MULTISTAGE FEATURE-ASSISTED DEEP LEARNING AND ITS APPLICATION IN FINE-GRAINED FAKE NEWS DETECTION

by

FUAD MIRE HASSAN

A thesis submitted to
University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

School of Computer Science
College of Engineering and Physical Sciences
University of Birmingham
June 2021

UNIVERSITY OF
BIRMINGHAM

University of Birmingham Research Archive

e-theses repository

This unpublished thesis/dissertation is copyright of the author and/or third parties. The intellectual property rights of the author or third parties in respect of this work are as defined by The Copyright Designs and Patents Act 1988 or as modified by any successor legislation.

Any use made of information contained in this thesis/dissertation must be in accordance with that legislation and must be properly acknowledged. Further distribution or reproduction in any format is prohibited without the permission of the copyright holder.

Abstract

The rapid increase of real-time news posted in social media has led to the emergence of fake news. Assessing the veracity of news claims requires an enormous labour of human fact-checkers and therefore automating the sub-tasks of fake news detection pipeline could help them identify false claims. Although the related literature addresses fake news detection tasks in a simple binary or a multiclass classification setting, challenges still remain. For instance, (1) this domain suffers from a lack of large scale datasets and a large proportion of the instances belongs to legitimate news which creates a class-imbalance problem, (2) the characteristics of fake news are not yet known in order to generate effective discriminative features (3) and the content of multiclass categories can be very similar which makes it hard for multiclass classifiers to capture the finer distinctions between them.

The major focus of this thesis is to investigate novel models in Natural Language Processing (NLP) and Machine Learning (ML) that can help classify the veracity of a claim with respect to textual evidence into multiclass categories. Our first contribution is related to boosting the performance of multiclass stance detection. We show that using a feature-assisted neural model, aided with augmented training samples to deal with data imbalance, provides state-of-the-art performance on the FNC-1 dataset. The second contribution explores a way to improve stance detection, especially the minority categories, by proposing multistage classification approaches. We show a significant performance increase by breaking down the multiclass categories into different sub-stage feature-based and feature-assisted neural classifiers with category-specific features. Inspired by the multistage classification approaches, the final contribution proposes five-stage and three stage feature-assisted neural classifiers into multiclass fake statement detection. We conclude that sub-dividing the fine-grained task into multiple feature-specific classifier provides state-of-the-art performance.

Acknowledgements

In the name of Allah, the Most Gracious and the Most Merciful

I thank Allah for granting me the opportunity to come to University of Birmingham and pursue my PhD in the School of Computer Science which has such an amazing and supportive research community, and I have been lucky enough to learn from colleagues who are eager to help newcomers.

My sincere gratitude goes to my supervisor Dr Mark Lee, whom without his unwavering support and guidance non of the work in this thesis would have been possible. I thank him once again for being such a great supervisor and for pushing me beyond what I thought were my limits. I don't see Mark only as a supervisor who've introduced me the field of Natural Language Processing (NLP), I see him as a mentor and a friend whose motivation and encouragement will always guide me throughout my professional career.

My appreciation also goes to the committee members of my thesis group, Dr Peter Hancox and Dr Rowanne Fleck for their additional guidance throughout my PhD process.

I could not forget to express my thanks to my colleagues within the School, Irfan Muhammad, Abdullah Alharbi, Phan Trung Hai, Akram Alofi, Hayatullahi Adeyemo, Abdulla Aldoseri and Hassan Labani for their insightful disccusions and suggestions.

This doctoral project would not have been possible without the financial support from the Islamic Development Bank – IDB. Again, and I cannot stress this enough, I thank for their generous support which reduced my financial burden and allowed me to focus on my research. Similarly, I am very grateful for the support and patience I have received from SIMAD UNIVERSITY during my study leave.

Last but not the least, I would like to take this moment to express my gratitude to my family for their help and encouragement to complete this work. I would like to especially thank my wife (Hamdi) and children (Mas'ud, Mohamed and Manal). Without the love and support of my beloved wife, I could not have accomplished this success.

Publications

1. **Fuad Mire Hassan** and Mark Lee *Imbalanced Stance Detection by Combining Neural and External Features.* Proceedings of the 7th International Conference on Statistical Language and Speech Processing (SLSP 2019), Springer, Ljubljana, Slovenia. (Hassan and Lee, 2019)

2. **Fuad Mire Hassan** and Mark Lee *Multi-stage News-Stance Classification Based on Lexical and Neural Features.* Proceedings of the 13th International Conference on Computational Intelligence in Security for Information Systems (CISIS 2020) - Special Session: Fake News Detection and Prevention, Springer, Burgos, Spain (Hassan and Lee, 2020a)

3. **Fuad Mire Hassan** and Mark Lee *Political Fake Statement Detection via Multistage Feature-assisted Neural Modeling.* Proceedings of the 18th Annual IEEE International Conference on Intelligence and Security Informatics (ISI IEEE 2020), IEEE, Arlington, VA, USA (Hassan and Lee, 2020b)

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

After the rapid development of the World Wide Web, the internet has become one of the most used channels for people to communicate with each other and distribute information throughout the entire world. Traditional media are being challenged by so many other internet communication channels, such as blogs, social networking sites (Facebook, Twitter, etc,.), forums, etc., with the aim to facilitate the dissemination of real-time news among social media users (Shu et al., 2017). The easy accessibility for the users of social media strengthens the consumption of large amounts of information and it is possible for a post to reach hundreds of millions of users. In the mid of 2017, a survey study discovered that 67% of American adults get their news from social media[1]. This has led to the emergence of fake news which can no longer be fact-checked in real-time as traditional media do before a story gets published. Therefore, fake news detection has become one of the important research directions in Natural Language Processing (NLP). This chapter lays the ground work in understanding the motivation of this study, problem formulation, research questions and finally, our thesis contributions.

[1]http://www.journalism.org/2017/09/07/news-use-across-social-media-platforms-2017/

1.1 Motivation

In the post truth era, the potential growth of online news communities through social media has enabled the proliferation of fake news for either political or financial related purposes. Therefore, fake news has become more popular and widely disseminated through social networks compared to traditional media (Shu et al., 2017). Sometimes, news events are created for a viral confusion to affect people's opinions about a political event for instance, fake news may have influenced the outcome of 2016 United States presidential election (Gunther et al., 2018). In terms of monetary perspective, fake news can also be a potential threat to severely damage the economy by creating misleading information about a crisis that can cause panic in the market which leads to an adverse effect on the stocks[2]. Given the massive influence of fake news, it was named the word of the year in 2016 by the Collins Dictionary[3].

Online misinformation or fake news is one of the top 10 challenges that the world faces today as reported by World Economic Forum[4]. The verification of news claims remains a challenging problem in computational journalism in this age of a technology-driven world. As such, manual fact-checking websites such as *Politifact*[5], *Snopes*[6] and *factcheck*[7] investigate and judge potential false information but they are facing difficulties to fact-check massive amounts of rumours, hoaxes and claims by using human experts. The manual process is currently time consuming to reach a verdict with evidence as it may take from a few hours to a few days considering the complexity of the claim (Hassan et al., 2015a). By the time false claims are determined through a manual process, they have been widely shared, commented and caused political or social harm as intended by the perpetrators. In order to alleviate the complexity and time-consuming manual process, fake news detection (e.g. automated fact-checking) has become one of the most active

[2]https://www.cnbc.com/id/100646197
[3]http://www.newsweek.com/fake-news-word-year-collins-dictionary-699740/
[4]http://reports.weforum.org/outlook-14/
[5]http://www.politifact.com/
[6]https://www.snopes.com/
[7]https://www.factcheck.org/

research areas in NLP.

As manual fact-checking is a complex task, automating any part of the process could help human fact-checkers speed up their judgement about a claim (Konstantinovskiy et al., 2018). However, determining the veracity of a news claim is not as easy as it sounds for either human fact-checkers (Conforti et al., 2018) or for automated systems (Borges et al., 2019). Considering the negative effect of fake news on political and social issues, urgently required initiatives are underway for automating the sub-problems of fact-checking by Fullfact[8], factmata[9], ClaimBuster[10], Google News Initiative[11], etc,.

Despite considerable efforts on determining whether a claim, rumour, clickbait-headlines or news article is misleading or not, most existing work has been interchangeably referred to by domain dependent names such as fake news classification (fake political statement detection) (Wang, 2017; Pérez-Rosas et al., 2017; Potthast et al., 2017b; Rashkin et al., 2017), automated fact-checking (e.g. stance detection) (Popat et al., 2017; Karadzhov et al., 2017; Nadeem et al., 2019), rumour detection (Castillo et al., 2011; Zubiaga et al., 2016a; Derczynski et al., 2017) and clickbait detection (Potthast et al., 2016; Chakraborty et al., 2016). Generally speaking, a complete fake news detection pipeline should extract key features from the news contents and understand what other factual sources are saying about the topic to distinguish fake from legitimate news. However, the existing approaches rely on either news-content or social-context (Shu et al., 2017) with the underlying premise that there could be discriminative features (e.g. lexical or neural features) extracted from the claims and metadata information (e.g. speaker profile, user information and social interactions) (Wang, 2017), claims and articles (Pomerleau and Rao, 2017), writing styles of news (Horne and Adali, 2017; Rashkin et al., 2017) or rumours (Zubiaga et al., 2016a; Derczynski et al., 2017) in conjunction with multiclass classifiers to assess the veracity of fine-grained news categories. In this thesis, we focus on stance detection (Pomerleau and Rao, 2017) and political fake statement detection (Wang,

[8]https://fullfact.org/automated
[9]https://factmata.com/
[10]https://idir.uta.edu/claimbuster/
[11]https://newsinitiative.withgoogle.com/intl/en_gb/

2017) to build novel multistage feature-assisted neural classifiers to better determine the veracity of a claim with respect to textual evidence from fine-grained categories.

1.2 Challenges

In this section, we present the three main challenges addressed in this dissertation with respect to the application of Machine Learning (ML) models for fine-grained fake news detection.

1.2.1 Class Imbalance

In the context of NLP, many classification problems such as spam detection (Jin et al., 2015; Liu et al., 2017) and hate speech detection (Zhang and Luo, 2019; Rizos et al., 2019) suffer a class imbalance problem where the distribution of data is highly unbalanced. This common issue arises when one or some of the categories have a significantly higher number of instances in the dataset compared to other categories and this is also a common problem in computer vision (Kubat et al., 1998; Xiao et al., 2010). ML models trained with imbalanced datasets can be easily influenced to introduce a bias towards the majority categories. This may produce poor to mediocre performance where the predictions of underrepresented categories are considered important to the task. Models can achieve higher accuracy on unbalanced datasets since it is not difficult to predict more on majority classes; and because of that it is recommended to use the Macro-F1 metric to evaluate the performance of the task taking into account the unweighted mean for each category.

Most real-world datasets have a class imbalance problem and fake news detection research area is not an exception (Hanselowski et al., 2018; Kochkina et al., 2018). There are very few fake news samples since the vast majority of the news is legitimately published in social media by a reliable sources. In such a scenario, it is often very difficult to train a classifier with an imbalanced dataset and expect to obtain satisfactory predictions on category-wise and improve the overall performance. The major challenge is, how do we

improve the predictions of minority categories and the overall performance of the task. The most common approaches to deal with data imbalance problems are data sampling methods (Buda et al., 2018) and ensemble learning (Jin et al., 2015; Liu et al., 2017). To address the data imbalance problem for minority categories, we firstly summarize the news documents by using extractive text summarization and we then use a synonymous replacement method to create augmented data based on the original labeled data.

1.2.2 Text Representation

Feature-based ML relies on heavy feature engineering and using different supervised learning systems to build models with relatively interpretable results. These systems assume that the important cues or signals in the text can be extracted by using NLP techniques and those cues can be generated as features in order to train an ML model. To build feature-based models requires a huge amount of human labour, domain knowledge and language-specific knowledge and even then they may not adequately represent the contextual information of long text (Sharma et al., 2019). With respect to fake news detection, previous studies (Pérez-Rosas et al., 2017; Aker et al., 2017; Gencheva et al., 2017; Rashkin et al., 2017) proposed ML models in combination with linguistic features such as lexical semantic classes, n-grams, TF-IDF, sentiment markers, entities, word embeddings, topics, discourse, etc,. However, we do not have enough information about the characteristics of misinformation yet; (Shu et al., 2017) and because of that we cannot generate effective manual features from news articles without taking the context into account. Therefore, a feature-based model alone is not sufficient to combat against misinformation since we do not have a comprehensive grasp about the misinformation writing styles, their different topics and insights into the nature of fake news in social media (Ruchansky et al., 2017).

Instead of extracting manual features, Deep Learning (DL) models especially Gated Recurrent Unit (GRU) (Cho et al., 2014) were proposed to represent the sequence of textual information taking the context into account by using word embeddings (Mikolov et al., 2013; Pennington et al., 2014). Although deep neural models can automatically

learn the patterns and the semantic relations from the text, they can only effectively work when we have large scale training dataset for that specific domain. One of the issues for fake news detection is the lack of an available large labeled dataset (Torabi Asr and Taboada, 2019) and this directly effects the ability of the neural models to automatically learn from text in order to achieve good performance. In this thesis, we are taking advantage of the services of feature-assisted neural models to find the balance between a rich set of hand-crafted features and DL models.

1.2.3 Multiclass classification

Existing work on text classification models with multiple classes present different challenges than models with binary classes. Unlike binary classifiers, multiclass classification problems require a model that can capture the finer distinctions among the fine-grained categories. Although multiclass categories of a text classification task can be very similar in terms of their contents, they can also be correctly classified by their discriminative semantic information. Some of the text classification approaches tried to explore more discriminative feature representations to build multiclass classification systems (Davidson et al., 2017; Long et al., 2017; Alhindi et al., 2018). However, multiclass classifiers cannot pay attention to category-specific features in order to separate text of a category from the rest.

In addition to that, the unbalanced distribution of the dataset complicates the training of multiclass classification models (Zubiaga et al., 2016b; Rizos et al., 2019; Hanselowski et al., 2018) and therefore demands countermeasures in order for the models to better classify between the fine-grained classes. Some of the previous studies on fake news detection have pursued multiclass classifiers to distinguish between fine-grained categories and they are very poor in predicting samples of underrepresented categories or separating samples that have similar textual contents (Wang, 2017; Pomerleau and Rao, 2017; Horne and Adali, 2017; Rashkin et al., 2017; Zubiaga et al., 2016a; Derczynski et al., 2017). This motivates to go beyond multiclass classification and explore multistage

classification approach with the aim to build a sub-stage classifier that pays attention to category-specific features that are more effective in improving the category-wise predictions and the overall performance.

1.3 Problem Definition and Research Questions

In this thesis, the main objective is to explore novel NLP and ML models, particularly models that can identify the fine-grained veracity categories of a claim in relation to a textual evidence. We adopt the definition of *"Fake News"* proposed by (Shu et al., 2017) as a news story that is intentionally and verifiably false. This adopted definition considers two key elements which are authenticity and intent. First, the inaccurate story (claim) can be verified through available means such as potential sources which either support or refute the claim. In line with this, the **Stance Detection** problem will be the focus of the research presented in Chapters 3 and 4. Second, the false information is propagated with bad intention and the perpetrators usually mix true and false claims to mislead readers. Another problem that this research explores in Chapter 5 is **Political Fake Statement Detection**. This section presents the description of the research problems and the formulation of related research questions.

1.3.1 Multiclass Stance Detection

The focus of this task is to detect the stance of a headline by predicting its class as *agree*, *disagree*, *discuss* or *unrelated* in relation to an article body. Table 1.1 demonstrates illustrative snippets about the claim and document pairs with their labels from the FNC-1 dataset. Existing models approach this problem using complex DL models such as deep Recurrent Neural Networks (RNNs) (Hanselowski et al., 2018; Borges et al., 2019), deep Convolutional Neural Network (CNNs) (Baird et al., 2017; Xu et al., 2019) and Memory Networks (Mohtarami et al., 2018). The increase in complexity of these models is ineffective when predicting instances from the minority classes. It may also produce

Table 1.1: An illustrative example from the FNC-1 dataset

***Claim**: No, a spider (probably) didn't crawl through a man's body for several days*	
Stance	**Body**
Agree	[...] a lot of arachnologists saying simply, "No, that didn't happen." Thomas may have been told that a spider invaded his flesh, but the arachnologists we spoke to about this story sincerely doubt that one actually did. "I think this is extremely suspect, unusual, and likely not possible," Christopher Buddle, [...]
Disagree	[...] Dylan Thomas, 21, was on holiday with friends when he woke up one morning to find a red, scar-like trail on his stomach. He was initially given antihistamine cream to treat what doctors in Bali thought was an insect bite. But the red trail spread upwards and by Monday it had started blistering so he was sent to see a dermatologist, who discovered the small tropical spider and removed it. [...]
Discuss	[...] Twenty-one-year-old Dylan Thomas says he was on holiday in Bali last weekend when he awoke to find a mysterious red trail on his chest. He says doctors initially thought it was an insect bite but later discovered a spider inside his stomach. Thomas's story has not been verified by the Guardian [...]
Unrelated	[...] A female passenger dressed in a hazmat suit — complete with a full body gown, mask and gloves — was spotted Wednesday waiting for a flight at the airport. Another traveler snapped a photo of the woman and provided it to The Daily Caller. Thomas Eric Duncan, the first person to be diagnosed of Ebola on American soil, [...]

mediocre overall performance due to overfitting on smaller datasets. The limited size of data for agree and disagree classes (class imbalance) can have an adverse impact on the overall unweighted performance for the task. Several approaches attempted to alleviate the data imbalance problem by using undersampling and oversampling techniques but that did not also work well to solve the class imbalance.

- **Research Question 1:** To what extent can the use of lexical and similarity feature representations influence the outcome of feature-based stance classification? In addition, can a neural model be improved through the use of regularization techniques and lexical-overlap features for stance detection? And what is the effect of text augmentation for minority classes on this task?

By addressing these questions on document-level stance detection problem over the FNC-1 dataset, Chapter 3 explores three avenues (a) the assessment of important lexical and similarity features and their predictive power with respect to feature-based ML models; (b) since the underrepresented classes don't have enough data to compete with other classes,

text augmentation (text summarization and synonym replacement methods) techniques are applied to create more samples with the aim to improve their accuracy (c) and finally, a feature-assisted DL model (optimized its performance by using regularization layers) is proposed to detect the multiclass stances.

1.3.2 Multistage Stance Classification

Given a claim (or headline) and news article pairs, the stance classifier should determine whether the pairs are *related* or *unrelated*. If the pairs are related, the stance detector should further classify these predictions as *agree*, *disagree* or *discuss*. The first part of the problem is to assess the relatedness of the pairs so that proposed models filter away those very often *unrelated* instances as they generate class imbalance problem; The second part of the problem is how to detect the stance which is the most difficult to distinguish among the remaining stances: *agree*, *disagree* or *discuss*.

However, as most popular NLP tasks, stance detection also suffers a class imbalance problem and the state-of-the-art multiclass classification systems (Bhatt et al., 2018; Borges et al., 2019; Hanselowski et al., 2018; Saikh et al., 2019) have shown to be easily influenced in predicting the majority classes. The major problem is, how do we improve the predictions of related categories, especially - agree and disagree, which can reveal the veracity of the claims as either true or false.

- **Research Question 2:** Given the success of feature-based ML and feature-assisted neural classifiers, to what extent does a multistage classification affect the class-wise and overall performance of stance detection?

We tackle this problem by proposing two multistage classification approaches to overcome the data imbalance problem as we distribute the training of multiclass categories into several different stages in order to improve the accuracy of underrepresented classes (e.g. *agree* and *disagree*) and the overall performance.

Table 1.2: An illustrative example from the LIAR-PLUS dataset

Label	**Statement**	**Justification**
Barely-true	Jim Dunnam has not lived in the district he represents for years now.	But determining that would take significant detective work, far more than a few photos. A broader interpretation would allow for the possibility Dunnam hasn't lived exclusively in his House district for years, but instead flits between – and lives in – both houses.
Mostly-true	Says GOP primary opponents Glenn Grothman and Joe Leibham cast a compromise vote that cost $788 million in higher electricity costs.	Considering that the $532 million figure covers just 5 years, it's reasonable to assume the 15-year total is higher. Our rating Stroebel's ad says Glenn Grothman and Joe Leibham cast a "compromise vote that cost $788 million" in higher electricity costs.They cast that vote, and higher costs have followed.
False	I dont know who (Jonathan Gruber) is.	Pelosi said, "I don't know who (Jonathan Gruber) is." Video showing Pelosi citing Gruber's work offers the clearest evidence that she did indeed know who he was, and even her office now acknowledges that she meant to say that she didn't know Gruber personally.

1.3.3 Multistage Fake Political Statement Detection

In this work, we address the problem of Political Fake Statement Detection. We are given a set of claims, relevant justifications and their metadata where each entry is associated with a fine-grained class label (as shown in Table 1.2) corresponding to one of these classes: *pants-on-fire*, *false*, *barely-true*, *half-true*, *mostly- true* and *true*. Based on the literature (Long et al., 2017; Liu et al., 2019; Wang et al., 2019b), the existing DL approaches have significantly improved the performance of Political Fake Statement Detection by modeling statement with the speaker's credit history.

However, the credit history may not be available in reality and most approaches did not consider about the evidence that supporting or denying claims when detecting fake news. In addition, state-of-the-art models may struggle to detect fine-grained labels using multiclass deep neural classification models because the statement of the speaker expresses factual and incorrect instances at the same time.

- **Research Question 3:** Considering statement and justification pairs as a stance

detection task, can neural models benefit from the inclusion of lexical features in a multistage classification hierarchy for this task?

Recognizing that the metadata may not be available, we pursue another approach that does not require credibility-history and instead considers pairs of claims and justifications as an input in a stance detection manner. Based on the observation that claims could belong to classes that are partially true and false at the same time, we realize that multiclass classifiers would struggle to predict the fine-grained classes; therefore, in Chapter 5, we approach this problem as a multistage learning process.

1.4 Contributions

Our aim in this thesis is to explore the application of feature-assisted DL to fake news detection in a multiclass or multistage classification setting. We first approach the fake-news stance detection problem (e.g. FNC-1 dataset) as a multiclass classification using feature-based ML and feature-assisted neural models. Aiding augmented training instances to overcome the data imbalance problem and adding Batch-normalization and Gaussian noise layers enable the feature-assisted model to prevent overfitting and improve class-wise and overall accuracy. In addition, we evaluate the proposed model on the Argument Reasoning Comprehension (ARC) dataset to assess the generalizability of the model. The experimental results of our models outperform the current state-of-the-art (see Chapter 3).

We then describe an improved approach to the fake-news stance detection using multistage feature-assisted DL approaches. We break down the multiclass classification problem into two-stage and three-stage classifiers by proposing feature-based classifier for the first-stage and feature-assisted neural models for the other stages in an effort to mitigate the class imbalance problem. The experimental results demonstrate that the proposed models improve upon the state-of-the-art Accuracy and F1 score for stance detection. We also show experimentally that our models achieve solid results on minority

classes i.e. agree and disagree without using fine-tuning approach or adding more training samples (see Chapter 4).

And finally, we approach the Political Fake Statement Detection problem by proposing two multistage feature-assisted neural models that consider claims and justifications as an input in a stance detection manner. We explore five-stage and three-stage classification strategies to better discern between the fine-grained labels of fake news. The proposed model in each stage is built on the powerful combination between dual GRU layers and lexical features which we further optimise by using Gaussian Noise. An extensive experimental work on a real-world benchmark LIAR-PLUS (an extended version of LIAR) dataset shows that three-stage model achieves state-of-the-art Accuracy and F1-score without using metadata of the speaker. We also experimentally demonstrate that modeling the credit history in conjunction with statement and justification gives more than 6% improvement (see Chapter 5).

1.5 Thesis Organization

This section presents the successive chapter contents of this thesis.

In Chapter 2, several research areas of fake news detection have been presented. We start by exploring the terminologies related to fake news, main tasks and sub-tasks of fake news detection & fact checking, and we continue by reviewing related works on clickbait detection, rumour detection, truth discovery, stance detection as well as political fake statement detection. Finally, as classical methods and deep neural models are used in our research, we also summarize the details of those different NLP techniques, data augmentation and multistage text classification approaches.

Chapter 3 describes feature-based classifiers, text augmentation and a feature-assisted neural model. It also presents the experimental setup, baselines and empirical results using Accuracy metric for the stance detection task - FNC-1 dataset. The chapter concludes the empirical results of cross-domain validation via FNC-1 and ARC datasets using F1-score.

Chapter 4 presents the details of two-stage classification, three-stage classification, a feature-based model for the first stage and feature-assisted models for other stages. It also describes multistage experimental procedure, state-of-the-art baselines and comparative results using both Accuracy and F1-score metrics for stance detection - FNC-1 dataset. Finally, the chapter wraps-up with a discussion on error analysis.

Chapter 5 provides the discussion on five-stage classification, three-stage classification and a feature-assisted DL model proposed for every stage in both of multistage settings. It also presents model settings, baselines, experiments and the evaluation results on LIAR dataset. Lastly, the chapter concludes the description of errors analysis and conclusion remarks.

The last chapter draws the contributions from the current study and discusses areas of future work directions.

CHAPTER 2

RELATED WORK

This chapter undertakes the recent literature into the research area of fake news detection. There are different kinds of fake news detection tasks and methods therefore, this chapter will present an in-depth analysis with respect to the previous studies related to this work. Section 2.1 provides an overview of the topic while Section 2.2 goes deep into the different tasks of fake news detection. The next two Sections (2.3 and 2.4) present a discussion about text classification and data augmentation methods. The next three Sections (2.5, 2.6 and 2.7) present the discussion of the related work towards our three main contributions. The last Section (2.8) concludes the chapter with a discussion of traditional and DL methods.

