-md-3"></div>

```



```
<divclass="col-md-6">
  <divclass="alert alert-success"><strong>Searched
Term:</strong><?php echo $searchterm; ?></div>

  <br><br>
  <divclass="box">
  <divid="chartContainer"style="height: 370px; width: 100%;"></div>
  <buttonclass="btn invisible"id="backButton">< Back</button>

  </div>
  </div>

</div>
</body>
```