2.1 Fake News Detection: An Overview

In recent years, fake news on online platforms has been a serious problem which hinders the reliability of news being shared on social media. To tackle this problem, fake news detection has become an important research area in NLP. Therefore, fake news detection is defined as the task of categorizing news along a continuum of veracity, with

an associated measure of certainty (Conroy et al., 2015). The following surveys (Shu et al., 2017; Sharma et al., 2019; Pierri and Ceri, 2019; Zhang and Ghorbani, 2020) offer a comprehensive discussion on recent advances in terms of characterization, terminology, detection and mitigation of fake news in social media. The surveys by (Saquete et al., 2020; Oshikawa et al., 2020) present a systematic review of different fake news sub-tasks, dataset resources, competitions, models and performances using NLP solutions as well as comparative studies. An alternative way of dealing fake news that can help detect misinformation is through automated fact-checking and these surveys covered on this topic (Thorne and Vlachos, 2018; Saquete et al., 2020). Lastly, these recent surveys focused on rumour detection (Zubiaga et al., 2018a; Li et al., 2019a), clickbait detection (Saquete et al., 2020) and truth discovery (Li et al., 2015). In order to explain the purpose of fake news detection and different NLP approaches employed within the scope of detecting the misinformative language on news text, we first need to understand the definition of fake news and the related terms. However, various surveys in this domain have proposed different definitions regarding fake news but we adopt ours from Shu et al. (2017). Fake News - is defined as a news story that is intentionally and verifiably false. However, fake news stories can come in various forms and they are referred to different keywords with close definitions in different studies. The most common keywords used in this domain are fake news and rumours, but there exist several other related terms. Some of those related terms which are sometimes referred to as fake news are: misinformation, disinformation, hoax, propaganda, satire, rumour and clickbaits. But researchers have long looked at the differences between these terms with the aim to clearly understand their authenticity and the intention of the people spreading the false news.

- **Misinformation** - is defined as an inaccurate story that is being disseminated unintentionally through social media due to a genuine mistake or incorrect facts (Pierri and Ceri, 2019).

- **Disinformation** - is a news story that is spread intentionally with the aim to deliberately mislead a target population (Pierri and Ceri, 2019).

- **Hoax** - is a fiction or paranoia fueled news content that is used to intentionally deceive readers (Rashkin et al., 2017).
- **Propaganda** - is a type of fake news that convinces readers to believe omitted or one-sided news contents which are disseminated to intentionally influence their emotions for political or social agenda (Pierri and Ceri, 2019).
- **Rumour** - is a post circulating through social media of which the veracity is yet to be verified (Zubiaga et al., 2018a).
- **Satire** - is a humorous news story that is created to entertain readers and is not meant to mislead for people to interpret as true news (Rashkin et al., 2017).
- **Clickbaits** - is a news headline that causes visitors to click a catchy or exaggerated headline with the aim to mislead their expectations (Chen et al., 2015).

2.2 Fake News Detection Tasks

Fake news detection is a multidisciplinary research field, which studies **Fake News Classification**, **Fact-checking**, **Clickbait Detection**, **Rumour Detection**, **Truth Discovery** and spans across different areas such as journalism, language and computer science. One way to tackle fake news is to analyse the cues of the textual contents (**Content-based Detection**) with respect to news articles, short statement, clickbait headlines or rumours in social media and classify them into predefined categories using ML models. Another way is to use different kinds of data from multiple sources such as claims, news articles, speaker profiles, reliability of the sources, etc., in order to build a fact-checking system by modeling through the combination some of these sources (**Knowledge-based Detection**). Currently a large body of fake news detection literature exists for different domains but we surveyed some of the relevant topics to get the background knowledge in the area. Prior approaches of fake news detection can be grouped into the following five main categories and their sub-categories as shown in Figure 2.1.

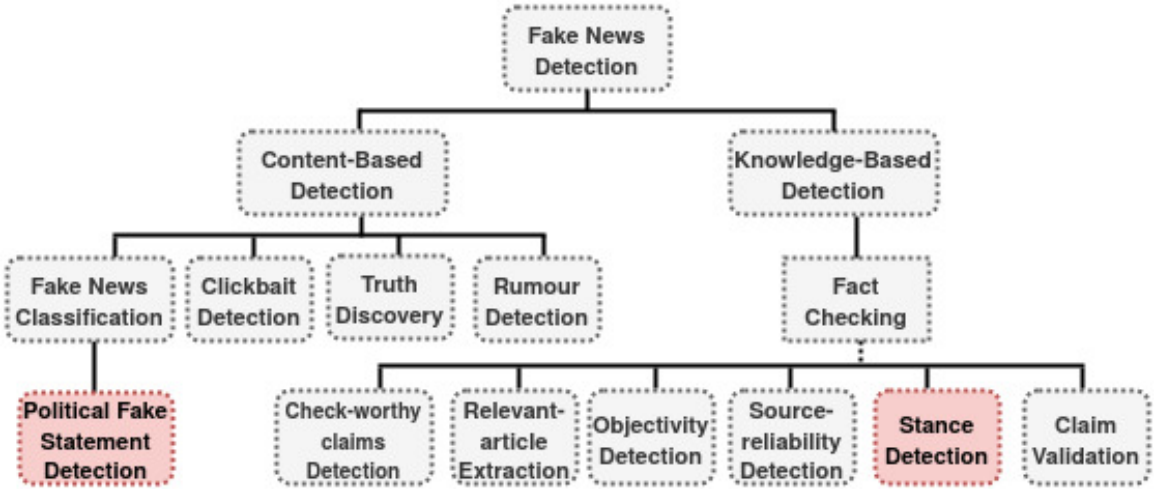

Figure 2.1: Fake news detection tasks

2.2.1 Fake News Classification

Fake news perpetrators employ deceptive writing style to persuade or appeal to a wide range of social media users in order to spread false posts. Fake news classification approaches capture the misinformative cues or signals in the textual content of the news stories. These approaches can be categorised into statement level (presented in Section 2.7) and article-level as explored in the following paragraphs.

Previously, Horne and Adali (2017) conducted textual analysis on a couple of hundred to a thousands of news documents (bodies and titles) from different categories (true, false and satire), utilizing a linear Support Vector Machine (SVM) with features like stylistic, complexity and psychological characteristics. They observed similarities between the writings of fake and satire textual contents. They also found that fake news packs the main points of its claim in the title while the body of the article tends to use short and repetitive textual content. Another study by (Pérez-Rosas et al., 2017) created a dataset for fake news detection with a binary setup as either legitimate or fake. They employed linear SVM classifier with different kinds of linguistic features including readability, punctuation, syntax, ngrams and Linguistic Inquiry and Word Count (LIWC) (Pennebaker et al., 2001). The model achieved higher accuracy that is comparable to human ability to identify fake news.

Moreover, Potthast et al. (2017b) also focused to differentiate the categories of news (satire, false, hyperpartisan and mainstream) by assessing their writing style using a meta-learning approach (unmasking) which is originally developed for author verification. In follow-up work, Kiesel et al. (2019) organized the first Hyperpartisan Detection shared task[1] at the SemEval to promote the development of ML systems that can detect media bias within the writing of the news stories as right, left, or main-stream. The organizers released a large dataset consisting of 754K news stories and the challenge received attention from the NLP wide community with 42 system submissions.

News reliability detection has been addressed as a task in (Rashkin et al., 2017) and the authors created a dataset with textual claims (statement-level) of fine-grained categories from Politifact and collected another dataset (article-level) from websites in four different reliability categories: trusted news, hoax, satire and propaganda. They also conducted deeper textual comparison on the language use of trusted news and that of different categories of fake news such as hoax, satire and propaganda by using LIWC. Finally, they trained Max-Entropy with L2 regularization and LSTM classifiers to predict the varying levels of truth (fine-grained categories) as well as the reliability categories. In follow-up work, propaganda detection (Da San Martino et al., 2019b) has been specifically explored at the NLP4IF shared task[2] with the aim to identify sentence-level propaganda or non-propaganda as well as phrase-level linguistic propaganda techniques designed as a fine-grained categories. The phrase-level sub-task is about extracting fragments or spans that contain 18 possible propagandistic techniques (e.g. name-calling, whataboutism, repetition, red-herring, straw-man, etc.). This challenge received more than 30 participating teams and they submitted their predictions for more than 3000 submissions (Da San Martino et al., 2019a).

However, all these fake news classification methods rely heavily on smaller domain-specific datasets and do not utilize the evidence reported by various sources with respect to news claims, thereby limiting their ability to identify fake news by stylistic analysis

[1]https://pan.webis.de/semeval19/semeval19-web/
[2]https://propaganda.qcri.org/nlp4if-shared-task/

alone. Due to the lack of large labelled datasets, the majority of these models provide mediocre performance as they mostly employ stand-alone feature-based or neural-based models that are not well suited for this domain as explained in Chapter 1 especially, Section 1.2. In this thesis, we present multiclass and multistage classification approaches for fake news detection by determining the stance of news claims in relation to their evidence.

2.2.2 Clickbait Detection

Clickbait is a type of fake news that is designed to provoke the user's curiosity into clicking the web link of an attractive headline which often results in misleading or missing article content. The purpose of the clickbait generators is to expand their reach by spreading clickbaits in social media and subsequently make revenues through advertisements. As the adverse impacts of clickbaits increased, research on clickbait detection in NLP and ML has been explored in recent years. Earlier work on the subject, Biyani et al. (2016) defined 8 types of clickbait and investigated novel hand-crafted features using an ML classifier. Also, Potthast et al. (2018) compiled the crowdsourced dataset focusing on tweets and the same authors (Potthast et al., 2016) used an earlier version of the dataset to develop different ML classifiers with a rich set of features to detect clickbaits. Chakraborty et al. (2016) explored the differences between clickbait and non-clickbait textual contents with the aim to extract features that can be used in classifying clickbaits. They also developed a browser extension to protect the readers by providing alerts about the presence of clickbaits.

To raise awareness and accelerate AI technology to automatically detect clickbaits, Potthast et al. (2017a) organized a Clickbait Challenge in 2017. The lab challenge released a benchmark dataset to invite submissions (e.g. 13 systems submitted) of ML models capable of distinguishing clickbaits from legitimate headlines. With the emergence of DL models, Agrawal (2016) used TextCNN model in conjunction with pre-trained word vectors to learn more generalized features for clickbait detection. In addition to that, Anand

et al. (2017) introduced different DL architectures with distributed word embeddings and character-level word embeddings to minimize the reliance of heavy hand-crafted features when detecting clickbaits.

However, most clickbait detection methods only utilize headlines and they overlook the relationship between the headlines and the target news bodies, which could potentially improve the performance of clickbait detection.

2.2.3 Truth Discovery

Truth discovery is an early research area of fake news detection which resolves conflicting information among multi-source noisy data. For example, users may send queries through search engines to look for answers (truth about doubtful statement) when they want to know more about certain facts and this may result conflicting information. Therefore, the main purpose of truth discovery is to estimate the source reliability in order to determine which one is the true fact among multiple pieces of noisy information present in the web or in the knowledge bases. Hence, the challenge is to find credible answers from the conflicting results of the search engines because some of the sources contain erroneous or untruthful information (Li et al., 2015). Previous research about truth discovery tried to develop methods for verifying facts through the entire web or known structural knowledge bases.

In one of the early works that distinguishes truthful information from untruthul on the web, Li et al. (2011) described a system called "T-verifier" that takes a doubtful statement as an input, and returns a truthful statement by assessing extracted similar (alternative) statements using Yahoo search engine. Also, Samadi et al. (2016) proposed an integrated ClaimEval system that extracts a set of conflicting arguments from the web based on given claims and performs joint estimation of source credibility and claim evaluation using Probabilistic Soft Logic (PSL).

In the case of fact verification through knowledge graphs, Ciampaglia et al. (2015) explored computational truth-discovery mechanism based on assessing the veracity of

claims by discovering the shortest paths among concept nodes in a knowledge graphs. Recently, Shi and Weninger (2016) suggested a way of detecting if some claims are true or false by using link prediction in a knowledge graph.

Similarly, Nakashole and Mitchell (2014) proposed a method that computes a believability score of a generated fact candidate by using an objectivity language score of a source and a co-mention score (e.g. if similar sources mention the fact candidates then, the believability should have the same score) to reach a verdict. A similar work, Vlachos and Riedel (2015) and Thorne and Vlachos (2017) explored numerical truth-discovery systems that can identify the overlapping named entity, statistical property and year between unverified claims and a knowledge-base entries in order to deduce a verdict regarding the truthfulness of the unverified claim. However, the truth discovery methods assume that factual statements follow a specific structure (e.g. subject-predicate-object triples) therefore, they cannot handle claims of long sequences or paragraphs.

2.2.4 Rumour Detection

Social media is a platform used by many users to post or share information including news events (which could be a rumour or fake) in a way to deliver the news instantly throughout their networks. For example, a social media user starts a claim (rumour) by posting throughout his/her network, other users react to the source post by a set of replies and this creates multiple conversation branches.

Earlier work on rumour detection in microblogs built based on supervised ML. The work of (Castillo et al., 2011) attempted to develop automatic methods to determine the credibility of given real world events extracted from Twitter. A combination of linguistic and propagation features were extracted from the tweets and retweets. They then used SVM, decision trees and Bayesian networks to classify the credibility of newsworthy tweets. Moreover, Kwon et al. (2013) proposed a methodology for detecting rumours on twitter data streams (urban-legends dataset). The authors used temporal, structural and linguistic features to identify the characteristics of rumours and misinformation as

they employed random forest, decision trees and SVMs for classifying rumours and non-rumours. Likewise, Jin et al. (2016) proposed a news verification method in microblogs. The method detects conflicting viewpoints between tweets and their retweets (e.g. whether the retweets support or oppose the tweet) by constructing topic modeling as well as credibility network formulated as an optimization problem.

Recent works (Zubiaga et al., 2016a; Derczynski et al., 2017) have attempted to verify the twitter conversations using rumour verification and stance detection with the purpose to assess the truthfulness of rumours and prevent misinformation. In their setup, a rumour verification task has three veracity labels (true, false or unverified) while its sub-task, stance detection, has four multiclass categories such as supporting, denying, querying or commenting. Some of the existing methods on rumour detection and stance detection in microblogs are built based on supervised ML incorporated with a very rich set of hand-crafted features (Enayet and El-Beltagy, 2017; Bahuleyan and Vechtomova, 2017; Singh et al., 2017; Srivastava et al., 2017; Wang et al., 2017b) or DL models (Kochkina et al., 2017; Lozano et al., 2017) using the publicly available dataset from RumorEval2017 shared task.

Zubiaga et al. (2016a) have also released a second public dataset called PHEME for both of the tasks rumour and stance detection. Zubiaga et al. (2016b) and Aker et al. (2017) utilized a feature-based approaches to achieve state-of-the-art results using PHEME dataset. In addition to that, prior approaches addressed these tasks separately but Kochkina et al. (2018) and Ma et al. (2018a) attempted multi-task learning to jointly train rumour detection, stance detection and veracity classification. The purpose to unify these tasks in a multi-task learning strategy was to learn common features related to both tasks while strengthening their task specific features. The authors showed that a multi-task learning approach was the best strategy as it consistently outperformed against single-task baselines.

However, detecting rumours and false information on microblogging services is important and still a challenging task because even humans cannot reliably distinguish

between falsehood or rumours from truth and this would be out of the scope of this dissertation.

2.2.5 Fact Checking

To date, there has been several important lines of research into the automated fact-checking and we summarize the relevant topics to get the background knowledge, for example: Check-worthy claims Detection, Relevant-article Extraction, Objectivity Detection, Stance Detection, Source-reliability Detection as well as Claim Validation.

Check-worthy Claims Detection:

The fact-checking process starts with identifying check-worthy claims in a given document that should be prioritized to verify for political debates. This task focuses on recognizing if there is a check-worthy claims in a corpus of sentences from political debates by categorizing them as "non factual", "unimportant factual" and "check-worthy factual" to show their importance for fact checking (Hassan et al., 2015b). In the light of this challenge, the authors presented the first-ever end-to-end automated fact-checking system, called ClaimBuster. This system used named entities, sentiment, TF-IDFs and part-of-speech (POS) tags as features.

Follow-up research by (Gencheva et al., 2017) extended the investigation to add the contextual information of presidential debates (speaker) when detecting the check-worthy claims. Empirically, they have created several features (e.g. TF-IDF, named entities, sentiments, word embeddings, topics, discourse, contradictions, etc,.) and used supervised-based SVM as well as deep feed-forward neural network models.

Jaradat et al. (2018) introduced ClaimRank which is trained on English and further extended to support Arabic. This system used the same features as Gencheva et al. (2017) to detect the check-worthy claims. In a follow-up work, Vasileva et al. (2019) developed a multi-task learning neural network for computing the check-worthiness of statement-level

claims in political debates and predicting the choice of whether each of the nine reputable fact-checking organizations (e.g. PolitiFact, FactCheck, ABC, CNN, NPR, NYT, Chicago Tribune, The Guardian, and The Washington Post) consider prioritizing a check-worthy claim. Given the transcripts of all presidential debates from 1976 to 2016, TATHYA is being developed to identify check-worthy claims (Patwari et al., 2017). This system used sentence-level features including Part-of-Speech tuples, LDA topics, bag of words and entity history.

Two years ago, the first CheckThat! Lab (CTL'18) (Atanasova et al., 2018) attracted the attention of the NLP community with respect to developing systems for the shared task of check-worthy claims detection. After a year, the second shared task of CheckThat! Lab (CTL'19) (Atanasova et al., 2019a) was launched with the focus on the same task of checkworthiness to foster the development of systems predicting which claims from political debates should be given priority to be fact-checked. As a result, 7 and 11 teams actively participated in each of the lab challenges respectively, and submitted their results. Afterwards, they released the code of their systems and published entry papers at the venue of Conference and Labs of the Evaluation Forum (CLEF) in 2018 and 2019 respectively.

In a recent work, the authors in (Lespagnol et al., 2019) revisited the first CheckThat Lab (CTL'2018) to investigate different feature combinations (e.g. word-embeddings, POS tags, syntactic dependency tags, named entities, etc.) using several ML models including Stochastic Gradient Descent (SGD), Random Forests and SVM to predict the information of checkworthines.

The check-worthy claim detection task lacks a large dataset which hinders the ability to explore state-of-the-art DL approaches and that is why most of the related work used traditional ML models. However, these models only use features that are created from claims and they do not capture the interaction between claims and their evidence (e.g. news articles) when detecting check-worthy claims. In this thesis, we propose similar feature-based models but ours detect the stance between claims and relevant articles.

Relevant-articles Extraction:

The Internet is a huge resource which, through search engines, we can use to extract documents of relevant information. In fact-checking, different approaches have been used to retrieve relevant documents or text snippets.

First, the relevant fragments or text-snippet extraction problem was approached from a knowledge-base prospective (Ciampaglia et al., 2015) which restricts the relevant information to the small portion contained in the knowledge graph as a structured data. Second, another approach is proposed to extract the relevant information by using repositories of fact-checking outlets in the web or in an offline database (Hassan et al., 2017; Zhi et al., 2017). This approach is also limited with respect to the manually fact-checked claims and articles composed in those databases.

The third and final approach extracts relevant articles or snippets from the web documents (Popat et al., 2017; Karadzhov et al., 2017; Zhi et al., 2017; Choudhary et al., 2018; Nadeem et al., 2019). One of the key challenge in automated fact-checking is generating a query out of the claim using natural language with the aim to search relevant documents. Previous automated fact-checking approaches mainly rely on full claim sentences (Popat et al., 2017; Zhi et al., 2017) as their queries or use the Rapid Automatic Keyword Extraction (RAKE) module (Choudhary et al., 2018) to extract their queries while other models convert claim sentences into an intermediate form of verbs, nouns and adjectives (Karadzhov et al., 2017; Nadeem et al., 2019) for retrieving relevant documents through search engine.

However, the challenge to extract the relevant information still persists and should be further investigated as mentioned on one of the future works in Chapter 6.

Objectivity Detection:

Objectivity analysis is a common practice used in journalism to eliminate opinionated and biased stories, therefore it can be used to identify subjectivity cues and inflammatory language by analysing the writing style of a text. In journalism, writing should be fair and

factual (Schudson, 1981; Kaplan, 2002) so, objectivity assessment should be taken into account after collecting the relevant documents from the web sources. Despite impressive advances in opinion mining (Bakshi et al., 2016; Sun et al., 2017), current research works on automated fact checking mostly rely on linguistic features to analyse the subjectivity or objectivity language style (Nakashole and Mitchell, 2014; Popat et al., 2017; Zhi et al., 2017; Nadeem et al., 2019). Features include assertive and factive verbs, hedges, implicative words, report verbs, discourse markers, subjectivity and bias lexicon, Wiki-bias lexicon, sentiment cues, etc,.

To capture the subjective or sensational writings, one has to understand the underlying characteristics of the text such as stylistic cues or deceptive signals. On this note, suitable task for objectivity assessment could be fake news classification (Wang, 2017; Horne and Adali, 2017; Pérez-Rosas et al., 2017; Potthast et al., 2017b; Rashkin et al., 2017) which has already been investigated in its own right.

Stance Detection:

When a claim (or headline) is related to an article, it could be expected that the article body will have a stance. One of the key challenges in automated fact-checking is to deduce the stance, e.g. *agree* or *disagree*, of a news article towards a target claim (or a news headline).

Initial efforts by (Ferreira and Vlachos, 2016) proposed an emergent dataset for stance classification as they have investigated the *stance* of news article headline towards its associated claim. Also, Popat et al. (2017) and Zhi et al. (2017) developed a stance classification method with all the unigrams and bigrams (Popat et al., 2017) or embeddings (Zhi et al., 2017) for a text snippet towards a claim either as support or refute. As this thesis focuses on stance detection, we will discuss more about this task in Section 2.5 and 2.6 respectively.

Source-reliability Detection:

The quality and the reliability of sources are important to ensure that web sources are trustworthy because the accuracy of the fact verification is dependent on it. The web source trustworthiness should be taken into account when extracting relevant documents in order to fact-check emerging claims.

Initial truth discovery methods assumed that all sources are equally reliable and they computed a majority vote or the average in multi-sourced data so that they take the majority-agreed value as the truth (Li et al., 2015). In (Nakashole and Mitchell, 2014), a source is considered trustworthy if it reports objective news otherwise it is untrustworthy. The work of (Popat et al., 2017) used the following concept "given the ground truth of the claims, if a source supports true claims and refutes false claims, then it is considered reliable otherwise unreliable".

In line with this, there has been different techniques considered on source quality assessment (1) Zhi et al. (2017) and Choudhary et al. (2018) utilized Web of Trust API[3] to compute the reputation of the sources based on evaluations from third-party services and crowd-sourced ratings (2) FAKTA (Nadeem et al., 2019) used three types of news media (curated by Media Bias/Fact Check (MBFC)) along with Wikipedia (3) and finally, Karadzhov et al. (2017) retrieved a list by manually assessing the 100 most frequent media sources which they collected from many queries and experiments.

A recent work by (Baly et al., 2018) explored the source reliability task in its own right although the problem remains largely under-studied. The authors created a dataset from Media Bias/Fact Check (MBFC) website and they also utilized hand-crafted features generated from various sources (e.g. sample of articles, Wikipedia page, Twitter account, URL structure and web traffic) with the aim to train two separate SVM classifiers for predicting bias and factuality respectively.

There are so many signals that can be combined with the current solutions to better estimate the source reliability. Those other signals could include PageRank, visit history,

[3]www.mywot.com

spamminess, etc. This under-studied problem is out of scope of the current experimental work.

Claim Validation

With the vast amount of data published in the Internet, not all being factual or true, a fact-checker's job becomes difficult even though people would not be able to accurately distinguish fact from false. In the context of automated fact-checking on unstructured textual contents, Popat et al. (2016, 2017) proposed a credibility assessment model for emerging claims. The authors modeled the interplay between the reliability of sources of evidence or counter evidence, objectivity analysis of relevant evidence as well as the stance detection between claims and relevant evidence in providing the final credibility verdict of textual claims.

In yet another work, Karadzhov et al. (2017) proposed a unified framework for automated fact-checking using external knowledge bases to support or reject claims taking the reliability of media sources into account. Given a claim, their setup generates a query from the claim to search through search engines (e.g. Google or Bing) and retrieves a list of relevant web evidence. The system builds text representations of the claims and the snippets of web evidence to automatically train for the task using LSTMs. Finally, the system uses a combination of DL and task-specific embeddings to feed all the hidden representations with pairwise similarities as features to SVM for veracity prediction.

Also, Zhi et al. (2017) proposed ClaimVerif end-to-end system that can judge the truthfulness of a claim. The authors implemented the following components to reach a verdict (1) embedding-based semantic similarity method to extract the relevant evidence, (2) two-step training procedure (e.g. stance classifier and article classifier) to judge whether an article is agreeing or disagreeing a claim (3) and finally, a reweighing module to assess the trustworthiness of sources.

Moreover, Choudhary et al. (2018) presented a unified neural network architecture with different modules that can (1) generate keywords from the claim, (2) extract rele-

vant documents from media sources, (3) compute the trustworthiness of the author and the sources, (4) capture the similarity between claims and relevant credible evidence (5) analyse the sentiment of the relevant evidence with the purpose of assessing the credibility of a given claim. Similarly, an end-to-end fact-checking system defined by (Nadeem et al., 2019) combines various modules to predict the veracity of a given claim (1) top-k evidence retrieval module to retrieve relevant documents from predefined media sources and Wikipedia dump (2) stance classification module to distinguish the stance categories (agree, disagree or discuss) between the relevant evidence and claims (3) linguistic analysis module to compute the objectivity score for the evidence (4) and lastly, the aggregation module to make the final decision.

Within the small body of work on fact-checking, the main issue was how to fact-check the veracity of a claim against the abundant data within the media sources and provide a verdict whether it is true or false, challenges still remain and could be further explored as stated in one of the future works in Chapter 6.

2.3 Text Classification

Text classification is a process of assigning a set of predefined labels to documents by analyzing the text content. Traditional text classification techniques rely on hand-crafted features by transforming the textual information as vectors and different types of ML algorithms for classifying text into predefined categories. General (e.g. N-grams, linguistic, sentiment, POS tags, topic-based, Word2Vec, similarities, etc.) and domain-specific (e.g. lexicon-based, twitter-based, quantifiers, etc.) hand-crafted features have been investigated in rumour stance detection (Aker et al., 2017; Zubiaga et al., 2018b; Ghanem et al., 2019), twitter-stance detection (Zhang and Lan, 2016; Dey et al., 2017; Mohammad et al., 2017), textual entailment (MacCartney et al., 2006; Zhao et al., 2015; Tawfik and Spruit, 2019), fact-checking (Wang et al., 2018; Atanasova et al., 2019b) and hate speech detection (Gitari et al., 2015; Davidson et al., 2017).

RNN models have also been applied successfully in different NLP areas including text classification. For instance, several research studies have utilized LSTMs for the task of rumour stance detection (Kochkina et al., 2017; Veyseh et al., 2017; Zubiaga et al., 2018b), twitter stance detection (Augenstein et al., 2016; Dey et al., 2018; Siddiqua et al., 2019), sentiment analysis (Chen et al., 2017; Baziotis et al., 2017; Ma et al., 2018b) and hate speech detection (Badjatiya et al., 2017; Rizos et al., 2019). In addition to that, some other text classification studies focusing on rumour stance detection (Ma et al., 2016, 2018a), twitter stance detection (Zhou et al., 2017; Hiray and Duppada, 2017; Benton and Dredze, 2018), sentiment analysis (Wang et al., 2016; Jabreel and Moreno, 2017) and hate speech detection (Zhang and Luo, 2019; Founta et al., 2019) have made use of models based on GRUs.

While stand-alone traditional ML and DL models have been exploited in so many text classification tasks, they sometimes do not perform well when trained with smaller datasets. Feature-assisted neural models were applied on several text classification studies (Hiray and Duppada, 2017; Bogdanova et al., 2017; Tommasel et al., 2018; Tawfik and Spruit, 2019) and they have shown state-of-the-art performance where stand-alone models did not produce promising results. Hiray and Duppada (2017) presented a novel feature-assisted neural model based on a combination of GRU and hand-crafted features with the aim to distinguish multiclass agreement classification and achieve state-of-the-art results. Bogdanova et al. (2017) also improved the performance of the answer re-ranking task by proposing a feature-assisted neural classification model that consists of dual GRUs (encoded with questions and answers vectors) incorporated with some discourse features.

Moreover, Tommasel et al. (2018) proposed a feature-assisted neural approach that considers an LSTM sequential model combined with composed features for detecting aggressive social media behaviour content and the combined model improved the results of the stand-alone models. Tawfik and Spruit (2019) investigated a Natural Language Inference (NLI) model by incorporating linguistic and domain-specific hand-crafted features with a Siamese-like neural architecture which relies on sentence pair representations

generated from InferSent or Universal Sentence Encoder (USE) embeddings.

Feature-assisted neural methods have the advantage of capturing both the lexical and semantic information which were found to be effective in sequence modeling tasks (Hiray and Duppada, 2017; Bogdanova et al., 2017; Tommasel et al., 2018; Tawfik and Spruit, 2019). One of our thesis contributions is the systematic integration of semantic information learned through embeddings (e.g word or sentence) and the use lexical-overlaps between claim and evidence pairs into the dual GRU architecture, which achieved state of the art results.

2.3.1 Multistage Models for Text Classifications

Generally, multistage classification strategy divides binary or multiclass classification tasks into two or more stages and addresses them from first-stage to the last using hierarchical classifiers. As stated in the literature of text classification, multistage classification models have significantly improved the performance in different fine-grained NLP tasks including rumour detection (Hamidian and Diab, 2015), sentiment analysis (Alfaro et al., 2016), stance detection (Zhang and Lan, 2016; Wojatzki and Zesch, 2016) and offensive language detection (Park and Fung, 2017).

In rumour detection and classification, Hamidian and Diab (2015) explored the use of single-step classification and two-step classification with hand-crafted features to detect rumour which is a type of fake news. This study observed that two-step classifier significantly outperforms than single-step classifier. In another rumour detection study, Poddar et al. (2018) proposed a conversation-aware two-step classifier to detect the veracity of a rumour. The first-stage determines the stance of a tweet (rumour) in relation to twitter conversations and then based on the predicted stances, the second stage detects the veracity of the rumour. Experimental results showed that a two-stage classifier outperformed compared against the state-of-the-art for both stance detection and rumour-veracity detection tasks.

Multistage models are widely adopted in sentiment analysis. For example, Alfaro

et al. (2016) presented a multistage system with three levels of supervised classification and unsupervised learning. The first-stage and second-stage utilized two supervised learning methods for textual content classification and sentiment analysis (positive, negative or neutral). The third-stage, which is related to opinion mining, generates relevant keywords from the comments using k-means clustering. Similarly, Mukherjee et al. (2012) developed a multistage system that classifies a sentiment into positive, negative or objective using spam filter, spelling checker, pragmatic handler (e.g. hapyyyyyy (happy), guuuuud (good)) and entity detection modules. Nguyen et al. (2013) introduced a two-stage system with a reject option for document-level sentiment classification. The first-stage determines whether a given document is positive or negative and if this model cannot decide the category of the document, it is passed to the second-stage as a rejected document. Those rejected documents are classified as positive or negative using a second-stage classifier. Message-level sentiment analysis system with two levels of classification was also proposed by (Dong et al., 2015). The first-level classifier distinguishes positive from negative sentiment and with the help of these sentiment predictions, the second-level classifier separates between subjective and objective polarities with the purpose to improve the state-of-the-art results.

Some of the stance detection studies on twitter data employed two-stage solutions. Zhang and Lan (2016) and Wojatzki and Zesch (2016) deployed two-stage classification systems where they first predict if a given tweet is neutral and then determine in the second stage the stance polarities as either favor or against. These models follow the traditional ML approach with a set of hand-engineered features and they found a two-stage classification system is suitable for this task. In follow-up work, Dey et al. (2017) proposed the same approach but they utilized subjectivity features in the first-stage and sentiment features in the second-stage to improve the state-of-the-art results. In yet another follow-up work, Dey et al. (2018) explored a two-stage LSTM with attention model to predict the same three classes as they outperform the state-of-the-art results.

Several abusive or offensive language classification studies, mostly using two-stage

classification scenarios, exist in the literature. For example, the work of Park and Fung (2017) described a two-step scenario that first sorts between abusive and non-abusive text, and further classifies the abusive sub-classes using logistic regression and various CNNs. Also, the authors of (Suseelan et al., 2019) developed a two-level system for identifying and categorizing offensive language in social media. Their first-level classifier sorts between offensive or non-offensive while the second-level model uses string comparison based on a compiled dictionary of offensive words to find whether an offensive-tweet (if predicted with less than 70% probability) is again offensive or not. This two-level classification model increased the performance of their system.

Multistage classification approaches can exploit the hierarchical nature of the classes to create multiple stages of classification hierarchies, where each sub-problem, whether it is binary or multiclass, can be addressed in a specific stage classifier with class specific features. This stage-wise strategy can be used to deploy different classifiers for different stages with the aim to optimize per-class and overall performance of the task at hand. Motivated by the aforementioned, multistage classification approaches along with the feature-based and feature-assisted neural models have been proposed and detailed in Chapters 4 and 5.

2.4 Augmentation

Many research areas in NLP and ML have data scarcity or class imbalance problems due to the high cost of manual labeling. Generating new data for minority category is one of the most commonly used methods to balance the class distribution instead of downsizing the training data of the majority represented category. In this section, we provide a brief literature about the augmentation methods adopted for our study.

2.4.1 Text Summarization

The main goal of text summarization is to automatically summarize a long text and produce a shorter version while preserving the most important points expressed in the original text (Radev et al., 2002).

Based on the literature, text summarization has two main research areas: extractive and abstractive summarization. An extractive summarizer selects the most relevant sentences from the given text and generates them as a summary while an abstractive summarizer generates a novel rephrased summary in order to produce a grammatically and semantically coherent shorter version of the original text (Yao et al., 2017). The following established surveys (Jones, 2007; Nenkova and McKeown, 2012; Saggion and Poibeau, 2013) covered a comprehensive review of text summarization.

We chose extractive summarization for our research because the goal is to summarize news articles so that representative sentences of the original content can be generated not a coherent summary. The extracted summary will then be used to create more augmented data for minority classes of FNC-1 dataset (Pomerleau and Rao, 2017). The common process of extractive summarization should start by constructing an intermediate text representation model, then each sentence from the original text should be assigned by a score using proper technique and finally there should be a module to pinpoint the most relevant sentences in order to generate the summary (Nenkova and McKeown, 2012). Some of the text summarization approaches used Bag of Words (BoW) (Radev et al., 2004), Latent Semantic Analysis (LSA) (Ozsoy et al., 2011), Non-Negative Matrix Factorization (NMF) (Lee et al., 2009) as an intermediate text representation methods for the sentence scoring and selection components.

Centroid-based summarization (Radev et al., 2004) was adopted for data augmentation purposes. Instead of BoW representation (Radev et al., 2004), the work of (Rossiello et al., 2017) proposed a centroid-based text summarization using word embeddings with the purpose to inject a bit of semantic understanding into the extraction process. The algorithm first extracts those meaningful tokens from the text where each token is repre-

sented by an embedding vector and then the centroid vector is computed as the sum of the top ranked vectors. To generate an embedding vector for each sentence, the algorithm sums-up the embedding vector of each token in that particular sentence. For generating a sentence score, the algorithm then computes the cosine similarity between the centroid vector and the sentence vector. The algorithm finally generates the summary version of the original text by selecting top ranked sentences iteratively until the target length which can either be given as the number of words in the expected summary or a compression ratio.

2.4.2 Data Augmentation

Data augmentation is a method used to expand the training data or create more balanced datasets and it is designed to transform the original data by altering its contents with the aim to generate new data while class labels are maintained. Data augmentation is commonly used in many tasks ranging from image recognition to speech processing and NLP research areas. For instance, the common data augmentation techniques for image manipulation are geometric and color transformation to increase the number of samples for training by introducing noise into the original data (Krizhevsky et al., 2012). Similarly, techniques such as modifying or speeding the tone of speech signal or creating artificial noise background are the common data augmentation techniques in speech processing (Hannun et al., 2014).

In the context of text classification, Zhang and LeCun (2015a), Zhang et al. (2015b), Mueller and Thyagarajan (2016) and Giridhara et al. (2019) used thesaurus-based augmentation to replace words or phrases of a given text with their synonyms found in Wordnet (Miller, 1995) to create more training data for text classification. Also, Wang and Yang (2015) and Giridhara et al. (2019) proposed a novel data augmentation technique to expand training data for text categorization. The data augmentation technique is based on finding the synonyms of a given text by querying word embeddings (Mikolov et al., 2013) and replacing them with k-nearest neighboring words. Recently, Wei and Zou

(2019) proposed a simple set of Easy Data Augmentation (EDA) methods that can take original labeled data to create more training data for NLP tasks with smaller datasets. For a given labeled sentence, the EDA can generate augmented sentence using text editing techniques such as synonyms replacement, random insertion, random swap and random deletion.

Different from previous augmentation methods, Yu et al. (2018) presented a back-translation data augmentation to enrich the number of training instances. The authors of this study built two translation models to create the paraphrases of original text by translating from English to French, and then back again, from French to English. In addition to that, Kobayashi (2018) developed a bi-directional language model to predict words in relation to textual context after which those words are used as substitutes for data augmentation.

Although zero-shot and few-shot learning techniques are not directly related to data augmentation, they tackle the issue of data scarcity in NLP tasks by taking advantage of the knowledge contained within large-scale pre-trained language models (Qin et al., 2021). These techniques have been shown to be effective in many different tasks (Stappen et al., 2020; Wang et al., 2021; Allaway and McKeown, 2020) but they require tremendous amounts of computational resources (Qin et al., 2021). Zero-shot learning is an extreme case of transfer learning, where a model entails the classification of text into multi-class categories without seeing any training examples for some of the classes (Zhang et al., 2019a). This technique has been applied to textual entailment (Wang et al., 2021), sentiment analysis (Pelicon et al., 2020) and view-point detection (Allaway and McKeown, 2020). Few-shot learning is another form of transfer learning to teach models on how to make accurate predictions where an extremely small number of training examples is available for some of the classes (Schick and Schütze, 2020). This technique has also been used in several NLP tasks such as hate speech detection (Stappen et al., 2020), sentiment analysis (Yan et al., 2018) and view-point detection (Allaway and McKeown, 2020). A future research direction can be to use zero-shot and few-shot learning techniques in order

to overcome the data scarcity and class imbalance problems.

Although the back-translation and bi-directional language models are valid, they are computationally expensive and they generate too much noise which could alter the meaning of the sentence. In Chapter 3, we adopt synonyms-based data augmentation which generates a wide range of substitute words by using GloVe embeddings. The generated words can be replaced with the original text, hence creating more training data for minority classes.

2.5 Multiclass Stance Detection

The Fake News Challenge (FNC-1) (Pomerleau and Rao, 2017) was organized with the aim to assess the veracity of a news claim using ML approaches. More than 50 participating teams from the AI community, fact-checking experts and media journalists gathered to develop fake news detection tools in order to combat misinformation. Acknowledging the complexity of fake news detection tasks, the challenge organizers (Pomerleau and Rao, 2017) noted that tackling the fake-news stance detection problem could be the first step to help prevent the spread of misinformation. It could also assist human fact-checkers to identify incorrect claims by detecting the stance of relevant articles in knowledge-bases.

The stance detection task is an important sub-task in the automated fact-checking pipeline which also has many tasks as presented in sub-section 2.2.2. It is also related to stance classification in Online debates (Thomas et al., 2006; Sridhar et al., 2015), tweets (Augenstein et al., 2016; Mohammad et al., 2017) and rumours (Zubiaga et al., 2018b; Kochkina et al., 2018) that determines the multiclass stances (a.g. pro, against or none) of a pair of text. The stance detection task in social media is to consider pairs of short statement or tweet as an input while news-stance detection is concerned with detecting the stance of news document towards a target claim.

Most of the approaches for document-level stance detection can be classified as Feature-based, Neural-based or Feature-assisted Neural models. Feature-based models

(Ferreira and Vlachos, 2016; Masood and Aker, 2018) usually use lexical cues and ML models to predict the stance of a news body towards its paired headline. Different DL architectures together with hand-crafted features were also proposed to capture the lexical, semantic and the contextual similarity of the headline and body pairs. The SOLAT team (Baird et al., 2017) presented the best performing model in the FNC-1 contest by combining a gradient-boosted decision trees classifier (with various classical features) and a deep CNN. The two other best performing teams (Hanselowski et al., 2017; Riedel et al., 2017) used different architectures of Multilayer Perceptrons (MLP) classifiers along with different hand-engineered features to predict the stance. Thorne et al. (2017) also used a stacked ensemble of five models including the baseline model of the contest in addition with another three MLP architectures.

Subsequent to the FNC-1 contest, Bhatt et al. (2018) combined statistical, external and neural features as they also employed an ensemble of MLP classifiers for the stance detection. A recent trend in DL towards Memory Neural Networks encouraged to deploy an end-to-end memory network (MemN2N) combined with Bag-of-Words (BoW) and its cosine similarity features to improve the performance of the stance detection task (Mohtarami et al., 2018). A thorough feature ablation analysis of the FNC-1's top three systems was conducted by (Hanselowski et al., 2018) where they also proposed a stack LSTM architecture using the best features from the analysis. Xu et al. (2019) conducted a study of transfer learning, called adversarial domain adaptation, from the FEVER domain to stance detection domain as they tried to improve on *agree* and *disagree* classes respectively. Recently, Borges et al. (2019) proposed a deep neural network model for stance detection that is a combination of Bi-directional RNNs, an attention mechanism, max-pooling as well as external hand-crafted features. A recent work by (Conforti et al., 2018) used conditional encoding and co-matching attention neural models to classify the *related*-part (*agree*, *disagree* and *discuss*) of the stance detection pipeline.

However, we observe that the increase in complexity of deep neural layers and memory networks would overfit on smaller datasets. Also, the highly imbalanced distribution

affected those models and they were ineffective in predicting stances of underrepresented classes where these stances are considered important to the task of fake news detection. In Chapter 3, we present different feature-based models and a feature-assisted neural model where in we use a simple dual GRU enhanced with regularization methods as well as different lexical and similarity features. We also used text augmentation techniques to enlarge the data for minority classes as explained in Section 2.10.

2.6 Multistage Stance Detection

Most of the models in the literature proposed multi-class classification strategy where they frequently predict the majority class as they struggle to prevent the bias induced due to class imbalance problem. Hanselowski et al. (2018) provided an overview about the top three models participated in the contest as they largely applied an ensemble of different lexical features with MLPs (Hanselowski et al., 2017; Riedel et al., 2017) and a weighted ensemble of gradient-boosting trees and deep CNN (Baird et al., 2017). Bhatt et al. (2018) used an ensemble of MLP classifiers with lexical and neural features while Saikh et al. (2019) presented an MLP model based on universal sentence encoder incorporated with textual entailment features. Borges et al. (2019) proposed a stance classifier based on an ensemble of Bi-directional RNNs, an attention mechanism, max-pooling and lexical features. Similar to the prior models, Chapter 3 presents an ensemble of GRU model with various lexical features and we also perform data augmentation using text summarization and synonymous replacement methods to enlarge the samples of minority classes. However, we observe that all of these models proposed multiclass classification strategy where they frequently predict the majority class as they struggle to prevent the bias induced due to class imbalance problem.

Multistage approaches have also been proposed to achieve higher performance compared to multi-class classifications. (Bourgonje et al., 2017) proposed a multistage rule-based and logistic regression classifier with a set of rules and n-gram matching to determine

the relevance and stance classifications. (Masood and Aker, 2018) used 17 lexical features including sentiment, BoW, co-occurrences and similarities to experiment with two-stage and three-stage classifications. However, these studies only considered lexical features which makes hard to improve the performance of the related classes.

In addition, (Zhang et al., 2019b) introduced a two-layer DL framework that jointly optimizes the hierarchical representation of relevance classification layer and stance classification layer by controlling the distribution discrepancy between the layers using Maximum Mean Discrepancy (MMD) regularization technique. More recently, other studies have proposed stance detection models based on general purpose language models by fine-tuning the pre-trained sentence embeddings like BERT (Jwa et al., 2019) or multi-task learning (Fang et al., 2019). However, these models leverage massive external corpora and they are extremely computationally expensive. In Chapter 4, we adopt feature-assisted neural approaches that combines with lexical-overlap and DL features in a multistage setting.

2.7 Multistage Poltitical Fake Statement Detection

Political Fake Statement Detection is a sub-task of fake news classification which has focused on analysing the degree of truthfulness - whether a short statement expresses one of the multiple predefined categories as defined in the LIAR dataset (Wang, 2017). Since the author presented this publicly available dataset, related work has proposed different models employing various NLP techniques (Wang, 2017; Long et al., 2017; Liu et al., 2019; Wang et al., 2019b; Alhindi et al., 2018; Karimi et al., 2018).

The work described in (Wang, 2017) presented a hybrid CNN model that combines the statement and the metadata from the speaker profile for fine-grained classification. Long et al. (2017) conducted experiments which incorporated the metadata and the statement topics as attention networks with LSTM and the statement as LSTM into a hybrid neural model. Wang et al. (2019b) utilized CovNet and Multi-head self attention to cap-

ture the textual content and the context dependent information from speaker profiles as they get certain improvements compared to the state-of-the-art. Liu et al. (2019) focused on fine-tuning BERT with this fine-grained dataset using a two-stage model. All of these models make use of metadata (e.g., speaker, party, state, subject and context/venue) and credit history of the speaker which have the potential to significantly improve the accuracy but both may not be available in real-time (Thorne and Vlachos, 2018).

Recent years, a number of NLP models have been proposed for the purpose of document-level stance classification, including StackLSTM (Hanselowski et al., 2018), Memory Networks (Mohtarami et al., 2018), feature-assisted GRU model with data augmentation (refer to Chapter 3) and multistage classification model using lexical and neural features as described in Chapter 4. A step towards this direction, Karimi et al. (2018) presented MMFD framework which stands for Multi-source Multi-class Fake news Detection, therefore combining statements, speaker profile, credit history and verdict reports (justification) in order to discriminate between the fine-grained categories of LIAR dataset. Similar to this, Alhindi et al. (2018) used statement, metadata and justification to improve the classification accuracy regardless of the model - whether it is feature-based or DL model. However, the accuracy of these models is largely dependent on credit history as much as justification and they also use a multiclass classifier to distinguish between fine-grained classes. In Chapter 5, we apply our multistage feature-assisted neural models into the multiclass fake statement detection task by automatically assessing the veracity of claims in relation to their evidence (justification).

2.8 Related Methods

The work presented in this thesis makes use of feature-based and neural based methods for stance detection. In this section, we briefly introduce different lexical features, lightGBM classifier and several neural methodologies.

2.8.1 Text Representations based on Statistical Analysis

In this thesis, the aim is to apply ML approaches to fake news detection tasks. For a classification task, ML models require vector representations with a fixed dimension as an input and since most documents differ in textual length, there are NLP techniques that can help transform textual features such as word, phrase, sentence or long document into a vector format. In this section, we present statistical representations of lexical and similarity features adopted for this work in order to produce the necessary vectorized input for our feature-based models.

The BoW representation model is one of the commonly used techniques in NLP with the aim of generating document representation as a bag of unique words where the frequency of every unique word is kept as a feature. In other words, this is a common feature which counts the Term Frequency (TF) or the occurrences of a term that appears in a text. The BoW model does not capture the order of the words which is important and it also ignores the spatial information present in the textual documents. With respect to spatial information, an n-gram model can be used to represent the co-occurrences of n words in a text document. Instead of weighing some of the important terms towards a document, BoW model considers them equal and as the terms or co-occurrence terms increase, then the vocabulary size increases. Term Frequency-Inverse Document Frequency (TF-IDF) is an alternative model of BoW which introduces IDF. If a word appears more frequently in most of the documents, TF-IDF recognizes it as a common word, otherwise rare words that appear in several documents will be considered an important word. As a result, this combined TF-IDF model eliminates commonly used words such as stop words and gives more weight to those rare words that carry more information to certain textual documents. Overall, the BoW model can capture word counts but fails to represent the semantic relationship between two words.

Topic modeling is an unsupervised learning algorithm which has the ability to represent original text with the aim to discover main topics and their associated words that occur in a collection of documents. The goal of this kind of algorithm is to reveal the

correlation between most important words that appear in a large body of text. There are common tools in NLP that can be used for representing hidden semantic patterns and clusters of unstructured texts. For instance, Latent Semantic Indexing or Analysis (LSI) (Deerwester et al., 1990) and Non-negative Matrix Factorization (NMF) (Lin, 2007) are topic modeling algorithms from linear algebra for discovering latent structure in textual documents. LSI takes a huge term-document matrix and performs a matrix decomposition with a dimensionality reduction method called singular value decomposition (SVD) to identify the relationships between terms and documents. This algorithm generates three matrices from the original text as *term x topic* matrix, the weight of *topic x topic* diagonal matrix and finally *topic x document* matrix. Similar to LSI, NMF uses term-document matrix input with no negative elements to generate two non-negative matrices as *term x topic* matrix and *topic x document* matrix. Yet another topic modeling for mining large bodies of text exists which is a probabilistic generative model called Latent Dirichlet Allocation (LDA) (Blei et al., 2003). The idea behind this algorithm is that documents are created based on a mixture of topics where each topic is identified by a collection of terms. LDA follows two main steps to discover the topics underlying a set of documents. In the first step, it goes through term-document matrix and assigns each term to a random topic. In the second step, the algorithm draws a term from the list of terms corresponding to a topic and re-assigns that term to another topic as it repeats the process again for all terms.

After generating the vector representations of two statements using BoW, TF-IDF and topic models, we can compute the lexical and topic similarity between them with cosine, Jaccard and Euclidean similarities that have been successfully applied in sentence pair modeling (Unankard et al., 2015; Wang et al., 2017a). With respect to textual overlaps, the word and character overlaps (Pomerleau and Rao, 2017) as well common entity measures can also be employed to compute the co-occurrences of the words, characters and entities between two given statements in order to determine their relatedness. In addition to that, polarity feature could be a strong signal that can be used to identify

subjective or extreme words present in fake statements by analyzing the emotional reaction of the text as well as its negative and positive dimensions. Some of the lexicon-based sentiment analysis such as VADER (Hutto and Gilbert, 2014), Depechemood emotion (Staiano and Guerini, 2014) and MPQA subjectivity lexicon (Wilson et al., 2005) can be utilized to extract the polarity features presented in the textual content. In NLP, a dictionary-based approach can be utilized to search and count the frequency of a given list of words throughout the textual documents. Some of the previous research deals with lexico-syntactic features to analyse certain words that belong to certain categories from the stance-taking language expressed within the statement. Fake news textual contents usually have certain words that indicate the presence of negation, certainty, doubt or discussion which is very useful in discriminating between the truth or false statements. Some of the dictionary markers belong to refuting words (Pomerleau and Rao, 2017), hedging words (Mukherjee and Weikum, 2015), discuss words (Ferreira and Vlachos, 2016) as well as implicative, positive and negation words (Ghanem et al., 2019).

2.8.2 Embeddings for Text Representation

Generating vectors using BoW or topic models may simply give reduced vector representations of textual document but they may also lose semantic and similarity information regarding words with semantically identical meanings which topic models could transform them as in different topic distributions. Word vectors are considered one of the efficient ways to represent the semantic information of textual documents. Essentially, words that occur in similar contexts tend to be mapped very closely to each other in a vectorized space. Therefore, the embedding model has the advantage to capture the similarity of words by capturing their surrounding context and the linguistic relationships between them.

For instance, Word2Vec (Mikolov et al., 2013) is a pre-trained unsupervised algorithm that is based on a two-layer neural network model with the purpose for learning word vectors from given text corpus. Word2Vec has two learning models (1) Continuous

Bag-of-Words (CBoW) which predicts the target word by looking at the context window and (2) skip-gram which predicts the context window based on the given word. GloVe embeddings (Pennington et al., 2014) is also a pre-trained unsupervised learning model that uses word to word co-occurrence counts in a corpus and it also has the advantage of combining features from global matrix factorization and the Word2Vec local window context. Word embeddings gained their popularity by transforming a given textual document into a fixed size distributed vector representation as an input for DL models to classify predefined document categories.

In addition to that, these embedding models can be used to generate the averaged embedding vectors for longer pieces of text (e.g. headline and news body) as a feature for shallow models and we can also compute the similarity between two averaged vectors using cosine similarity. As the computation of cosine similarity feature is based on the averaged vectors of textual documents, important information could be lost during the process of averaging. However, word mover's distance (WMD) (Kusner et al., 2015) is an alternative method that measures the distance between two textual documents. This distance-based method computes the dissimilarity between two textual document as the minimum cumulative distance that all vectors or tokens of document1 need to travel to another vectors from document2. For our work, we are going to take advantage the services of word embeddings for all of the aforementioned purposes.

Word embeddings have become important in many down-stream NLP tasks using a simple weighted average of all word vectors in a sequence to obtain a sentence representation. The main problem with word embeddings is their way of representing each word within the embeddings regardless of the context. Besides word vectors, recent work (Conneau et al., 2017) proposed sentence embeddings to capture the relationships between multiple words and phrases in order to generate vector representations for sentences and much larger input sequences. Sentence embeddings are able to generate vectors for sentence by taking into account the order of words within the sentence and capture much larger context in one vector. Learning sentence encoders can be unsupervised such as

SkipThought (Kiros et al., 2015) and FastSent (Hill et al., 2016), supervised such as the InferSent (Conneau et al., 2017) or partially supervised and unsupervised such as the Universal Sentence Encoder (Cer et al., 2018).

The commonly used sentence embeddings can be employed in one of two strategies: feature-based or fine-tuning (Devlin et al., 2018; Peters et al., 2019). In the feature-based approach, sentence representation are generated to be fed directly to the model as input vectors whereas in the fine-tuning approach, it is trained on a down-stream task by fine-tuning on the pre-trained sentence embeddings parameters. We adopt the Universal Sentence Encoder (USE) (Cer et al., 2018) as a feature-based pre-trained sentence representation model that can be used to extract vector representations for multi-sentence sequences. It can be incorporated into down-stream models to compute their sentence embeddings in two variations: Transformer-based Encoder and Deep Averaging Network. In our Chapter 4 experiments, we use the first encoder in relation to being performed competitively in various NLP tasks (Tawfik and Spruit, 2019; Saikh et al., 2019).

2.8.3 ML Classifier - LightGBM

LightGBM (Light Gradient Boosting Machine) (Ke et al., 2017) is one of the powerful GBDT (Gradient Boosting Decision Tree) (Friedman, 2001) implementations widely used in the data science community because of its predictive performance. The tree-based algorithms such as the baseline GBDT (Friedman, 2001) and XGBoost (eXtreme Gradient Boosting) (Chen and Guestrin, 2016) are other best performing classes of boosting algorithms. They are from decision tree family where the data split into branches using tree based hierarchies by predicting the value of each node as a single label. With the purpose to generalize an individual weak decision tree, the GBTD combines the prediction values of several decision trees in order to prevent overfitting of the dataset. LightGBM and XGBoost are two of several implementations from a GBDT sub-family and they have been competitively implemented in kaggle[4] data science competitions. In this study, we

[4]https://www.kaggle.com/

use a LightGBM, an ensemble of many weak learners (decision trees) that grows its tree as leaf-wise (vertically) compared to other tree-based algorithms which grow their trees as level-wise (horizontally e.g. GBDT and XGBoost). As the literature suggests (Yang et al., 2018; Li et al., 2019b; Al Daoud, 2019), the lightGBM model gives better predictive performance for data science projects compared to other GBTD implementations and it uses a histogram-based algorithm to make this model memory-efficient and faster in training-time.

2.8.4 Neural Networks

This section presents DL techniques used in this research with the aim to significantly improve upon the state-of-the-art performances of fake news detection tasks.

Multilayer Perceptrons (MLP)

MLPs are a family of feedforward neural architecture that consist of several layers with a set of nodes to solve complex problems. This type of shallow neural network has a single input and output layers as well as one or more hidden layers, forming fully connected layers (Goodfellow et al., 2016). Each node or perceptron from a layer has a directed connection to the next layer with respect to a certain weight between the layers. The training procedure of a MLP classifier relies on a supervised learning method called backpropagation that feeds input features forward through the network layer-by-layer until the nodes in the final layer which decodes the output labels and propagates the errors back to previous layers in order to adjust the weights and reduce the error value. The softmax or sigmoid output function is used for multiclass or binary classification respectively. Gradient descent is a method used to fine-tune the weights with the purpose to minimize the error function. In Chapter 3, we use shallow MLP model which can classify data that is not linearly separable because of the multiple connected layers and the non-linear activation functions.

Recurrent Neural Networks (RNNs)

RNNs are DL architectures that take sequential data of variable-length as input in current time-step and add information from previous time-steps. In this way, RNN is able to learn from the past data and it preserves the context of the relevant information. The basic RNN unit is illustrated in the following equation where $\mathbf{x}_t$ is the current input, $\mathbf{h}_{t-1}$ stands for the previous hidden state or the output of previous time-step, t represents the time step, $\mathbf{W}^h$ and $\mathbf{b}^h$ denote weight and bias vectors and finally f is the activation function.

$$\mathbf{h}_t = f(\mathbf{W}^h\mathbf{x}_t + \mathbf{W}^h\mathbf{h}_{t-1} + \mathbf{b}^h)$$

Unfortunately, RNNs can only carry so much information from the previous time steps but not information of long sequences because of the vanishing gradients problem where the gradient vector decays exponentially to zero, thereby making the learning of long-term dependency difficult. Two variants of RNN models were proposed including Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) and Gated Recurrent Unit (GRU) (Cho et al., 2014) with the purpose to prevent the vanishing gradients problem. Both models have different ways of gating mechanisms to track the long-term dependencies for multi-sentence sequences. LSTM has a memory unit $\mathbf{c}_t$ at time step t and three gating mechanisms (i.e. input $\mathbf{i}_t$, forget $\mathbf{f}_t$ and output $\mathbf{o}_t$ gates) to control how much of the new sequence should be kept and how much of the old sequence to forget or discard from the memory. $\mathbf{h}_t$ represents the output and $\mathbf{c}_t$ stands for the cell state (or long-term memory) at time step t.

$$\mathbf{i}_t = \sigma(\mathbf{W}^i\mathbf{x}_t + \mathbf{W}^i\mathbf{h}_{t-1} + \mathbf{b}^i),$$

$$\mathbf{f}_t = \sigma(\mathbf{W}^f\mathbf{x}_t + \mathbf{W}^f\mathbf{h}_{t-1} + \mathbf{b}^f),$$

$$\mathbf{o}_t = \sigma(\mathbf{W}^o\mathbf{x}_t + \mathbf{W}^o\mathbf{h}_{t-1} + \mathbf{b}^o),$$

$$\mathbf{g}_t = \texttt{tanh}(\mathbf{W}^g\mathbf{x}_t + \mathbf{W}^g\mathbf{h}_{t-1} + \mathbf{b}^g)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t,$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \texttt{tanh}(\mathbf{c}_t)$$

The GRU model is a simplified version of LSTM and also widely adopted in sequence modeling. GRU has two gates to regulate the flow of the sequence for instance, the reset $\mathbf{r}_t$ gate controls how much of the old sequence in the previous state is combined to the current hidden state and the update $\mathbf{z}_t$ gate controls how much information to be kept from the previous state and the current hidden state. However, GRU has fewer parameters and is faster to train than LSTM since it has only two gates compared to LSTM which has three gates as illustrated on the above equation.

$$\mathbf{r}_t = \sigma(\mathbf{W}^r\mathbf{x}_t + \mathbf{W}^r\mathbf{h}_{t-1} + \mathbf{b}^r),$$

$$\mathbf{z}_t = \sigma(\mathbf{W}^z\mathbf{x}_t + \mathbf{W}^z\mathbf{h}_{t-1} + \mathbf{b}^z),$$

$$\mathbf{g}_t = \texttt{tanh}(\mathbf{W}^g\mathbf{x}_t + \mathbf{W}^g(\mathbf{r}_t \odot \mathbf{h}_{t-1}) + \mathbf{b}^g),$$

$$\mathbf{h}_t = \mathbf{z}_t \odot \mathbf{h}_{t-1} + (1 - \mathbf{z}_t) \odot \mathbf{g}_t$$

The main difference is that GRU controls the sequence dependencies over different time-steps without having to use a memory unit like LSTM does. The performance of GRU for sequence modeling is on par (Cho et al., 2014; Chung et al., 2014; Jozefowicz et al., 2015) or sometimes better (Yin et al., 2017; Jabreel and Moreno, 2017) compared with LSTM and it is a good choice to efficiently train on smaller datasets. Motivated by these observations, we make use of the GRU model and significantly improve upon it by integrating lexical-overlap features as presented throughout the thesis main chapters.

Regularization Techniques

A supervised learning model commonly overfits when the model tries to memorize every pattern or sometimes noise from the training data instead of learning general features

which negatively affects the prediction on unseen data, this also called a generalization error. If acquiring more data is expensive, there are different regularization techniques used to prevent models from overfitting which could either assess the performance of the model on unseen examples or penalize the complexity of the models. Early stopping could be used as a form of regularization to stop the training when the model does not improve the performance on the validation (e.g. on unseen validation data) for a limited number of iterations, in other words, evaluating the performance of the model. Other forms of regularization, such as Dropout and Gaussian noise, are utilized to modify the complexity of the learning algorithm and improve their generalization capability. Gaussian Noise can be applied after concatenating neural and lexical features for the purpose of introducing random noise to each training sample. This layer increases the size of the training data and the model learns more general features from both inputs at the hidden layer, hence working as data augmentation (Tommasel et al., 2018; Zhang and Yang, 2018). The model generalizes better when the Gaussian noise is applied at the hidden layer rather than at the input so as to mitigate overfitting[5]. Dropout is another common regularization layer that randomly removes certain neurons from the network during training (Srivastava et al., 2014). Therefore, this technique prevents overfitting because the network is able to learn the same features in different ways, which leads to a better generalization. In our study, we adopted all aforementioned methods to prevent our DL models from overfitting.

2.9 Conclusion

In this chapter, we reviewed the existing literature of similar works to our research. We started by first reviewing the different domains of fake news detection, we continued by reviewing text classification, data augmentation as well as the related work of stance detection and political fake statement detection, and we concluded the chapter with a discussion of traditional ML and DL models. In the subsequent chapters, we present our

[5]towardsdatascience.com/noise-its-not-always-annoying-1bd5f0f240f

proposed methods and their experimental evaluations. The next chapter introduces the proposed feature-based methods, data augmentation and feature-assisted neural model for multiclass fake-news stance detection. An empirical analysis of the proposed methodology is provided.

CHAPTER 3

MULTICLASS NEWS-STANCE DETECTION

We conduct empirical assessment of different feature-based ML and feature-assisted neural models as multiclass classification setup with the aim to boost the performance of fake-news stance detection. The chapter is structured as follows. Section 3.1 formulates the stance detection problem and Section 3.2 presents the details of the proposed models. Section 3.3 and 3.4 discuss our experimental procedure and results. Finally, Section 3.5 and 3.6 draw some discussion and conclusion for our work.

3.1 Introduction

The goal of the stance detection task is to determine the "perspective" stance of two pieces of text (e.g. headline and body) as *agree*, *disagree*, *discuss* or *unrelated*. Recent advances on news-stance detection have mostly employed an ensemble of feature engineering with MLPs (Riedel et al., 2017; Thorne et al., 2017; Bhatt et al., 2018), CNNs (Baird et al., 2017; Xu et al., 2019), RNNs (Hanselowski et al., 2018; Borges et al., 2019) and Memory

Networks (Mohtarami et al., 2018). These complex ensembles with deep neural layers, attention networks and memory networks have been shown to achieve state-of-the-art performance. However, the increase in complexity of these neural models tends to overfitting on smaller datasets. In addition to that, we observe that previous classic and deep neural models (Pomerleau and Rao, 2017; Masood and Aker, 2018; Baird et al., 2017; Hanselowski et al., 2018) achieved better classification F1-score on *unrelated* and *discuss* classes with above 97% and 75% respectively but they also struggle at predicting the *disagree* class (e.g. 0-18%) (Hanselowski et al., 2018) and the predictions of *agree* class often fall short (e.g. 0-50%) (Hanselowski et al., 2018) as well because of the dataset's imbalanced class distribution.

Inspired by feature-assisted neural models (Hiray and Duppada, 2017; Bogdanova et al., 2017; Tommasel et al., 2018; Tawfik and Spruit, 2019) that apply DL models in combination with external features to sentence modeling tasks, we present feature-assisted GRU model. We also explore other potential methods which can reduce textual noise and generate more training examples for underrepresented classes in order to avoid overfitting and improve the models' robustness.

Our contributions are summarized as follows: (a) We propose feature-based models (e.g. LightGBM and MLP) with lexical and similarity features and provide a comparative report as they achieve state-of-the-art performances. (b) We use text augmentation techniques to enlarge the data of underrepresented classes through label-preserving transformations for better prediction. (c) We also combine the various important features with a dual Gated Recurrent Unit (GRU) model, fine-tuned using Batch-normalization and Gaussian noise, to better make predictions for news-stance detection task. (c) Experimental results show that this combined model with GloVe embeddings outperforms all previous models in all of the evaluation settings (e.g. the FNC-1, Accuracy and the Macro-F1 metrics) on the FNC-1 dataset. (d) We also conduct cross-domain validation using FNC-1 and ARC datasets.

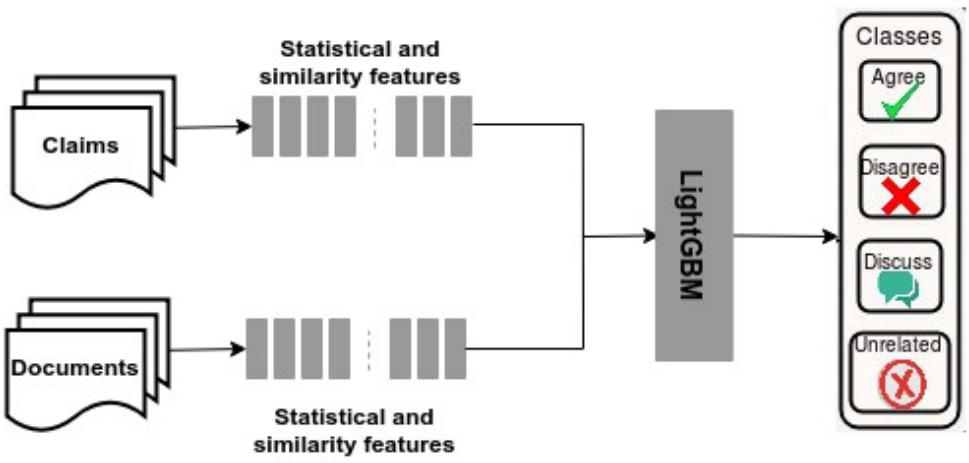

Figure 3.1: Feature-based LightGBM model

3.2 Methodology

The proposed models consist of three different modules for stance detection: (1) feature-based models with hand-crafted features (2) data augmentation (3) and a feature-assisted neural model. First, we build feature-based models to learn how the rich input representations of lexical and similarity information can help in this task. Second, we implement a data augmentation strategy to create new samples for minority classes with the aim to enhance their performance. Third, we present an approach that is based on DL combined with hand-crafted features previously shown to be beneficial for the stance detection. We describe the details of these modules that we adopt for the news-stance detection.

3.2.1 Feature-based Models

Our first approach uses Light Gradient Boosting Machine (LightGBM) and Multilayer Perceptron (MLP) classifiers to distinguish the multiclass stances between headlines and news documents. The architectures of these feature-based models are shown in Figure 3.1 and 3.2 respectively. LightGBM is a popular classifier that can be trained faster than other traditional approaches with a superior performance as shown in several NLP tasks.

Also, MLP is a well known feature-based neural model that is very often employed

Figure 3.2: Feature-based MLP model

as a baseline in the DL world. These feature-based models utilize a combination of lexical and similarity features. They capture the presence of word, sentiment, emotions and linguistic relations as well as lexical overlap between the text. They also provide more understandable results with regards to each feature's contribution which can easily be quantified about its performance. We conduct feature ablation study to extract the best lexical features by using LightGBM classifier. The two parts of Table 3.1 shows the description of the lexical and similarity features presented in this study.

In our feature-based approaches, we apply different pre-processing methods to create number of features composed in different groups from the dataset. To normalize words and remove noise in sentences, the text of news articles are tokenized, lowercased, and removed the stopwords, non-alphabets and punctuation in order to clean the dataset as we create Lexical and similarity features. In the following subsections we elaborate the details of our features.

BoW features

We enrich feature-based approaches with the Term Frequency (TF) representation of headlines and articles to provide additional relevant information however we only retain 3000 most frequent 1, 2 and 3-grams. In addition, we generate the cosine similarity

between headlines and documents BoW vectors. Besides the use of cosine distance to generate the similarity between headline and body 3000 most frequent BoW vectors, we also add Jaccard and Euclidean similarity distances that have been successfully applied in detecting Duplicate Questions (Baldwin et al., 2016). In addition, we use TF-IDF to obtain the representations of headlines and articles with the purpose to compute the cosine similarity between them. We also use BoW vectors as a negation handling (Das and Chen, 2007; Hanselowski et al., 2018) for example - we add a "_NEG" tag as a prefix for those 500 most frequent BoW that appear after a negative keyword (e.g. "no", "not", "don't") within the border of a clause-level punctuation mark.

Topic models

We use topic models such as LDA, LSI (SVD) and NMF. We first transform our text (e.g. title and body) into TF vectors for LDA and TF-IDF vectors for LSI and NMF. The challenge lies on how to generate the optimal number of topics required to improve the accuracy of the model. We examine different inputs (e.g. 50, 100, 200 and 300) to make sure we choose the number that leads to the highest scores. Therefore, we apply LDA, LSI and NMF algorithms to generate topic vectors and the optimal number of topics are set to be 100, 100 and 50 respectively. For each of the above topic models, we compute the cosine distance between the title and body vectors. Unlike other studies (Hanselowski et al., 2017, 2018) that use headline and article topic vectors as features, we only use the similarities between the topic vectors.

Word-counts features:

Based on different lexicons such as refuting words (fake, fraud, hoax, deny, etc,) (Pomerleau and Rao, 2017), hedging words (about, claim, essentially, perhaps, etc.) (Mukherjee and Weikum, 2015) and agreeing words (confirm, support, evidence, demonstrate, etc.), we compute how many of these signals appear in the headlines and their relevant bodies.

Table 3.1: Set of features used in this study

Lexical Features
overlapping (CW) character and word ngrams between headline and body
overlapping (WT) word ngrams of top 25 TF-IDF-body-vectors in the headline
(RC) refuting, (AC) agreeing and (HC) hedging word counts of headline and body text
(CE) common entities between headline and body text
(TP) polarity and (TS) subjectivity in the headlines and the top 25 TF-IDF body
(ES) emotional scores and (PS) polarity scores in each of the text
(BW) 3000 most common 3-gram BoW vectors from the headline and body text
(BW) negated-sign for 500 most frequent 2-grams BoW if negation found on text
Similarity Features
(BW) cosine, Jaccard and Euclidean similarities between headline and body BoW
(TC) cosine similarity feature between headline and body 3-grams TF-IDF vectors
(SC) cosine similarity feature between 100 headline and body SVD components
(LC) cosine similarity feature between 100 headline and body LDA topics
(NC) cosine similarity feature between 50 headline and body NMF topics
(WM) WMD similarity between the headline and body word-embedding vectors

Sentiment and emotion features

We compute scores for sentiment polarity, subjectivity and emotion features of the claim and the justification via VADER (Hutto and Gilbert, 2014) sentiment analysis and Depechemood emotion detection (Staiano and Guerini, 2014). We also generate polarity and subjectivity features by using the textblob library.

Word, character and entity co-occurrence features

For the co-occurrence feature, we extract the overlapping word and character ngrams between headlines and articles (Pomerleau and Rao, 2017). Using the spaCy toolkit, we also generate bag-of-entities representation from the statement and justification after which we then compute the co-occurrence or overlapping of common entities between the pair.

Word Embeddings

We extract a similarity feature based on Word mover's distance (Kusner et al., 2015) that uses word embeddings (Mikolov et al., 2013) to represent headlines sand bodies as

explained in Section 2.8.3 of Chapter 2.

3.2.2 Data Augmentation

Data augmentation is a common way to expand the number of training instances in order to avoid overfitting and control generalization error for ML models. As the amounts of *disagree* and *agree* training instances in FNC-1 dataset are very small compared to *unrelated* and *discuss* pairs, we paid a special attention to text augmentation techniques to enlarge the minority classes through label-preserving transformations. We hypothesize that summarizing news article body will help reduce the noisy text as extractive summarization (Rossiello et al., 2017) can produce a summary while preserving the meaning of the original text.

Our data augmentation approach involves using an extractive centroid-based text summarization technique (Rossiello et al., 2017) to summarize text pairs of minority classes. The centroid is the document vector which is computed as the average word embeddings of the most common words occurring in the document. To generate the summary sentences, this algorithm selects the closest vectors (sentences in the document) which have vectors similar to centroid embeddings.

In addition, the summarized version of the news body and the headlines can be used to create new training instances by replacing random words with their synonyms from the thesaurus (Zhang and LeCun, 2015a; Zhang et al., 2015b; Mueller and Thyagarajan, 2016). In contrast to other methods (Wang and Yang, 2015; Giridhara et al., 2019), we use GloVe embeddings (Pennington et al., 2014) to find synonyms for randomly-chosen words from the summary and the headlines after which we then replace randomly 30% of the original words with their synonyms.

Table 3.2 shows a data augmentation example from the dataset using text summarization and threshold-based synonyms replacement. The first part of the table presents the original version of the claim and the news document. We can see the generated summary of the document from the second part of the table and it does not change the

Table 3.2: An illustrative example of the augmentation process

***Claim**: Sugarhill Gang rapper Big Bank Hank dies aged 57*	
Stance	**Body**
Disagree	Big Bank Hank ... the Sugarhill Gang rapper best know as "the grandmaster with 3 MCs that shock the house for the young ladies"... died early Tuesday morning ... TMZ has learned. Hank ... born Henry Jackson ... had been suffering from cancer. We're told he passed away in the greater NYC area around 2 AM. Hank was one third of the famous Sugarhill Gang ... which had the first mainstream rap hit "Rappers Delight" in 1979. His group mates Wonder Mike and Master Gee told TMZ ... "So sad to hear of our brother's passing. Rest in peace Big Bank." Hank was 57 years old.
Stance	**Summary**
Disagree	Big Bank Hank ... the Sugarhill Gang rapper best know as "the grandmaster with 3 MCs that shock the house for the young ladies"... died early Tuesday morning ... TMZ has learned. Hank ... born Henry Jackson ... had been suffering from cancer. Hank was 57 years old.
***Augmented Claim**: Sugarhill Gang rapper Big Bank Hank died age 57*	
Stance	**Augmented Summary**
Disagree	Big Bank Hank we of Sugarhill Gang rapper better know also "the chess with 3 MCs but shock of houses time the young ladies"... died late Tuesday morning ... TMZ has learned. Hank ... born Henry Jackson ... had been suffering back cancer. Hank being 58 years old.

meaning of the document. We then generate 3,678 and 840 augmented text samples for *agree* and *disagree* classes with respect to the claim and news documents by using a synonyms replacement method. Although the transformed samples generate some amount of noise that might not be linguistically correct, it is still extremely helpful for training our model with the additional augmented samples. It is because the model learns new words to generalize the task and prevent overfitting with the aim to improve class-wise performance as well as the overall performance of the system.

3.2.3 Feature-assisted Neural Model

With the data imbalance problem, feature-based approaches demonstrate higher accuracy on majority represented classes. However, the results of feature-based MLP along with some of the NLP literature that shows the usefulness of combining lexical-overlap features in DL (Attardi et al., 2017; Bogdanova et al., 2017) inspired us to build a feature-assisted neural model. Therefore, the proposed methodology is a combined model of dual GRU

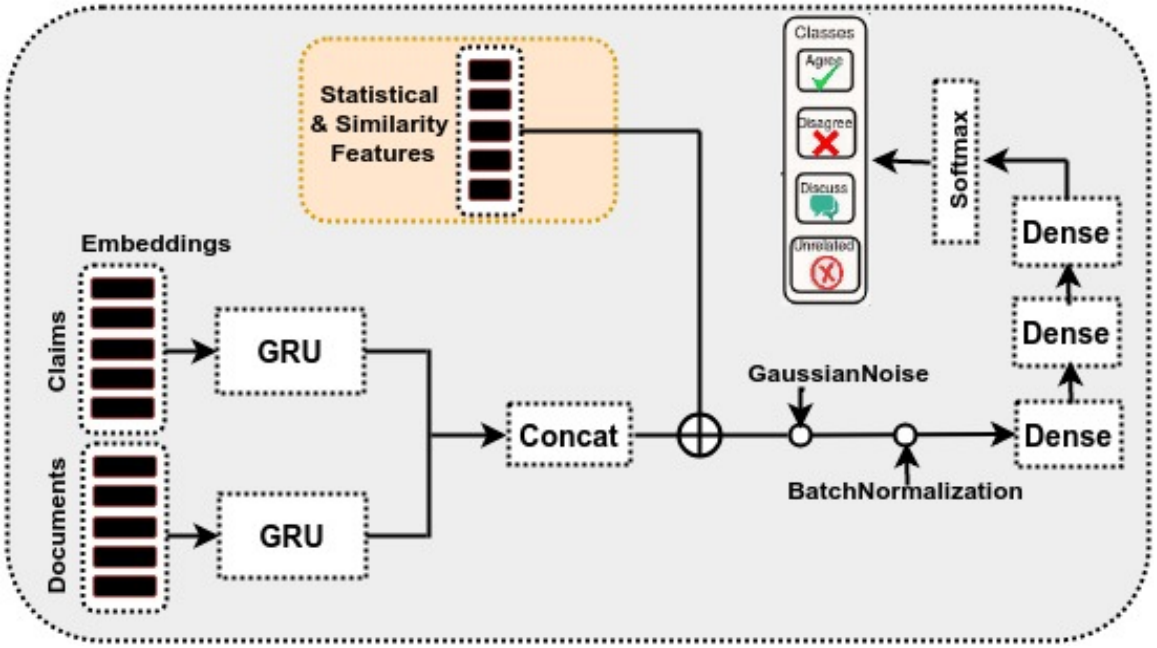

Figure 3.3: Feature-assisted Neural model

(Cho et al., 2014) and feature-engineering heuristics fed through a fully connected MLP network to predict the stance of four-class classifications. We also made use of regularization and text augmentation techniques to improve the overall performance of the model.

To improve the performance of feature-based models, we train dual GRUs model on top of pre-trained word embeddings to generate 100-vector (e.g. headlines and bodies) input sequences. We use 50-d GloVe embedding together with a single-layer GRU of 64 neurons for each of the headline and body vectors as we set the probability of dropout and recurrent_dropout to 0.2 and 0.1 respectively. This is followed by a concatenation layer with handcrafted features, a batch-normalization layer, a Gaussian Noise layer (e.g. set to 0.1) and then, three fully connected layers with 600 neurons and ReLU optimizer. Finally, the outputs of the 4 classes are decoded by a softmax classifier.

3.3 Experimental Study

This section provides the details of the FNC-1 and ARC datasets as well as the experimental setup for the proposed feature-based models and a feature-assisted neural model.

3.3.1 Dataset

For evaluating the proposed models on fake-news stance detection, we use two publicly available datasets and we discuss their details in the following subsections.

FNC-1

We evaluate the proposed approach on the FNC-1 dataset (Pomerleau and Rao, 2017) composed of textual documents for the task of Stance Detection. The dataset consists of 2,595 articles related to 300 claims derived from the Emergent project (Silverman, 2015) where each claim is possibly associated with 5 to 20 articles. The Emergent dataset is collected by Craig Silverman and his colleagues from different rumour sources including Snopes.com and twitter accounts for an online journalism project at Columbia University in 2015. It is related to various topics about technology, US and the world news.

Experts from journalism first determined the check-worthy claim and then searched for all the relevant articles after which they then summarized the relevant documents into headlines. These journalists manually labeled the claim/headline pairs as *for*, *against* or *observing*. They further labelled these instances into their veracity labels, *true*, *false* or *unverified*, with the aim to also establish this task as a fact-checking problem. This dataset which is also called an emergent dataset was first introduced by (Ferreira and Vlachos, 2016).

The FNC-1 challenge organizers released an extended version of stance detection dataset (Ferreira and Vlachos, 2016) which consists of 300 topics. Each document is matched with a summarized headline and the organizers labelled the pair with one of the four relative stances: *agree*, *disagree*, *discuss* and *unrelated*. The *unrelated* samples were

Table 3.3: The distribution of FNC-1 dataset

	ALL	**AGR**	**DSG**	**DSC**	**UNR**
Train	49,972	3,678	840	8,909	36,545
Test	25,413	1,903	697	4,464	18,349

created based on the random association of document and headline pairs that belonged to different topics.

As a result, there were 49972 instances (related to 200 topics) of headlines and news documents in the training-set whereas the test-set comprises 25,413 pairs from 100 topics. The distribution of the classes of headline/article pairs is highly imbalanced where the *unrelated* (UNR) pairs are approximately 73% while the other three classes share only the remaining 27% of the whole dataset (e.g. *agree* (AGR): 7.4%, *disagree*(DSG): 2% and *discuss* (DSC): 17.7%). Table 3.3 shows the distribution of the FNC-1 dataset.

ARC

To test the generalizability of the proposed model, we also use the Argument Reasoning Comprehension (ARC) dataset (Habernal et al., 2017) from the news domain, such as international affairs, immigration or schooling issues. The ARC (Habernal et al., 2017) dataset is constructed out of 188 debate topics that was created by crawling from the user debate section of the New York Times especially, the popular questions. For each topic, the authors extracted publicly available user posts (that were highly ranked by other users) on typical controversial topics after which they manually mapped each topic to two claims with opposing views. They then asked crowd-sourcing workers to generate labels to both kinds of claims as support or oppose and also a third label (neutral) when the claim does not have a stance.

This task is almost similar to the stance detection task (Pomerleau and Rao, 2017) but the difference is that one has shorter documents that expresses one viewpoint (Habernal et al., 2017) while the other has longer news documents with detailed perspective of an issue (Pomerleau and Rao, 2017). (Hanselowski et al., 2018) manually generated the

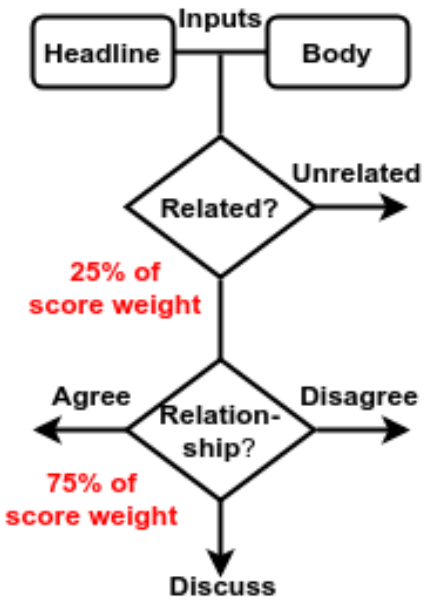

Figure 3.4: FNC-1 metric for the evaluation of systems in the competition

unrelated pairs by randomly mapping multi-sentence user posts to claims from different topics. This dataset, composed of 17,792 claims and multi-sentence user posts (e.g. *agree*: 8.9%, *disagree*: 10%, *discuss*: 6.1% and *unrelated*: 75%), is designed for stance detection by (Hanselowski et al., 2018). The dataset is divided 80/20 for training and test sets.

3.3.2 Metrics

The FNC-1 organizers introduced a mechanism to evaluate the performance of the models in the competition as shown on Figure 3.4. The evaluation metric weights the score of 25% for correctly classifying *related/unrelated* pairs and 75% for correctly classifying the *related* instances into further three-class classifications (e.g. *agree*, *disagree* and *discuss*). For evaluating the performance of the models, simply using the proposed weighted hierarchical evaluation metric by FNC-1 undermines the class-wise performance due to the unbalanced pairs in the dataset (Hanselowski et al., 2018; Bhatt et al., 2018). If a model predicts well on the majority classes (e.g. *unrelated* and *discuss* classes), the FNC-1 metric would produce a higher score. In this case, we use F1-score to evaluate the performance of our

Table 3.4: Baseline models for our study

Study	**Model**
(Pomerleau and Rao, 2017)	Gradient Boosting + lexical features
(Masood and Aker, 2018)	Multistage LR and random forest + lexical features
(Ghanem et al., 2018)	MLP or SVM + lexical features
(Hanselowski et al., 2017)	6-layer MLP + lexical features
(Riedel et al., 2017)	1-layer MLP + BoW and cosine features
(Thorne et al., 2017)	Esemble MLPs + lexical features
(Bhatt et al., 2018)	Esemble MLP + Neural and lexical features
(Baird et al., 2017)	Deep CNN and Gradient-boosting + lexical features
(Borges et al., 2019)	Bi-LSTM + External features
(Hanselowski et al., 2018)	StackLSTM + lexical features
(Mohtarami et al., 2018)	Memory network + BoW and cosine features
(Xu et al., 2019)	Two-level (1)MLP + BoW (2)CNN + domain adoptation

models based on a class-wise harmonic mean of precision and recall as well as the macro-averaged F1-score that averages the class-wise F1 scores. We also utilize an Accuracy (the proportion of all correctly classified classes compared to all samples) metric based on a per-class score and the unweighted average of all classes considering the imbalanced class distribution.

3.3.3 Baselines

We empirically evaluate our proposed models with the the following state-of-the-art baselines in Table 3.4 as we directly report the results from their publications.

3.3.4 Experimental Procedure

We estimate the best hyper-parameters using a grid search and we finally set the hyper-parameters of LightGBM as: learning rate - 0.09, number of leaves - 50, number of boosting rounds - 1000 and early stopping rounds - 50. The MLP model consists of the external features with three layers of 600 neurons and rectified linear unit (ReLu) activation function followed by a softmax classifier.

To improve the performance of feature-based models, feature-assisted dual GRU

Table 3.5: Comparison with the state-of-the-art traditional models

Models	UNR	AGR	DSG	DSC	**FNC-1**	**Accuracy**
Pomerleau and Rao (2017)	97.98	9.09	1.00	79.66	75.20	46.93
Masood and Aker (2018)	98.00	52.00	1.00	76.00	82.10	56.75
Ghanem et al. (2018)	–	–	–	–	–	58
This Work						
LightGBM	**99.13**	57.75	2.87	**80.00**	**83.40**	59.94
+Augmentation	98.88	**68.47**	4.16	71.42	82.27	60.73

with regularization layers has been employed as explained in the methodology sub-section. To evaluate the effectiveness of our text augmentation methods, we trained the same LightGBM, MLP and the dual GRU models as we add more training instances of 3678 and 840 for *agree* and *disagree* classes respectively.

The implementation of our models are based on Keras with Tenserflow as a backend and a LightGBM library. We use the Adam weight optimizer and we set epochs to 10 and batch size to 100 for the both of the GRU and the MLP experiments. We also utilize checkpoint and early-stopping to stop the training if the validation loss does not decrease for three consecutive epochs. 20% of the training-set is being used as a validation-set.

3.4 Results

In this section, we present the experimental results of the proposed models.

3.4.1 Feature-based LightGBM

Table 3.5 compares the results of our LightGBM model against the state-of-the-art traditional approaches with different hand-crafted feature for this task. We experimented with different lexical and similarity features representing the headlines and articles using LightGBM classifier. Our model performs better than the FNC-1's baseline Pomerleau and Rao (2017) and other state-of-the-art classical models for this task, with our LightGBM's performance increased by more than 1% in each of the evaluation metric as presented in

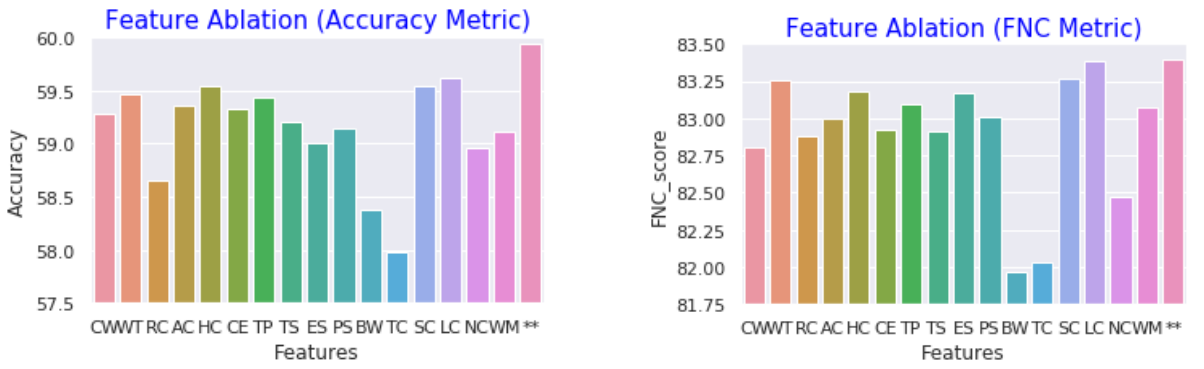

Figure 3.5: LightGBM ablation study (refer to Table 3.1 for the abbreviations)

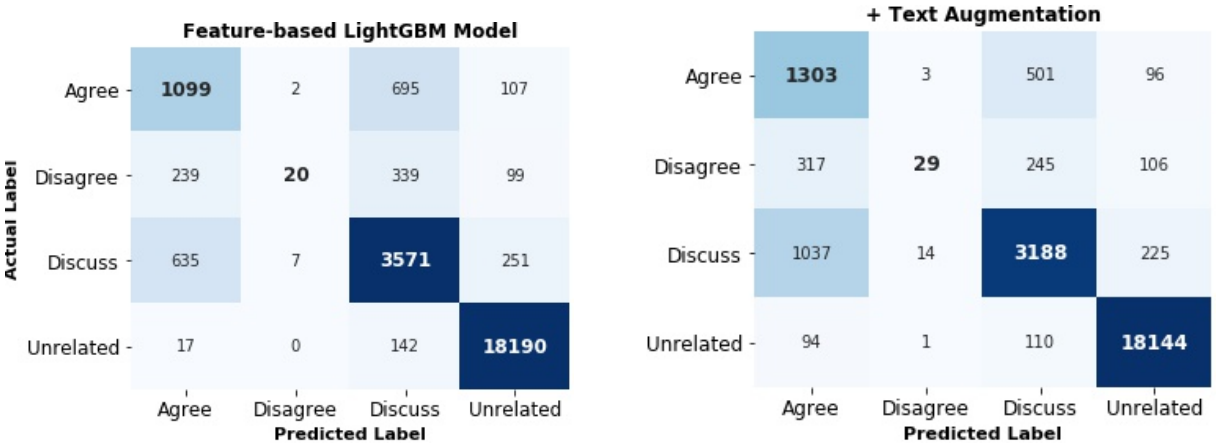

Figure 3.6: Feature-based LightGBM confusion matrices

Table 3.5. The LightGBM model with text augmentation effectively improves the class *agree* and the overall Accuracy.

Figure 3.5 reveals the ablation study of this model as we remove one feature-set (refer to Table 3.1) in each run of an experiment. We find that removing any of the feature-set from the model leads to a reduced performance in terms of FNC-1 and Accuracy scores. We also observe that leaving-out BW and TC features produces the worst FNC-1 and Accuracy scores of 81.97% and 57.98% respectively. The Figure 3.5 indicates that the least performance gains comes from LDA cosine similarity, SVD cosine similarity, hedging counts and overlapping features.

In addition to that, Figure 3.6 shows the confusion matrices of the LightGBM with and without text augmentation. On both matrices, *disagree* class is the most difficult class to predict, because our models are based on traditional approach which performs poorly

Table 3.6: Comparison with the state-of-the-art MLP models

Models	UNR	AGR	DSG	DSC	**FNC-1**	**Accuracy**
Riedel et al. (2017)	97.90	44.04	6.60	81.38	81.72	57.48
Hanselowski et al. (2017)	**99.25**	44.72	9.47	80.89	81.97	58.58
Thorne et al. (2017)	–	–	–	–	78.04	–
Bhatt et al. (2018)	98.04	43.82	6.31	**85.68**	**83.08**	58.46
Ghanem et al. (2018)	–	–	–	–	–	59.60
This Work						
MLP	98.95	53.13	12.20	79.61	**83.08**	60.97
+Augmentation	96.71	**60.64**	**21.52**	72.13	81.53	62.75

on a highly imbalanced datasets. On the other hand, the model with text augmentation significantly improved on the class *agree* while the predictions of class *discuss* is reduced to a lower number compared to model without augmentation. The reason for this is that the *agree* and *discuss* classes have very similar nature of textual content, however, creating another 7.4% new samples for class *agree* gives the advantage to increase the predictions of that class. We observe that the majority errors of *disagree* samples are misclassified as *agree* on the model with text augmentation and *discuss* on the other model.

3.4.2 Feature-based MLP

We also train a MLP model on the same lexical and similarity features. Variations of MLP approaches have been employed on this task with different combination of features as described in Table 3.4. The proposed MLP achieved 83.08% of FNC-1 and 60.97% of Accuracy scores as shown in Table 3.6 outperforming all previous MLPs (Riedel et al., 2017; Hanselowski et al., 2017; Thorne et al., 2017; Bhatt et al., 2018; Ghanem et al., 2018).

Moreover, MLP with augmentation model shows a substantial increase (more than 2% improvement compared to other methods) on Accuracy. Figure 3.7 shows the confusion matrices of our MLP with and without text augmentation.

The detection accuracy for *"disagree"* class is significantly improved by more than 15% compared to the feature-based LightGBM. We also obtain a competitive Accuracy

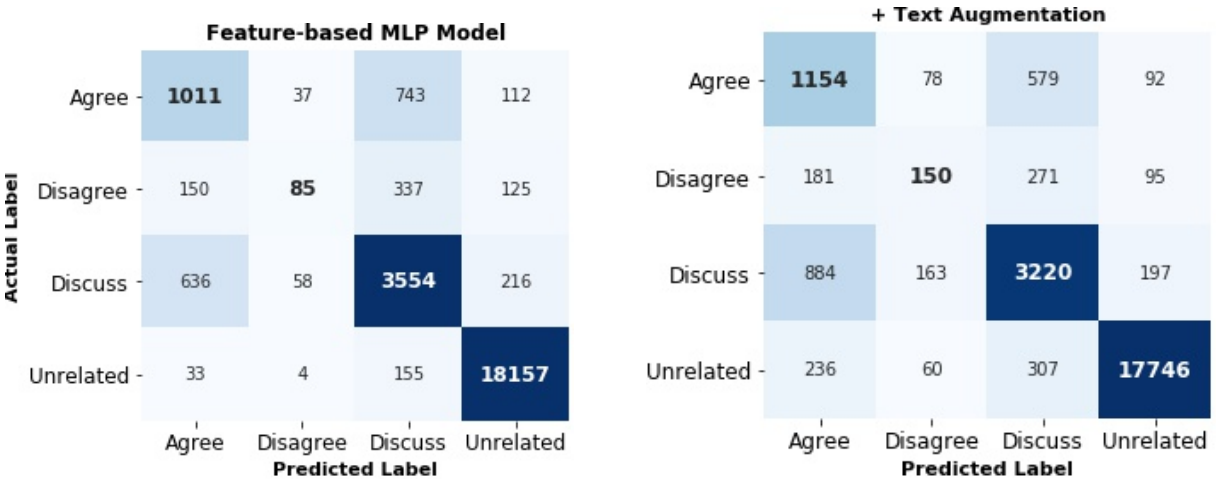

Figure 3.7: Feature-based MLP confusion matrices

score of 60% in *"agree"* class using this classifier with augmented samples. Besides, the improvement of these classes using feature-based MLP do not come at the expense of accuracy on other classes (*discuss* and *unrelated*), indicating that a feature-assisted neural model could be beneficial for this task.

3.4.3 Feature-assisted DL Model

Table 3.7 compares the evaluation results of the proposed model with the state-of-the-art neural models described in Table 3.4. We conduct experiments with a feature-assisted DL utilizing two different neural models separately such as LSTM and GRU with the purpose to inject semantic understanding and increase the detection of complex negation instances. However, the model with LSTM could not improve the results of *disagree* class compared to our feature-based MLP. Contrary to that, our feature-assisted GRU model is better able to capture more negation instances than all other models. We choose the model with the GRU layer because it can predict more on the *disagree* class which is the most difficult category to detect due to its unbalanced number of samples compared to other classes. GRU-MLP Baseline performs the worst on this dataset without using any of the external features, batch-normalization and Gaussian Noise layers. GRU-MLP-External baseline performs comparably good in terms of FNC-1 (82.88%) and Accuracy (59.59%)

Table 3.7: Comparison with the state-of-the-art Feature-assisted Neural models

Models	UNR	AGR	DSG	DSC	**FNC-1**	**Accuracy**
State-of-the-art						
Baird et al. (2017)	**98.70**	58.50	1.86	76.18	82.02	58.81
Borges et al. (2019)	96.74	51.34	10.33	**81.52**	82.23	59.98
This Work - LSTM						
LSTM-MLP-External-BN-GN	97.75	52.60	11.91	79.32	82.66	60.40
This Work - GRU						
GRU-MLP	86.94	26.54	0.86	49.96	60.37	41.07
GRU-MLP-External	98.98	56.28	4.30	78.79	**82.88**	59.59
GRU-MLP-External-BN-GN	98.48	60.43	15.64	74.33	82.36	**62.22**
+Augmentation	96.62	**66.47**	23.39	65.32	80.26	**62.95**

metrics but it does not improve the results for *disagree* and *agree* classes. The proposed ensemble GRU-MLP-External-BN-GN model with 50-d GloVe shows higher Accuracy score for class-wise and overall performance on stance detection setting.

However, when text augmentation is added with the proposed model, it demonstrates the best results on *agree* and *disagree* scores of 66.47% (from 58.50% (Baird et al., 2017)) and 23.39% (from 10.33% (Borges et al., 2019)) as it performs comparably well to previous approaches in other two classes. Table 4.7 shows that a weighted ensemble model achieves higher overall accuracy, 63.98% on the Accuracy metric, compared to the previous models' highest Accuracy score 59.98% (Borges et al., 2019). This shows that the simple ensemble GRU model optimized with text augmentation methods and regularization layers is robust in terms of class-wise accuracy compared to other complex ensemble models (Baird et al., 2017; Borges et al., 2019) and the overall Accuracy metric was improved by more than 3% from 59.98% to 63.98%.

3.4.4 Embeddings

As pre-trained word embedding capture the semantic representations among words, we observe that it has an important contribution to the performance of the proposed model. Therefore, we test four GloVe pre-trained vectors (Pennington et al., 2014) as well as Word2Vec (Mikolov et al., 2013) in order to choose which embedding model is best suited

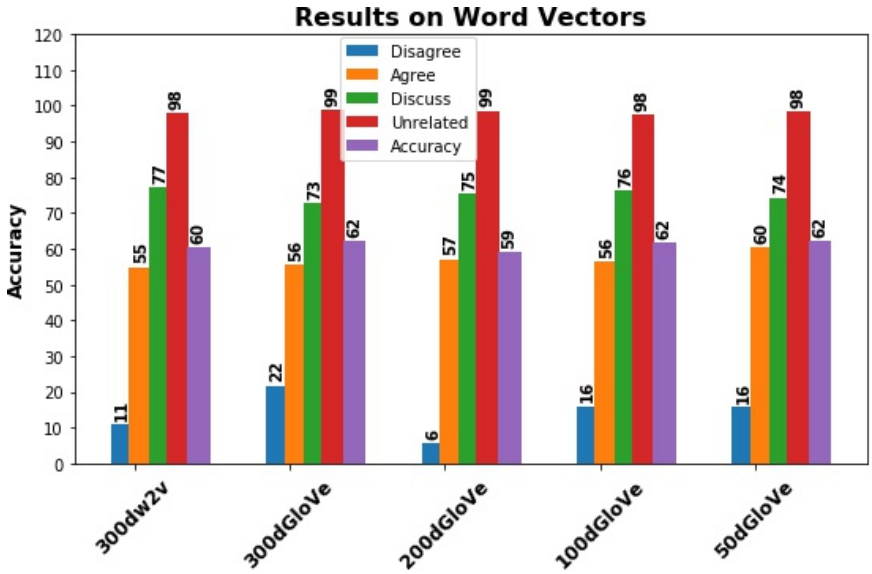

Figure 3.8: Performance comparison on word vectors

for our task. Figure 3.8 shows the performance of the different pre-trained word embeddings with GRU-MLP-External-BN-GN model as described in Table 3.1 and Figure 3.3. We can see that, among the different pre-trained models, the 50d, 100d and 300d GloVe vectors yield the best results. With respect to 300d-Glove and 200d-Glove, the training for each epoch takes more time to be completed compared to the 50d-Glove which does not require large computation due to its reduced dimension. Moreover, the results also indicate that the 50d-GloVe model performs better on *agree* and *disagree* classes but the difference is not that significant in terms of overall accuracy. Following the initial evaluation of these experiments, we choose 50-d GloVe (Pennington et al., 2014) for use in most of the experiments described in this chapter.

3.4.5 Cross-domain Validation on FNC-1 and ARC Datasets

Table 3.8 shows the performance of the feature-assisted neural model (without data augmentation) compared against the state-of-the-art models when using Macro-F1 evaluation metric on FNC-1 and ARC datasets. To put the numbers into prospective, we also show the cross-domain validation results of our model and the Hanselowski et al. (2018) system. We can see that the proposed model consistently outperforms prior three systems

Table 3.8: Cross-domain evaluation using F1-score on FNC-1 and ARC datasets

Models	UNR	AGR	DSG	DSC	**FNC-1**	**Macro-F1**
FNC-1-FNC-1						
Mohtarami et al. (2018)	–	–	–	–	81.23	56.88
Xu et al. (2019)	97.70	54.60	15.10	72.60	80.30	60.00
Hanselowski et al. (2018)	**99.50**	50.10	18.00	**75.70**	82.10	60.90
GRU-MLP-External-BN-GN	97.94	**54.92**	**24.01**	75.17	**82.36**	**63.01**
ARC-ARC$_1$						
Hanselowski et al. (2018)	**93.50**	45.10	51.80	19.4	68.50	52.40
GRU-MLP-External-BN-GN	93.48	**49.53**	**52.91**	**25.99**	**69.95**	**55.48**
ARC-FNC-1						
Hanselowski et al. (2018)	**95.00**	34.30	11.60	8.20	61.30	37.30
GRU-MLP-External-BN-GN	96.12	**39.63**	**11.61**	**15.85**	**63.94**	**40.80**
FNC-1-ARC						
Hanselowski et al. (2018)	91.00	**32.10**	**19.10**	18.20	**59.10**	**40.10**
GRU-MLP-External-BN-GN	**89.81**	27.41	8.93	**16.97**	56.05	35.78

(Hanselowski et al., 2018; Mohtarami et al., 2018; Xu et al., 2019) on *agree*, *disagree*, FNC-1 and the overall F1-score measures. This means that the hand-crafted features, dual-GRUs and the regularization layers have an important contribution to the the overall effectiveness of the classifier given the difficulty of the stance detection task. We also test the feature-assisted GRU model on the ARC dataset to determine its generalizability. Based on initial experimentation for cross-validation, 300-d-GloVe (Pennington et al., 2014) is chosen to generate sequences of word embeddings of a claim and multi-sentence document pair. Again, Table 3.8 presents the performance of the cross-domain evaluations. For in-domain ARC-ARC training and test scenario, our model gives a slight improvement on both of the evaluation metrics over the stackLSTM model (Hanselowski et al., 2018). For the cross-domain test, we find that the proposed model trained on the ARC training-set and tested on the FNC-1 test-set outperforms on most of the classes and the overall FNC-1 as well as F1 metrics compared to the stackLSTM model in (Hanselowski et al., 2018). We also observe that the cross-domain FNC-1-ARC test performance on our proposed approach is lower than the stackLSTM in (Hanselowski et al., 2018) and it can be improved by using text augmentation as we have shown on FNC-1 dataset.

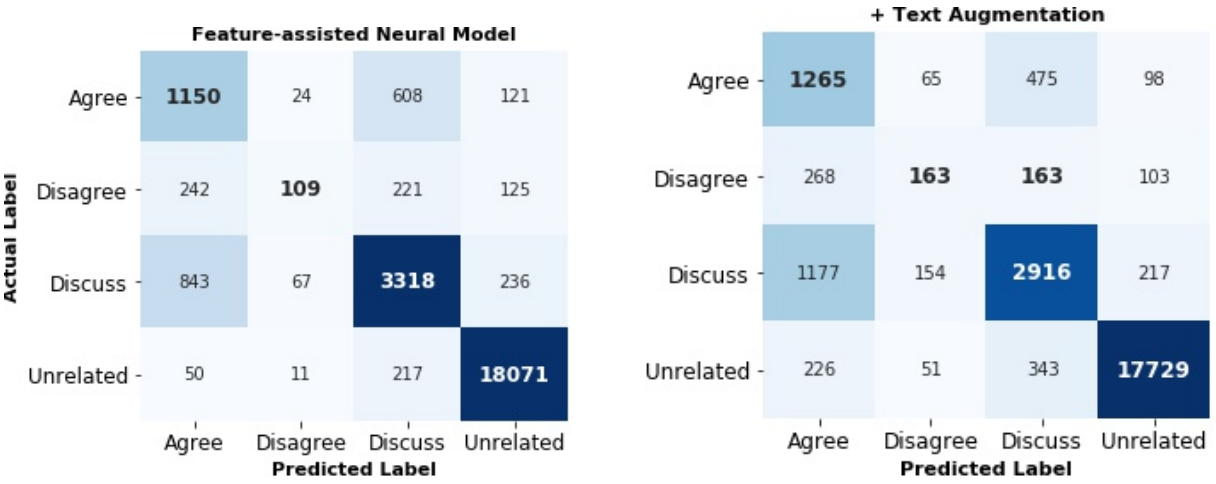

Figure 3.9: Feature-assisted GRU confusion matrices

3.5 Discussion

Our proposed LightGBM with all the external features achieves the highest FNC-1 weighted accuracy 83.40%. LightGBM with text augmentation model also performs the best in terms of *agree* class (68.47%). However, we observe that the LightGBM models lack the semantic understanding required in improving the detection of class *disagree* as the other traditional models have difficulties making better predictions due to the imbalance of training data. Furthermore, our experiments show the importance of the neural models by firstly employing the feature-based MLP. It is interesting to see that our MLP variations with all the hand-crafted features outperform with a significant margin compared against the state-of-the-art MLPs (Riedel et al., 2017; Hanselowski et al., 2017; Thorne et al., 2017; Bhatt et al., 2018; Ghanem et al., 2018). This proves that the lexical and similarity features as well as text augmentation have an important contribution towards stance detection. Feature-assisted GRU model have the strength and the semantic-understanding ability to predict and improve class-wise and overall F1 score by a significant margin. Our experiments also demonstrate that text augmentation and regularization methods such as batch-normalization and Gaussian noise are useful methods that can help prevent overfitting to overcome the class imbalance problem. We have seen that it is possible to outperform the state-of-the-art results with these simple models

compared with complex DL ensembles (Borges et al., 2019; Hanselowski et al., 2018; Mohtarami et al., 2018; Xu et al., 2019). Our simple feature-assisted GRU model improves on the current state-of-the-art by over 3% on F1 metric as illustrated in Table 4.8. The proposed GRU-MLP-External-BN-GN is also applied to ARC dataset for cross-domain validation and it has shown better performance compared to previous stack-LSTM model (Hanselowski et al., 2018).

As illustrated in all the above confusion matrices (Figures 3.6 and 3.7), we observe that the *unrelated* category is the most frequent to predict, with most of the training-set belonging to it and the prediction of *disagree* class is the lowest because of that category is underrepresented. All models can separate *unrelated* from *disagree* with fewer errors but they largely misclassify the *discuss* category as *agree* and vice versa.

According to Figure 3.9, the prediction of *agree* and *disagree* classes is improved slightly more than the feature-based models. We also maintained a good performance on the *unrelated* and *discuss* classes. Although the performance of underrepresented categories is improved compared to the state-of-the-art, the class imbalance problem can still influence the multiclass classification approaches towards predicting more on the majority represented classes. The models with text augmentation may have good performance compared to those without text augmentation, but the detection of *agree* and *disagree* stances is important for fact-checking (e.g. distinguishing legitimate from false claims). Therefore, we still need to reduce the bigger number of misclassified instances for these categories. We conclude that the performance of underrepresented classes can be improved through multistage learning strategy where the multiclass classification problem can be modeled using hierarchical classifiers separately.

3.6 Conclusion

In this chapter, we presented a simple combined model of DL with hand-crafted features for automating the stance detection on news headline and body pairs. We first generated

different groups of classical features from headlines and bodies. We show that LightGBM and MLP models with these hand-engineered features provide state-of-the-art results on this task. We then integrate the external features and a simple dual GRU neural network with 50-d GloVe pre-trained embedding to boost the semantic understanding of the model. This combined model is optimized with text augmentation, Batch Normalization and Gaussian Noise regularization methods to provide significant improvement compared to the state-of-the-art on FNC-1 dataset. The next chapter extends the proposed stance detection models in Chapter 3 by breaking down the multiclass classification problem into multistage classification hierarchies in order to improve the current state-of-the-art results.

CHAPTER 4

MULTISTAGE NEWS-STANCE CLASSIFICATION

In this chapter, we discuss an improved approach to tackle the stance detection problem by proposing two multistage classification approaches using a feature-based traditional model and feature-assisted neural models. The chapter is structured as follows. Section 4.1 presents the introduction of the stance detection problem. In Section 4.2, we describe the details of our methodology. Section 4.3 and 4.4 elaborates the experimental settings and the results of the proposed multistage settings respectively as we finally draw some discussion of our work and the concluding remarks in Section 4.5 and 4.6.

4.1 Introduction

This chapter focuses on extending multiclass stance detection which determines the stance of an article to a headline as *agree*, *disagree*, *discuss* or *unrelated*. Although the existing literature have investigated different solutions for the class imbalance problem, one potential solution was to generate new samples for minority classes using data augmentation

techniques as we have explored in Chapter 3.

As most popular NLP tasks, stance detection also suffers severe class imbalance problem due to the unbalanced number of *unrelated* and *related* (*agree*, *disagree* and *discuss*) samples. While multiclass classification approaches (Bhatt et al., 2018; Borges et al., 2019; Hanselowski et al., 2018; Saikh et al., 2019) have shown to be effective in predicting *unrelated* samples, they struggle in the related sample classification where the class imbalance is assumed to be learnable from the training data. Particularly, we assume that misclassification of *agreement* pairs as either *unrelated* or *discuss* classes can severely affect the overall verdict whereas the detection of *agree* and *disagree* stances is more important towards identifying legitimate and false claims. However, we relax those assumptions by proposing multistage classification approaches. Indeed, multistage approaches for sentence classification have proven to handle the imbalanced multiclass classification problems in other NLP tasks such as sentiment analysis (Mukherjee et al., 2012), twitter-stance detection (Dey et al., 2018) and offensive language detection (Park and Fung, 2017).

The contributions of this chapter include two multistage classification approaches designed to merge the benefits of lexical and neural features using DL classifiers. The first approach is based on a two-stage strategy, where in the first-stage a relevance classifier is trained to pre-classify our task into *related* and *unrelated* categories, and in the second-stage the related samples are further classified into the other three-classes (*agree*, *disagree* and *discuss*) using a stance classifier on the FNC-1 dataset. We also present a three-stage classification approach in which we assume that the prediction of *agreement* samples can be significantly increased. Firstly, we adopted the relevance classifier from the two-stage strategy. Secondly, we further split the related samples into two binary classifiers: stance and agreement classification. We leverage embedding techniques (e.g. word and sentence) such as GloVe and universal sentence encoder to capture the deeper semantics of the text. We also make use of lexical features that proven to convey additional relevant information and assist neural models in order to achieve solid performance.

4.2 Methodology

In this section, we introduce our multistage classification approaches to news-stance detection using lexical and neural based text representation. We first present the lexical and neural methods for stance detection. The multistage classification models will be discussed in the next subsection.

4.2.1 Lexical Features

Lexical features capture the presence of a word in a text and lexical overlaps between the headline and the news body. They also provide more understandable results with regards to each feature's contribution which can easily be quantified about its performance as shown in Chapter 3. We conduct a feature ablation study for each and every sub-classification model in our multistage setting to extract the best lexical features by using LightGBM classifier.

Based on our experimental analysis, the following lexical features are deemed important from the domain of news-stance detection (Baird et al., 2017; Hanselowski et al., 2017) and rumour-stance detection (Ghanem et al., 2019). We first transform our text (e.g. title and body) into TF-IDF vectors and then we apply LDA, LSI and NMF to generate **Topic** vectors of 100, 100 and 50 dimensions respectively (Hanselowski et al., 2017, 2018). For each of the above **Topic** models and the TF-IDF (**BoW**), we compute the cosine distance between the title and body vectors. Unlike other studies (Hanselowski et al., 2017, 2018) that use headline and article vectors as features, we only pass the **Topic** similarities to our multistage classification settings one way or another. We extract 2000 most frequent 3-grams TF vectors (**BoW**) from the title and body of the article and we also calculate the cosine distance between them (Riedel et al., 2017). Five **Word-count** features are also being extracted from the title and body text using refuting (Pomerleau and Rao, 2017) and discuss (Ferreira and Vlachos, 2016) lexicons as well as implicative, positive and negation lexicons (Ghanem et al., 2019). In addition, we use overlapping

(**Overlap**) character and word ngrams (Pomerleau and Rao, 2017) in the headline and body text as well as common entities (**Overlap**) between the pairs of the text. We further incorporate subjectivity (using textblob - **Polarity**), emotional valence (**Polarity**), VADER sentiment (**Polarity**), MPQA sentiment (**Polarity**) scores computed separately for each pair of the text. Finally, we extract the cosine (**Embedding**) and Word Mover's Distance (WMD) similarities between the headline and body vectors based on Word2Vec embeddings. In each stage of both multistage classification models, we use a combination of lexical-overlap features as presented in Table 4.1 and 4.2.

4.2.2 Neural Features

In our work, we consider GloVe word embeddings (Pennington et al., 2014) and feature-based pre-trained sentence embeddings based on Universal Sentence Encoder (USE) (Cer et al., 2018) to transform the input news documents and headlines into sentence vectors.

Global Vectors for Word Representation(GloVe) (Pennington et al., 2014) is one of the word embeddings used to represent input vectors for DL models. In this work, we adopted a pre-trained 50-dimensional GloVe vectors to obtain a vector representation of each word in the article and average all the vectors.

Universal Sentence Encoder (USE) (Cer et al., 2018) In this study, we compute title and body sequences using the transformer encoder to obtain 512-dimensional vectors for headlines and news bodies.

4.2.3 Multistage Classification Approaches

Two approaches of multistage classifications are put forward as they utilize multiple classifiers that determine either binary or multiclass classification task. The multistage hierarchies of both models are shown in Figure 4.1.

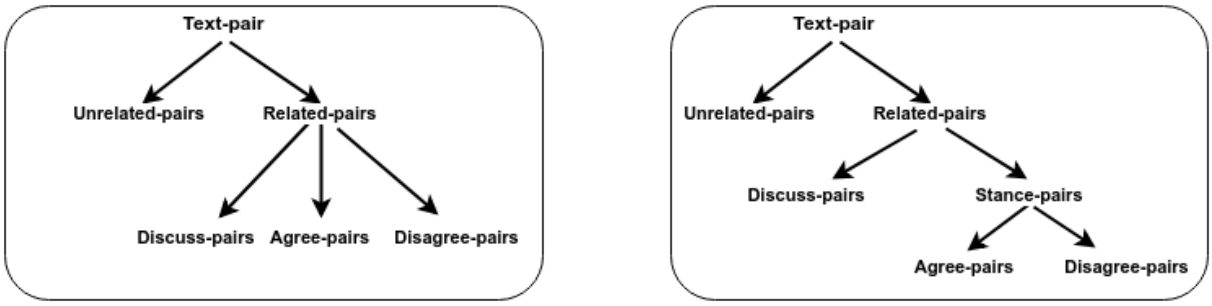

Figure 4.1: Hierarchies for Two-stage and Three-stage models

Two-stage classification approach

This approach has two stages of prediction.

A relevance classifier is trained in the first-stage to determine if the news document is related or not to a target headline as shown in Figure 4.2. State-of-the-art approaches have already achieved better classification accuracy on *related* vs *unrelated* with above 97%. The underlying assumption is that *unrelated/related* classification can often be determined easily and is less relevant for detecting fake news compared to the further classification of related pairs into stance classes. However, it is still important to correctly prune away irrelevant pairs which have a high number of samples compared to related pairs amid resulting class imbalance problems. Therefore, we hypothesize that a traditional classifier with keyword overlap and similarity based features between the headline and the article works well in predicting their relatedness. The first stage, therefore, applies LightGBM classifier with lexical-overlap features that are shown in the first part of Table 4.1.

Since we now have eliminated *unrelated* pairs, we build a second-stage stance classifier that is able to distinguish the stance classes. Stance classification is the process of discovering whether an article content expresses a stance (e.g. *agree*, *disagree* and *discuss*) towards a news headline. In this stage, we receive a headline and a set of relevant articles from the previous stage. Taking inspiration from our methodology in Chapter 3, a feature-assisted neural model is proposed for this stage. We present a dual Gated Recurrent Unit (GRU) with Global Max Pooling model as explained in the model settings, which can be a good solution for sequential data. As explained in Section 4.2.2, we adopt

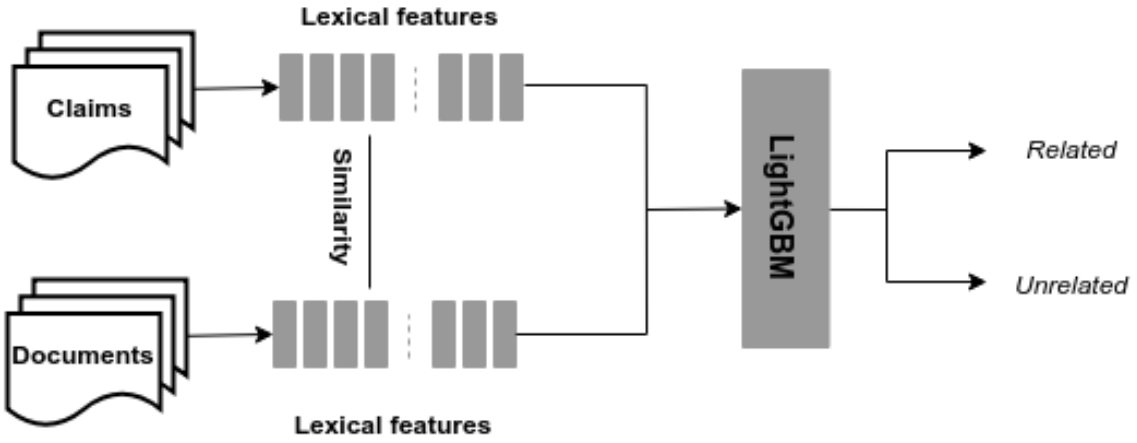

Figure 4.2: Feature-based LightGBM model for 1st Stage

Table 4.1: Set of features used in the Two-stage classification approach

Features for Relevance Classifier - 1st Stage
overlapping character and word ngrams between headline and body text
cosine similarity feature between headline and body 3000 BoW vectors
cosine similarity feature between headline and body 3-grams TF-IDF vectors
cosine similarity feature between 100 headline and body SVD components
cosine similarity feature between 100 headline and body LDA topics
cosine similarity feature between 50 headline and body NMF topics
Features for Stance Classifier - 2nd Stage
refuting, discuss, positive and implicative word counts of headline and body text
textblob's subjectivity in the headlines and the top 25 TF-IDF body vectors
emotional valence, vader sentiment, MPQA sentiment scores in each of the text
common entities between headline and body text
2000 most common 3-gram TF vectors from the headline and body text
cosine similarity feature between headline and body 3-grams TF-IDF vectors
cosine similarity feature between 100 headline and body SVD components
cosine similarity feature between 50 headline and body NMF topic vectors
WMD similarity between the headline and body word-embedding vectors
cosine similarity between the headline and body word-embedding vectors

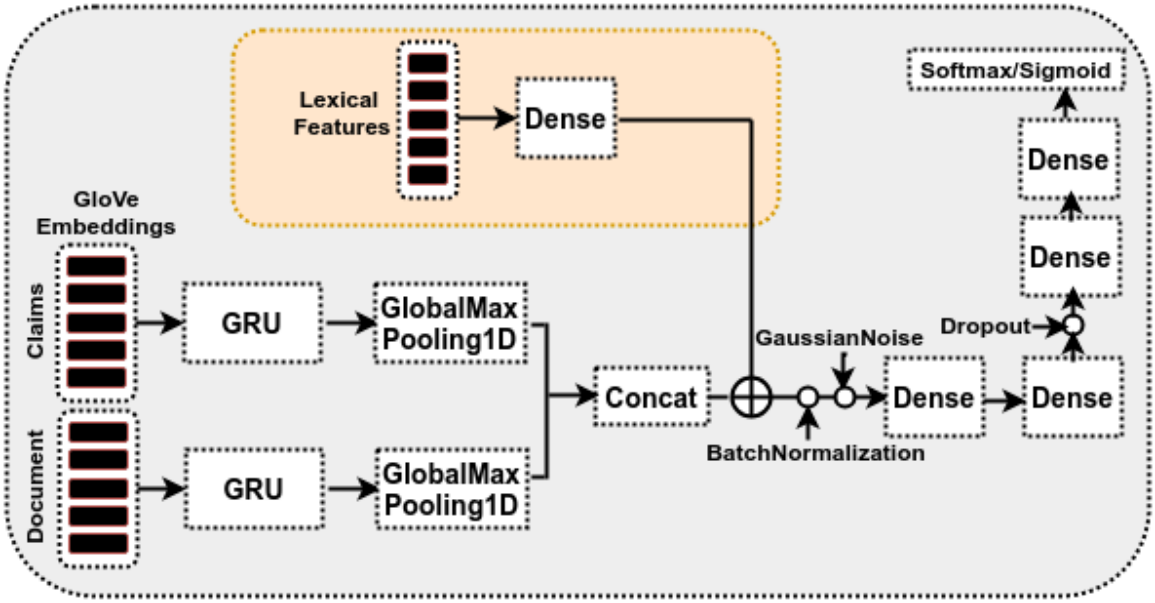

Figure 4.3: Feature-assisted GRU model encoded with GloVe vectors for 2nd Stage

word embeddings for this stage which encodes a limited length of text in order to generate a simple average of their corresponding word vectors as an input to our neural model. Although the neural model alone is supposed to overcome the semantic understanding limitation of traditional techniques, we observe that it does not generalize well due to the smaller size of the dataset. In order to improve the generalization capability of our model, we incorporate the lexical-overlap features (as shown in the second part of Table 4.1) in the representation layer obtained from Global Max Pooling layers. Table 4.3 provides the hyperparameter settings of two-stage neural model.

Three-stage classification approach:

We evaluate the multiclass classification at three levels with binary classifiers – *related/unrelated* level, *discuss/agreement* level and *agree/disagree* level.

The first-stage related classifier of the two-stage approach is adopted for the same purpose.

Similar to the second-stage stance classifier of two-stage approaches, a feature-assisted DL model encoded with GloVe embeddings is built from related pairs to further predict *discuss* and *agreement* classes. We make modifications to the DL hyperpameters

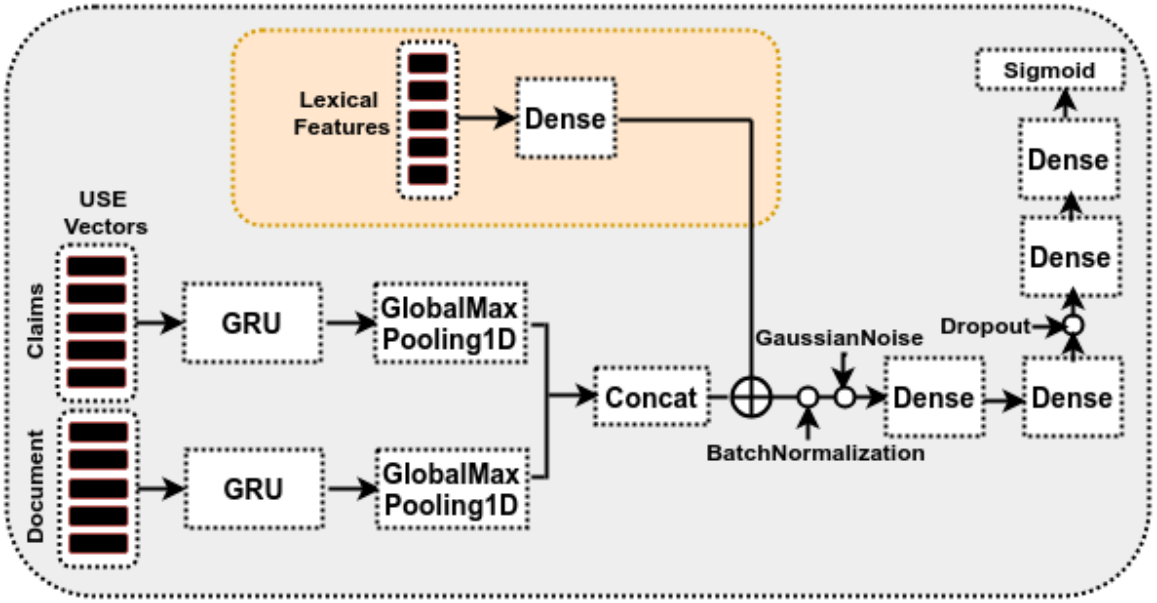

Figure 4.4: Feature-assisted GRU model encoded with USE vectors for 3rd Stage

(shown in Table 4.3) and lexical features (illustrated in Table 4.2) due to the hyperparameter optimization and feature ablation for this particular binary classification problem.

After obtaining *agreement* predictions, the third-stage agreement classifier aims to determine whether document/claim pairs *agree* or *disagree* with each other. Given the number of document/claim instances is small, we hypothesize that a feature-assisted neural model (as illustrated in Figure 4.4) encoded with sentence representations generated from USE vectors can be effective. As explained in Section 4.2.2, sentence representation models are able to capture more textual context than word embeddings. They can be used as pre-trained embeddings to represent the order of words within a long sequence of text and they have the potential to improve the performance of different NLP tasks. In summary, a hybrid model with lexical-overlap features and a DL layer encoded with USE vectors, instead of GloVe, is built for classifying the *agreement* into *agree* and *disagree* categories. The details of the DL models are presented in the coming sub-section and Table 4.3.

Table 4.2: Set of features used in the Three-stage classification approach

Features for Relevance Classifier - 1st Stage
The first stage features are presented in the first part of Table 4.1
Features for Stance Classifier - 2nd Stage
refuting, discuss, positive and negative word counts of headline and body text
emotional valence and MPQA sentiment scores in each of the text
2000 most common 3-gram TF vectors from the headline and body text
cosine similarity feature between headline and body 3-grams TF-IDF vectors
cosine similarity feature between 100 headline and body SVD components
Features for Agreement Classifier - 3rd Stage
refuting and negative word counts of headline and body text
textblob's subjectivity in the headlines and the top 25 TF-IDF body vectors
cosine similarity feature between headline and body 3-grams TF-IDF vectors
cosine similarity feature between 50 headline and body NMF topic vectors
WMD similarity between the headline and body word-embedding vectors

Table 4.3: Hyperparameters for the study (II:Two-stage - III:Three-stage)

Hyperparameters for our DL Models			
Parameters	**2nd Stage for II**	**2nd Stage for III**	**3rd Stage for III**
Embedding	GloVe (50d)	GloVe (50d)	USE (512d)
GRU	64	128	64
GRU-dropouts	D(0.15) RD(0.2)	D(0.2) RD(0.22)	D(0.1) RD(0.2)
Gaussian-noise	0.1	0.05	0.1

4.3 Experiments

In order to evaluate the performance of the proposed models, we provide the details of the multistage experimental setup and the model baselines as explained in this section.

4.3.1 Multistage Model Settings:

We build a feature-assisted multistage classifier by merging lexical-overlap features and DL features to improve the performance of stance detection. We optimize the hyperparameters that produce the optimal results for both of our multistage models based on the validation data. In order to evaluate the contribution of our lexical features in each stage, we employ LightGBM classifier (learning rate: 0.09 - 1.5, number of leaves: 50 - 90, number of boosting rounds: 1000 and early stopping rounds: 30) to understand

which hand-crafted feature is suited which stage in the classification. Table 4.1 and 4.2 present lexical-overlap features used in each stage of our proposed models. With the imbalanced nature of the dataset, lexical-overlap features with LightGBM demonstrate good performance on majority classes.

To further improve the performance, we build a hybrid model by combining DL with lexical features. The model starts by building a GRU (Gated Recurrent Unit) layer encoded with GloVe vectors in the second stage and the USE vectors in the third stage where the GRU dropout (D) and recurrent dropout (RD) values are set to their respective values shown in Table 4.1. Each vector (title or body) is assigned to a separate GRU layer followed by a global-max-pooling layer, which is used to minimize the dimensionality and capture the most important feature. We then perform a concatenation operation to combine the outputs from each pooling layer as well as the lexical-overlap features. We then use Batch-normalization and Gaussian noise regularization layer to prevent overfitting and optimize the neural network performance. Finally, we fed through four fully connected (FC) layers with 512 neurons and ReLU optimizers. There is also a dropout layer (0.15) in between the second and the third FC layers. Sigmoid or softmax is used as an output layer to produce the final predictions of the stance classes for the two-stage or three-stage DL experiments.

Table 4.3 shows the hyperparameter values employed in each stage of our proposed DL architectures with the exception of first-stage (e.g. LightGBM is employed for this stage). For our DL models, we use Adam optimizer with either categorical or binary cross-entropy as loss function for multiclass or binary-class settings respectively. The training epochs for these models are approximately 20 with a mini-batch size of 64 and we also used checkpoint and early-stopping to stop the training if the accuracy on the validation-set does not increase for three consecutive iterations.

Table 4.4: Baseline models for our study

Study	**Model**
Saikh et al. (2019)	USE + MLP + Text entailment features
This study (Chapter 3)	GRU + MLP + Augmentation + Lexical overlap features
Zhang et al. (2019b)	Two layer neural network - independent
Zhang et al. (2019b)	Two layer neural network - dependent
Jwa et al. (2019)	Fine-tuning BERT approach

4.3.2 Baselines

We consider the following state-of-the-art baselines and we report the results obtained from their research papers. We've described some of the baselines in Chapter 3 and we give a brief description about the new baselines.

4.3.3 Data and Evaluation Metrics

We use a stance detection benchmark dataset (Pomerleau and Rao, 2017) provided by the Fake News Challenge (FNC-1) for our model's training and testing. The dataset consists of 49,972 training pairs of headlines and news documents as well as 25,415 pairs of test-set which have close distribution of four stance labels: *agree (7.36%)*, *disagree (1.68%)*, *discuss (17.57%)* and *unrelated (73.13%)*. Therefore, the distribution of the stance labels is highly imbalanced. As a result, we divided the dataset into stage-wise class distribution as shown in Table 4.5 and 4.6. *Related (RTL)* category represents the class distribution of all labels except the *unrelated*. Also, the data from *agree* and *disagree* classes are merged into one category called *Stance (STC)*.

Moreover, we further divided the training instances into 90% training and 10% validation during sub-stage model training. Similar to our previous work (refer to Chapter 3), we use per-class and overall *Accuracy* and *F1-score* metrics to evaluate the performance of our models.

Table 4.5: FNC-1 dataset for Two-stage classification - (Related:RLT)

		1st-stage		2nd-stage		
	ALL	**UNR**	**RLT**	**AGR**	**DSG**	**DCS**
Train	49,972	36,545	13,427	3,678	840	8,909
Test	25,413	18,349	7,064	1,903	697	4,464

Table 4.6: FNC-1 dataset for Three-stage classification - (Stance:STC)

		1st-stage		2nd-stage		3rd-stage	
	ALL	**UNR**	**RLT**	**DCS**	**STC**	**AGR**	**DSG**
Train	49,972	36,545	13,427	8,909	4,518	3,678	840
Test	25,413	18,349	7,064	4,464	2,600	1,903	697

4.4 Results

We compare our multistage classification models with the state-of-the-art approaches on the FNC-1 dataset. For fair comparison, we compute the overall and per-class performances using Accuracy and F1 metrics as shown in Table 4.7 and Table 4.8 respectively. As shown in Table 4.7, our two-stage approach achieves an Accuracy of 66.75%, close to a 3-point improvement over the prior one-stage feature-assisted DL models such as ours in Chapter 3 and in (Saikh et al., 2019). It indicates that, simply dividing the task into relevance and stance classifiers can filter-out the majority class (e.g. *unrelated*) in the first-stage and improve the predictions of the related pairs (e.g. *agree* 72.10%, *disagree* 32.28%) in the second-stage. Also, our three-stage model effectively improves the results for *disagree* (44.76%) class and the overall Accuracy (69.51%) compared to our two-stage, independent and dependant two-layer DL framework (Zhang et al., 2019b).

Table 4.7: Comparison with the state-of-the-art on Accuracy metric

Models	*Unrelated*	*Agree*	*Disagree*	*Discuss*	**Accuracy**
State-of-the-art models					
Saikh et al. (2019)	97.17	61.06	21.38	**74.44**	63.51
This study (Chapter 3)	97.22	65.32	24.53	68.86	63.98
Zhang et al. (2019b)	**99.05**	61.34	42.93	59.38	65.68
Zhang et al. (2019b)	97.43	**72.41**	37.90	68.23	68.99
This work					
Two-stage	98.19	**72.10**	32.28	64.45	66.75
Three-stage	98.19	71.62	**44.76**	63.49	**69.51**

Table 4.8: Comparison with the state-of-the-art on F1 score

Models	*Unrelated*	*Agree*	*Disagree*	*Discuss*	**F1**
State-of-the-art models					
Baird et al. (2017)	99.40	53.90	3.50	76.00	58.20
Hanselowski et al. (2017)	**99.60**	48.70	15.10	78.00	60.40
Riedel et al. (2017)	98.90	47.90	11.40	74.70	58.30
Hanselowski et al. (2018)	99.50	50.10	18.00	75.70	60.90
Jwa et al. (2019)	98.90	**65.10**	14.50	**83.90**	65.60
This work					
Two-stage	98.31	56.44	**34.62**	71.82	**65.30**
Three-stage	98.31	57.68	**42.68**	70.98	**67.41**

The results from Table 4.8 indicate that our two-stage model has outperformed on F1-score compared with one-stage feature-assisted (Baird et al., 2017; Hanselowski et al., 2018, 2017; Riedel et al., 2017) models by simply adapting multistage learning strategy with the best possible lexical and DL features. Note that this approach also performs comparably close to the BERT model (Jwa et al., 2019). In the case of our three-stage model, it indicates the importance of using lexical and DL features in a multistage setting by achieving higher F1-score than the state-of-the-art models (Baird et al., 2017; Hanselowski et al., 2018, 2017; Jwa et al., 2019; Riedel et al., 2017). Even with this simple feature-assisted DL models, we achieve higher Accuracy and F1-score when compared with the state-of-the-art models mainly because of our assumption based on the multistage adoption.

4.4.1 Model Ablation Study

In the case of the model components, we conduct ablation study regarding the influence of lexical-overlap and DL features in a multistage setting. Figure 4.5 shows that two-stage LightGBM model highest reaches 62.05% Accuracy, 2% lower than our three-stage LightGBM. We also observe that three-stage LightGBM outperforms feature-assisted DL models present in Chapter 3 and in (Saikh et al., 2019), which proves the effectiveness of the lexical-overlap features and the multistage setting. In the case of two-stage Hybrid-DL, we find that DL model alone in the second-stage does not obtain better results

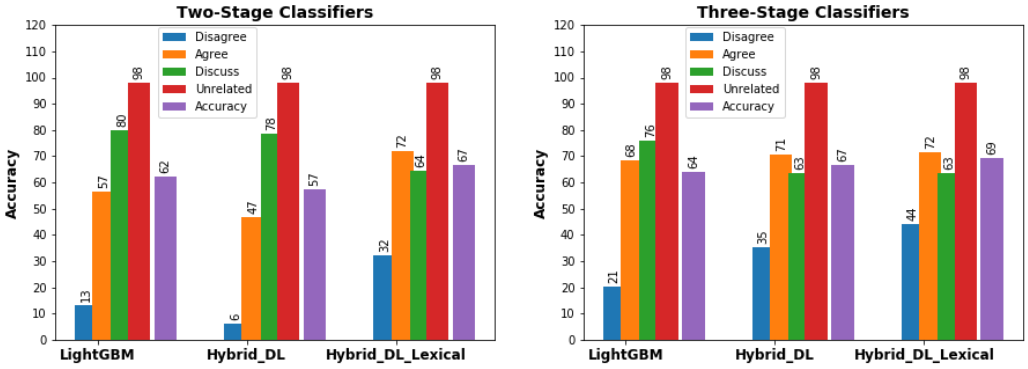

Figure 4.5: Ablation study for proposed models

but the performance of the model improved when combined with lexical overlap. On the other hand, using a DL model with only USE vectors in the third-level of three-stage approach obtains better performance. When we add lexical-overlap, this model outperforms the state-of-the-art which shows the effectiveness of multistage compared to one-stage classification approaches.

4.4.2 Feature Ablation Study

In our feature-based LightGBM, we conduct feature ablation study in each stage separately by excluding one type of a feature to record the performance of the entire model whether it is two-stage or three-stage classification experiments. The results of both approaches regarding the feature ablation set of experiments are shown in Figure 4.6 and 4.7 respectively. We provide our findings based on the overall Accuracy of the entire model.

In the 1st-stage of both models, we can see that removing any type of feature reduces the overall accuracy even by a tiny margin. Overlap and Topic are the most effective features for both models respectively as excluding those two features would yield more than 1.5% drop towards overall Accuracy. The BoW feature also has an impact although it is not as significant compared to when removed Overlap and Topic.

With respect to the 2nd-stage of our two-stage model, we can observe that BoW is the most effective feature and removing that feature largely decreases the Accuracy of

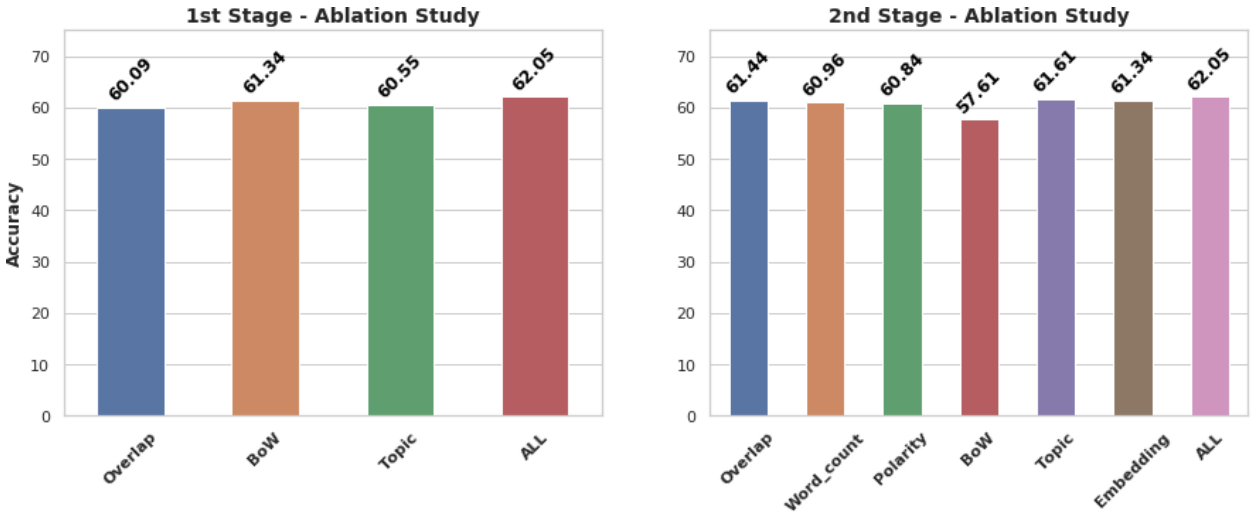

Figure 4.6: Feature ablation study for Two-stage LightGBM

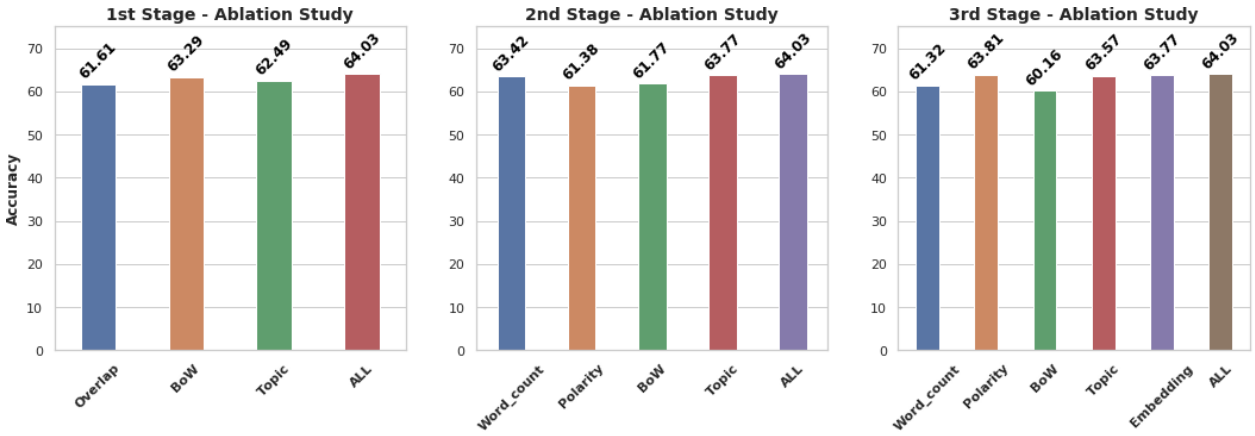

Figure 4.7: Feature ablation study for Three-stage LightGBM

the model. Excluding Overlap, Topic or Embedding feature causes a slight performance decrease which suggests that they exhibit low discriminative power in this stage. On the other hand, removing Word-count and Polarity decreases the Accuracy which proves their effectiveness at this stage.

As for the three-stage classification approach (especially for 2nd and 3rd stages), we find only a significant drop if we remove Polarity and BoW (for 2nd-stage), Word-count and BoW (for 3rd-stage). Removing Topic and Embedding features does not yield large performance decrease but they can generally improve the accuracy of both stages.

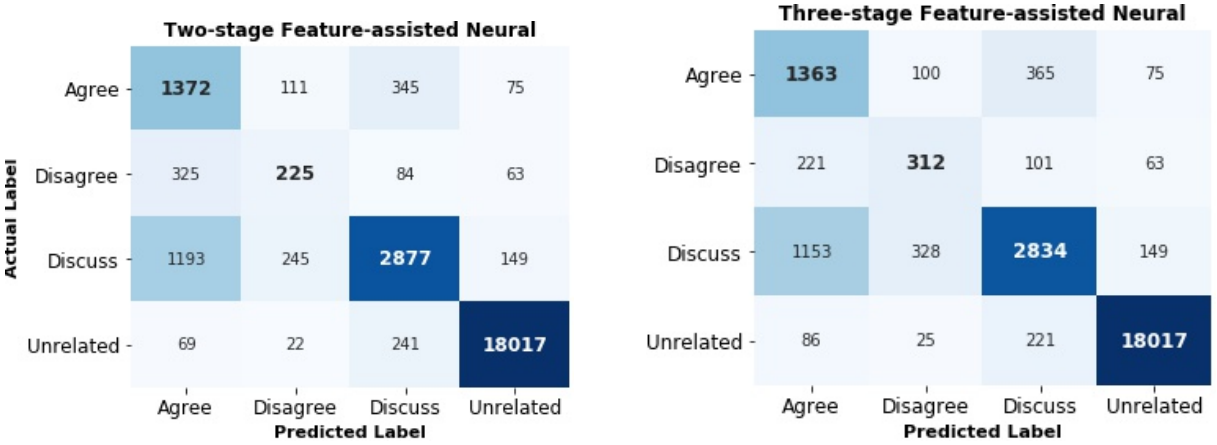

Figure 4.8: Multistage Feature-assisted Neural confusion matrices

4.5 Discussion

Although our models achieved state-of-the-art results in terms of Accuracy and F1-score, there are some misclassification cases which are shown in the confusion matrices for multi-stage feature-assisted neural models in Figure 4.8. To understand the cause of the errors, we conduct a deep analysis of the confusion matrix as well as the test dataset. As we mentioned before, the *unrelated* category is the easiest class to predict (regardless of any of the models) because two-thirds of the dataset belongs to that category. We observe the largest *unrelated* samples are misclassified as the *discuss* category for all four models due to the use of the same classifier at their first stage and there is some misclassifications between *unrelated* and *agree*. It is obvious that all four models can separate *disagree* from *unrelated* instances with few errors (63) compared to between other classes and 45 out of these 63 misclassification cases related to only one claim. The reason for these misclassifications comes from synonym use, especially when the claim and document use different vocabulary (synonyms) for the same word such as "bre-st" and "b–b". We can also see that feature-based systems have the advantage in discovering more on the *"discuss"* class while the feature-assisted neural models focus on improving the classification performance of *disagree* and *agree* classes. Most of the misclassifications occur in between *agree* and *discuss* classes in all four models. The reason for this is that these two classes (e.g. *agree* and *discuss*) share very similar textual contents and therefore, if a document is not taking

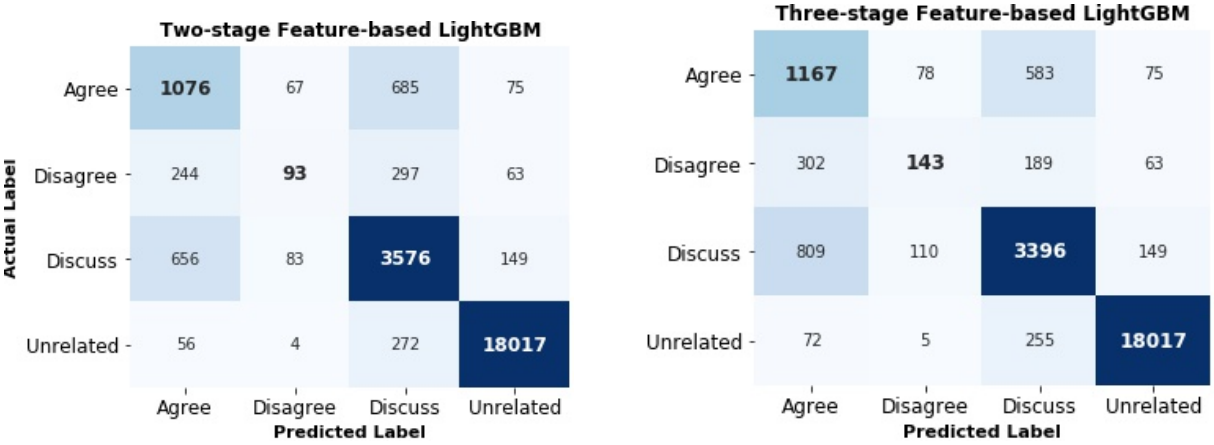

Figure 4.9: Multistage Feature-based LightGBM confusion matrices

a position (in other words *discuss* category) towards a claim then the document may also agree with the claim. Finally, the proposed models also classify a large portion of *disagree* pairs as *agree* category.

To further understand the reason for these *disagree* misclassifications towards *agree*, we take a closer look at a subset of errors from three-stage feature-assisted neural model as shown in Table 4.9. We observe errors from the following scenarios - (a) it is found that the model misclassifies 30 (1st example) and 21 (3rd example) out of 221 pairs that are associated with one claim *("Argentina's President Adopts Young Boy so He Won't Turn Into Werewolf")* and *("Justin Bieber Helps Defend Russian Fisherman From Bear")* respectively. The proposed model fails to semantically understand that these claims being in the category of *disagree* because the document discusses the claim very thoroughly without expressing explicit refuting e.g., ("The tradition of the president adopting a seventh child began in 1907 when then-president José Figueroa Alcorta, was asked by Russian immigrants") and ("The volume of the ringtone is probably what stopped the attack."). (b) On the other hand, there are 28 (2nd example) and 10 (5th example) out of 221 *disagree* instances that are misclassified as *agree* in Table 4.9. The text content may already give the model a clue (*"denied"* - 2nd example or *"denies"* - 5th example) but the model is unable to recognize these obvious negation instances that are mentioned in the news body or headline. (c) Finally, 21 out of 221 misclassifications belong to this claim ("This

Campaign Shows A Pretty Genius Way To Stop Men Catcalling In The Street"). Although there is no clear disagreement between the claims and documents, these instances are categorized as *disagree* pairs but our model predicts them as *agree*.

We explored two multistage classification approaches for stance detection. The two-stage and three-stage models we have presented were based on feature-based ML classifiers and feature-assisted neural models. It turns out that feature-based LightGBM models have outperformed compared with multiclass feature-assisted neural models in Chapter 3 and in (Saikh et al., 2019) by simply adapting multistage learning strategy with the best possible lexical-overlap features in each stage. However, having observed the effectiveness of the lexical-overlap features and the multistage setting, we have explored feature-assisted neural models that are particularly tailored for second-stage and third-stage classifiers. The three-stage feature-assisted neural model performs best on *disagree* class and the overall performance compared to the state-of-the-art (Baird et al., 2017; Hanselowski et al., 2018, 2017; Jwa et al., 2019; Riedel et al., 2017). The model also maintained a good performance on the other classes.

Even though high performance for underrepresented classes have been reported, there is still room for future research, mainly because the model is unable to recognize where there is no clear disagreement expressed within the news body or headline. As future work, we should investigate how different state-of-the-art language modeling architectures (e.g. BERT) can be leveraged to understand the deeper semantics of the article sentences and their interactions with respect to the headlines. We should use these powerful models in order to reduce the misclassified instances of agree and disagree categories and improve the performance of underrepresented classes in a multistage setting.

4.6 Conclusion

We address the problem of fake news by detecting the document-level stance from news articles in relation to news claims. We run various experiments in multistage classification

scenarios by taking advantage of the rich lexical-overlap and DL features built on top of sentence embeddings. The two-stage setting in the proposed multistage approaches is composed of two separate classifiers, namely a **relevancy** and a **stance** whilst the three-stage scenario further splits the stance detection problem into three binary classifiers: **relevance**, **stance** and **agreement**. Overall, both of our approaches achieved higher Accuracy and F1-score on the FNC-1 dataset compared to the state-of-the-art models. The next chapter makes use of multistage classification approaches based on feature-assisted neural models with the aim to explore the effectiveness of the model when applied on a different task.

Table 4.9: Examples of incorrect predictions

***Claim**: Argentina's President Adopts Young Boy so He Won't Turn Into Werewolf*
Stance: *disagree* - ***Model predicts***: *agree*
Body: [...] Apparently this is what happens when a tradition and an urban legend get intertwined. The tradition does dictate that the seventh child born to an Argentine family with six consecutive children of the same sex is eligible to become the godchild of the president; and the urban myth says this child's fate is unlucky as it is meant to become a werewolf sooner or later. So technically Lair Tawil was safe. [...] The tradition of the president adopting a seventh child began in 1907 when then-president José Figueroa Alcorta, was asked by Russian immigrants Enrique Brost and Apolonia Holmann to become their son's godfather. "The couple wanted to maintain a custom from Czarist Russia, where the Tsar was said to become godfather to seventh sons, and Argentina's president accepted." [...]
***Claim**: Joan Rivers: Doctor Took Selfie With Unconscious Comedian, Performed Biopsy Without Consent – Details*
Stance: *disagree* - ***Model predicts***: *agree*
Body: According to TMZ, Joan Rivers's personal doctor, Gwen Korovin, has denied CNN's allegations that she took a selfie in the operating room while the 81-year-old was under anesthesia, saying that the network's source is "making up lies." Korovin also denied performing the unauthorized biopsy that allegedly caused Rivers to go in to cardiac arrest. TMZ claims that it "pressed to find out if she performed some other procedure" but did not receive an answer.
***Claim**: Justin Bieber Helps Defend Russian Fisherman From Bear*
Stance: *disagree* - ***Model predicts***: *agree*
Body: A Russian man was recently able to fend off a bear attack with a Justin Bieber song. Forty-two year old Igor Vorozhbitsyn was recently fishing in Russia's Yakutia Republic when he was attacked by a bear. [...] The volume of the ringtone is probably what stopped the attack. One expert told Central European News that an unexpected noise, like a ringtone, can stop an angry bear in its tracks.
***Claim**: This Campaign Shows A Pretty Genius Way To Stop Men Catcalling In The Street*
Stance: *disagree* - ***Model predicts***: *agree*
Body: The Peruvian TV show "Harassing Your Mother" performs secret makeovers on the mothers of habitual catcallers, then uses hidden cameras to record catcallers shouting sexual remarks at their own mothers, who furiously upbraid them in the middle of the busy streets of Lima.
***Claim**: US denies it threatened Foley family*
Stance: *disagree* - ***Model predicts***: *agree*
Body: The US threatened to prosecute James Foley's family over ransom payments.

CHAPTER 5

MULTISTAGE POLITICAL FAKE STATEMENT DETECTION

In this chapter, we present the application of multistage classification approach based on feature-assisted neural models for Political Fake Statement Detection. The chapter is categorized as follows: the Section 5.1 formulates the problem and we further discuss the details of the methodology in Section 5.2. Section 5.3 presents the dataset and the experimental setup. Finally, the results are presented in Section 5.4 followed by some discussion and the conclusion of the chapter in Section 5.5 and 5.6 respectively.

5.1 Introduction

Fake news is disseminated to mislead the audience which is a huge problem and it is not always a simple binary classification. Rather, it can be regarded as a multiclass classification problem and therefore, Wang (2017) has introduced a benchmark fine-grained LIAR dataset of claims with varying degrees of truth (e.g, *pants-on-fire*, *false*, *barely-true*, *half-true*, *mostly-true*, and *true*) and the associated metadata for each claim. This

dataset is extracted from PolitiFact website[1] that fact-checks the accuracy of political claims based on Truth-O-Meter which has six ratings from *"True"* for accurate claims to *"Pants-on-Fire"* for completely false statements.

The current literature in fake news detection consists of models that detect fake-news content based on different news lengths (Rashkin et al., 2017) - short comments (Political Fake Statement) or news articles . The underlying assumption is that datasets with news articles have rich features whereas features created from short comments are insufficient to identify fake news. However, some of the existing approaches contributed to improving the detection of political fake statement (Wang, 2017) by combining the short statement and its associated speaker metadata where credit history of the speaker significantly increases the performance of the models by a large margin (Long et al., 2017; Liu et al., 2019; Wang et al., 2019b). However, in reality, the credit history of the speakers may not be available to help the model decide if the claims are true or not (Long et al., 2017).

Fake-news detection models, particularly our stance detection methods in Chapters 3 and 4, have shown that it might be useful to understand what reporters or human fact-checkers are reporting about a particular claim in order to identify fake news. As a step towards this direction, Alhindi et al. (2018) and Karimi et al. (2018) proposed models that combine claims, associated metadata and their justifications - where the justification is provided by human fact-checkers from Poltifact as a rationale of their verdict in relation to a specific claim. Despite reporting significant improvement compared to the state-of-the-art, we believe there is room for improvement. The performance of these models largely rely on credit history to detect the degree of fakeness as much as justification. In addition, the state-of-the-art models usually tackle this task as a single-stage multiclass classification which makes the problem more difficult to distinguish between fine-grained classes considering a single statement could be partially true and false at the same time.

In order to analyze the fine-grained categories of the LIAR dataset, we present

[1] http://www.politifact.com/

two novel multistage models for Political Fake Statement Detection. First, we explore a five-stage classification model which divides the six fine-grained classes into five binary classifiers and applies a feature-assisted neural model in each sub-stage. Second, we design a three-stage model that merges the benefits of DL and lexical-overlap features in a multistage setting: where the initial stage a four-class classifier is trained to categorize the short comments as factual, incomplete, Manipulative and hoax. In the ensuing stage, we further classify the factual and hoax samples into their respective subcategories using two separate classifiers. We specifically apply a Gated Recurrent Unit (GRU) model with the aim to learn the sequence dependencies and the semantics of the text. We also compute lexical-overlap features including BoW, Word count, sentiment and overlapping features. We empirically demonstrate that the multistage feature-assisted neural models improve the state-of-the-art results over the LIAR dataset.

5.2 Methodology

Inspired by the multistage models in (Mukherjee et al., 2012; Dey et al., 2018; Park and Fung, 2017), we build a neural model enhanced with a number of lexical features in a multistage setting. In other words, a number of traditional NLP features are created to capture the lexical cues which are deemed important in identifying the deceptive information in short statements. A dual encoder GRU model is built with the effect of these domain specific features for the purpose of detecting fake news and improving its classification accuracy. This section presents the details of our methodology for this task.

5.2.1 Multistage classification approaches

Since political fake-statement detection categorizes the veracity labels of the text into six fine-grained classes from *true* to *pants-on-fire*, we can use multistage classification settings. This subsection presents two multistage classification design for a robust and intuitive solution:

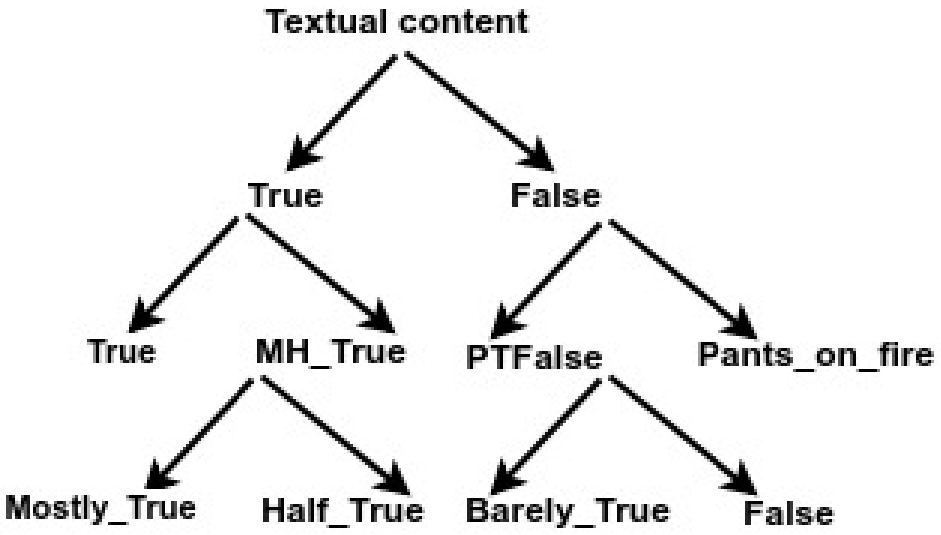

Figure 5.1: Five-Stage classification hierarchies

Five-stage classification

The FSD task can be approached as a number of binary classification problems in a multistage setting, as illustrated in Figure 5.1, where a feature-assisted neural model is trained in each stage with respect to identify the target binary labels. The first stage aims to categorize the statement into two primary classes of the task which is either *true* or *false* by using the designated stage one classifier. In the subsequent stages, we attempt to further distinguish into the more refined subcategories of the *true* (e.g. *true*, *mostly-true* and *half-true*) and *false* (e.g. *pants-on-fire*, *false* and *barely-true*) predictions by training two binary classifiers for each subcategory. In the case of *true* samples, the second stage classifier predicts *half-true* and *all-true* classes where *all-true* class stands for the *true* and *mostly-true* sub-classes. The third stage classifier further divides the *all-true* predictions with respect to the target sub-classes. Following this procedure, the fourth and fifth classifiers are trained in such a way as to predict the remaining categories respectively: *barely-true* vs *all-false* as well as *pants-on-fire* vs *false*.

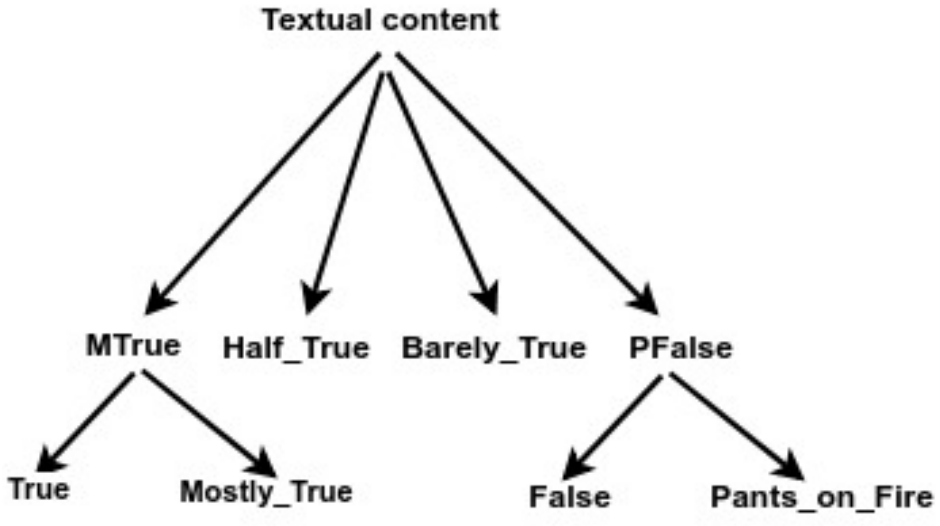

Figure 5.2: Three-Stage classification hierarchies

Three-stage classification

We follow the taxonomic hierarchy suggested by (Wang et al., 2019a) which proposes four categories of politic factual statements: - *factual*, *incomplete*, *Manipulative* and *hoax*. Therefore, we approached the FSD task in a three-stage classification manner as we have investigated deep neural models with the help of various lexical-overlap features. A hierarchical illustration of this model is shown in Figure 5.2. In the first-stage, an initial classifier is trained with the aim to classify among the following 4 classes: *factual* (*true* and *mostly-true*), *incomplete* (*half-true*), *manipulative* (*barely-true*) and *hoax* (*pants-on-fire* and *false*). This is followed by two binary classifiers to further split the factual and hoax samples into the remaining four categories. For example - the second classifier aims at categorizing those samples identified as factual into *true* and *mostly-true* while the third classifier aims at distinguishing the hoax samples into *pants-on-fire* and *false*.

5.2.2 DL Model

Figure 5.3 presents the architectural overview of the feature-assisted neural model. The statement and the justification pairs are given to GloVe embeddings (Pennington et al., 2014), which transforms the text into input representations for dual GRU layers. The

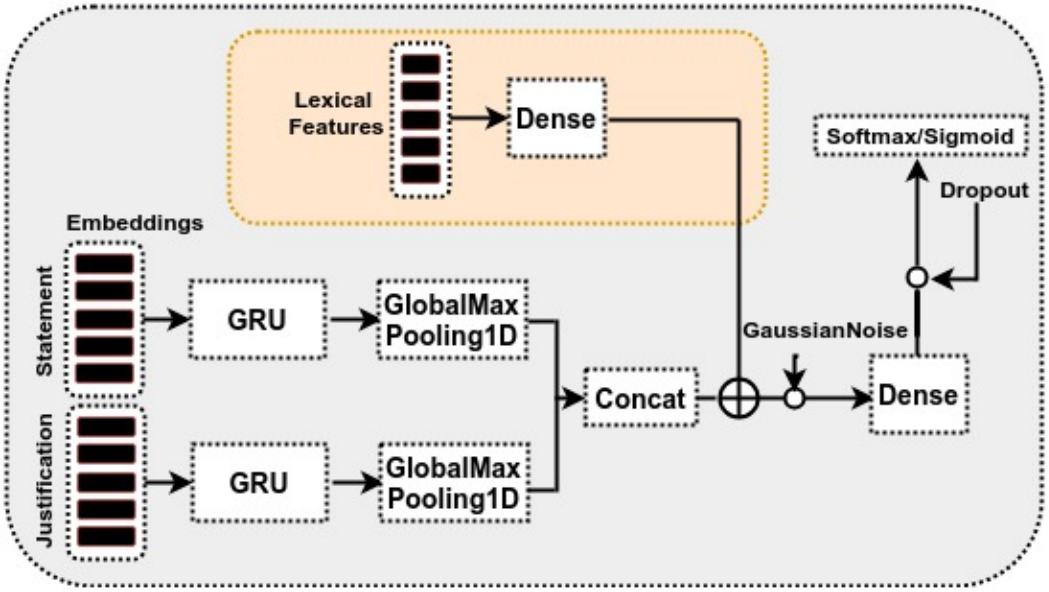

Figure 5.3: Feature-assisted Neural model

output of GRU layers go through global-max-pooling layers separately, which helps to minimize the dimensionality and generates the most important features. For merging the GRU layers, we concatenate these two vectors with lexical features (see next section) to incorporate important cues that can improve the performance of the neural models. Gaussian Noise layer is applied after the concatenation. Following this, the resulting output vector is passed through dense and dropout layers. Finally, the result produced is given to a final layer with either softmax (e.g. multiclass) or sigmoid (e.g. binary-class) activation function for final prediction.

5.2.3 Lexical Features

Our previous work on fake-news stance detection (refer to Chapters 3 and 4) explored different lexical features which can be categorized into word-count, sentiment and emotion, BoW and word co-occurrence categories. We conduct feature-ablation study for each sub-stage classifier in our two multistage settings to extract the best lexical features by using LightGBM classifier. Based on our experimental analysis, the following lexical features in Table 5.1 and 5.2 are incorporated into the neural models. We give concise details about the adopted features:

Table 5.1: Features and hyperparameters settings for the Five-stage study

Lexical-overlap Features					
	1st	**2nd**	**3rd**	**4th**	**5th**
Word Counts	RFT, CUE, HDG	CUE, HDG, IMP	CUE, HDG, IMP	RFT, CUE, NEG	RFT, CUE
Sentiments	POL	EMO, MPQA			EMO
BoW	BoW, BSIM, BNEG	BoW		BoW, BNEG	BoW, BSIM
Overlaps	ENT	ENT			
Hyperparameters for our DL Models					
GRU	64, 200	100, 256	64, 200	64, 200	64, 200
Dropout	D(0.1) RD(0.2)	D(0.1) RD(0.2)	D(0.1) RD(0.2)	D(0.1) RD(0.2)	D(0.1) RD(0.2)
Gaussian-noise	0.05	0.05	0.05	0.05	0.05

Word-count features

Based on different lexicons such as refuting-RFT words (fake, fraud, hoax, deny, etc,) (Pomerleau and Rao, 2017), cues-CUE words (doubt words, denial words, fake words, negative words, etc.) (Bahuleyan and Vechtomova, 2017), hedging-HDG words (about, claim, essentially, perhaps, etc.) and implicative-IMP words (manage, misfortune, neglect, decline, etc.) (Mukherjee et al., 2012), we compute how many of these markers appear in the claim statements and their relevant justifications.

Sentiments and emotional features

We compute sentiment polarity score, subjectivity score and emotion score features of the claim and the justification via VADER (Hutto and Gilbert, 2014) sentiment-POL analysis, MPQA subjectivity lexicon (Wilson et al., 2005) and Depechemood emotion-EMO detection (Staiano and Guerini, 2014).

Table 5.2: Features and hyperparameters settings for the Three-stage study

Lexical-overlap Features			
	1st	**2nd**	**3rd**
Word Counts	RFT, HDG	CUE, HDG, IMP	RFT, CUE
Sentiments	POL		POL
BoW	BoW, BSM, BNG	BoW, BSIM	BoW, BSM, BNG
Overlaps	OVP, ENT	OVP, ENT	OVP
Hyperparameters for our DL Models			
GRU	64, 200	64, 200	128, 256
Dropout	0.1, 0.2	0.15, 0.25	0.2, 0.25
Gaussian-noise	0.05	0.11	0.1

BoW Features

We enrich our multistage approaches with the TF representation of statement and justification to provide additional relevant information however we only retain 2000 most frequent 1, 2 and 3-grams BoW. Moreover, we generate the cosine similarity between statement and justification TF-IDF vectors-BSIM. We also use BoW vectors as a negation handling for example - we add a "_NEG" tag as a prefix for those 500 most frequent BoW-BNG that appear after a negative keyword (e.g. "no", "not", "don't") within the border of a clause-level punctuation mark (refer to Chapter 3).

Word and entity co-occurrence features

For the co-occurrence feature, we extract the overlapping-OVP word ngrams between statement and justification (Pomerleau and Rao, 2017). Using spacy toolkit, we generate bag-of-entities representation from the statement and justification after which we then compute an entity co-occurrence-ENT.

5.3 Experiments

In this section, we present the model settings and the used datasets to evaluate the proposed multistage models.

5.3.1 Model Settings

We use Keras for the implementation of our multistage models. Figure 5.1 and 5.2 present the high-level overview of our models. The input text is transformed into low-dimensional vectors using 100-dimensional GloVe vectors (Pennington et al., 2014). We use the validation dataset to select the best lexical features and tune the hyperparameters that produce the optimal results for our multistage approaches. As a result of this, the key lexical-features and hyperparameters considered for the proposed models are presented in Table 5.1 and Table 5.2. The length of the sentence and the justification are set to the maximum of 55 and 100 respectively. The batch size of 64 and Adam optimizer, with either categorical or binary cross-entropy as a loss function, are employed to train our models for 30 epochs. Early stopping is applied to stop the training once the validation loss does not decrease after three consecutive epochs.

5.3.2 Data

We evaluate the performance of our approach on one of the largest public FSD dataset called LIAR (Wang, 2017) that is commonly used by previous literature. This dataset is manually collected from Politifact and it contains a total of 12,836 short statements (S) and the associated metadata (e.g., speaker, party, state, subject, context/venue, and credit-history (CH)) labelled with six predefined veracity labels - *pants-on-fire*, *false*, *barely-true*, *half-true*, *mostly-true* and *true*. We specifically use the LIAR-PLUS dataset (Alhindi et al., 2018) which is an extended version that adds one more field to the contents of the LIAR - the ruling justification (J) provided by the Politifact human experts. Based on stance detection setup, we can use the justification as an evidence that confirms or denies with respect to the claim. We use the same split of the benchmark dataset where the author randomly divided into three partitions: train (80%), validation (10%) and test (10%). The distribution of the fine-grained classes is fairly balanced for all labels which is in between 2,063 to 2,638 instances except for the pants-on-fire samples (e.g. 1,050). For

Table 5.3: Baseline models (Text (T), Justification (J) and Metadata (M))

Study	Model	Sources
5-stage baseline	LightGBM + lexical features	T and J
3-stage baseline	LightGBM + lexical features	T and J
Wang (2017)	CNN + BiLSTM + word2vec	T and M
Long et al. (2017)	LSTM + Attention + word2vec	T and M
Wang et al. (2019b)	CNN + multi-head self-attention	T and M
Liu et al. (2019)	Fine-tune BERT	T and M
Alhindi et al. (2018)	Traditional + neural classifiers	T, J and M
Karimi et al. (2018)	MMFD	T, J and M

evaluation purposes, we report Accuracy and F1-score metrics that are separately used in the literature on the LIAR dataset.

5.3.3 Baselines

We empirically compare the performance of our multistage approaches against the following baselines including LightGBM classifier with the proposed lexical features. We perform an exhaustive comparison with prior approaches on Politifact dataset using Accuracy and F1 evaluation metrics. We directly report the state-of-the-art results from the publications. Table 5.3 presents the state-of-the-art and LightGBM baselines.

5.4 Results

In this section, we report the experimental results of the proposed models.

Comparison with the state-of-the-art models

Table 5.4 reports the performance of our models as well as the previous best performing models. The five-stage and three-stage LightGBM baselines surpass the state-of-the-art systems by a slight margin except the model proposed by (Wang et al., 2019b). This suggests that a feature-assisted neural model can be beneficial for this task in a multistage setting where most of the previous studies (Wang, 2017; Long et al., 2017; Liu et al., 2019;

Table 5.4: Comparison with the state-of-the-art models

Models	Accuracy(%)	F1-score(%)
Baselines		
5-stage LGBM	40.73	40.19
3-stage LGBM	**41.28**	**40.83**
State-of-the-art with S, ALL metadata and CH		
Wang (2017)	27.40	–
Long et al. (2017)	41.50	–
Liu et al. (2019)	40.58	–
Wang et al. (2019b)	45.30	–
State-of-the-art with S&J, ALL metadata and CH		
Karimi et al. (2018)	38.80	–
Alhindi et al. (2018)	–	37.00
This Work - Feature-assisted DL		
5-stage S&J	45.04	45.02
3-Stage S&J	**46.13**	**45.31**
5-stage S+CH	43.94	43.06
3-Stage S+CH	**44.49**	**43.97**
5-stage S&J+CH	51.91	51.51
3-Stage S&J+CH	**52.23**	**52.26**

Wang et al., 2019b; Karimi et al., 2018) employ multiclass DL models.

We explore two multistage feature-assisted neural models where we incorporate the lexical-feature values into dual GRUs. By comparing with the state-of-the-art baselines that utilize statement, justification, credit history and other metadata ((Karimi et al., 2018) and (Alhindi et al., 2018)), we show more than 7% improvement in Table 5.4 by using multistage models with only two textual inputs - statement and justification. It turns out that five-stage and three-stage classification hierarchies also show state-of-the-art results in this task when compared to previous studies that use statement, credit history and other metadata (Wang, 2017; Long et al., 2017; Liu et al., 2019; Wang et al., 2019b). Both models achieve an Accuracy of 45.04% and 46.13% as well as F1-score of 45.02% and 45.31%. Although credit history cannot be expected to be available in real time, we assess its usefulness by adding our model as one of the lexical features and the results are shown in Table 5.4. In the statement and credit history condition for both multistage models, we show relatively close performance compared to the statement and

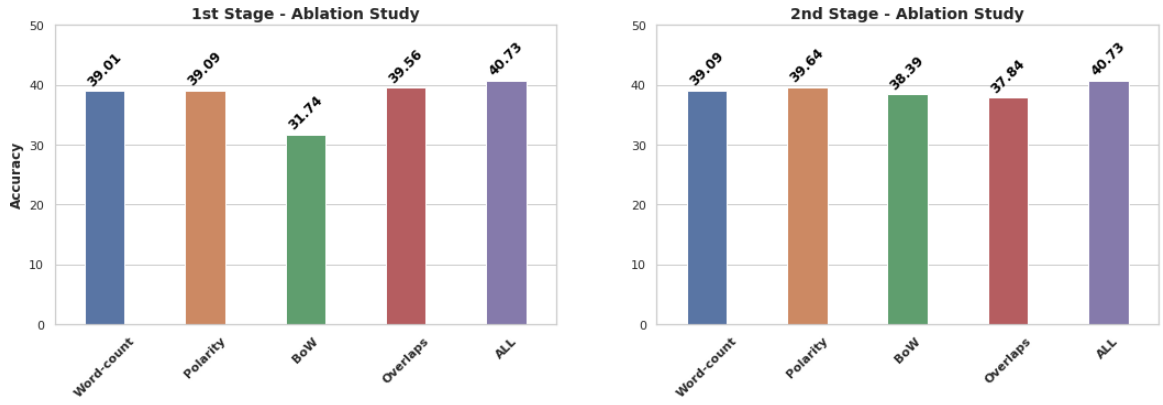

Figure 5.4: Five-stage feature ablation study (1st and 2nd)

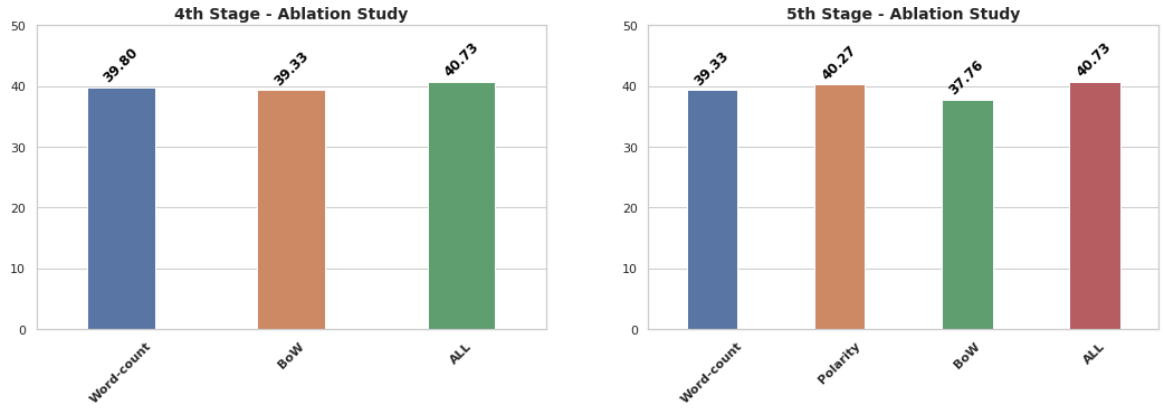

Figure 5.5: Five-stage feature ablation study (4th and 5th)

justification condition. As shown in Table 5.4, the classification performance of our models significantly improved by over 6% when using statement, justification and credit history condition. This indicates that the use of credit history might force the models to learn from the counts of speaker's prior inaccurate statements. Finally, we observe that the Accuracy and F1-score of our models are equivalent as Wang (2017) stated in his study since the LIAR dataset is well balanced.

Feature Ablation

We conduct feature ablation study using two multistage classification models with a LightGBM as a sub-stage classifier and four types of lexical features such as word-counts, polarities, BoW and overlaps. We start by comparing the performance of the five-stage classification model when removing one type of a feature-set from each sub-stage experi-

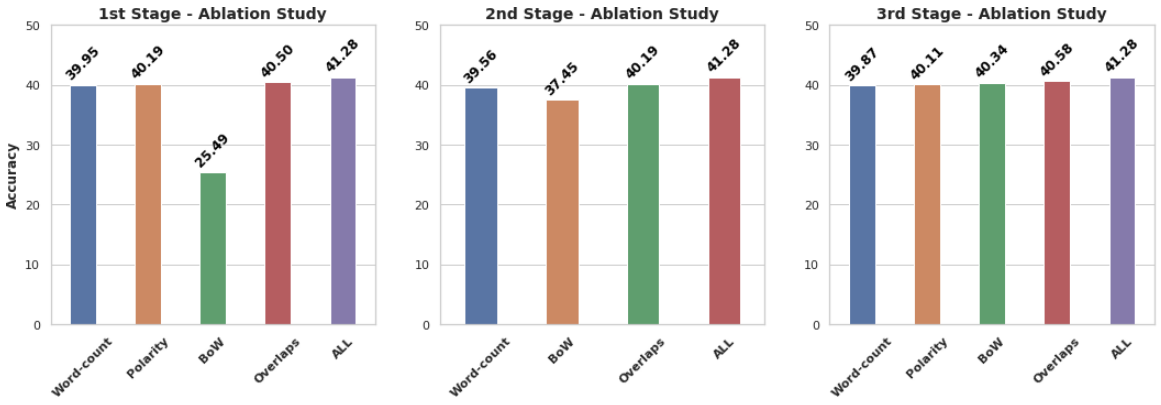

Figure 5.6: Three-stage feature ablation study

mental study (except the third stage which only utilizes the word-counts) and the results are presented on Figures 5.4 and 5.5. It is evident from the results that BoW are the most contributing features in the first-stage classification where all other three features closely contribute towards the final results. As can also be seen from the results, all four types of features are closely important towards the final performance of the second stage model although overlaps and BoW are most helpful to achieving the best accuracy. The best overall accuracy was achieved with a third-stage model utilizing only word-counts and a fourth-stage model leveraging word-counts as well as BoW while they are both important to obtain good performance. Our fifth-stage model without overlap features produces the highest classification accuracy besides that, if the BoW and word-counts are removed, then the overall accuracy drops confirming their importance for this stage.

Moreover, we evaluate the performance of lexical features in each sub-stage for a three-stage classification model. The results are given on Figure 5.6, presenting accuracy scores when again removed one type of a feature-set from training a sub-stage classifier. For instance, first-stage and third-stage classifiers achieved the highest accuracy to the usage of complete four types of features while the second-stage classifier would achieve good overall accuracy without using one of the lexical features (e.g. polarities). With respect to the first-stage and the second-stage of the pipeline, BoW are the most important type of features while all other features are almost equally important to obtain the final results. Regarding the third-stage classification, all the lexical features are essential to achieving

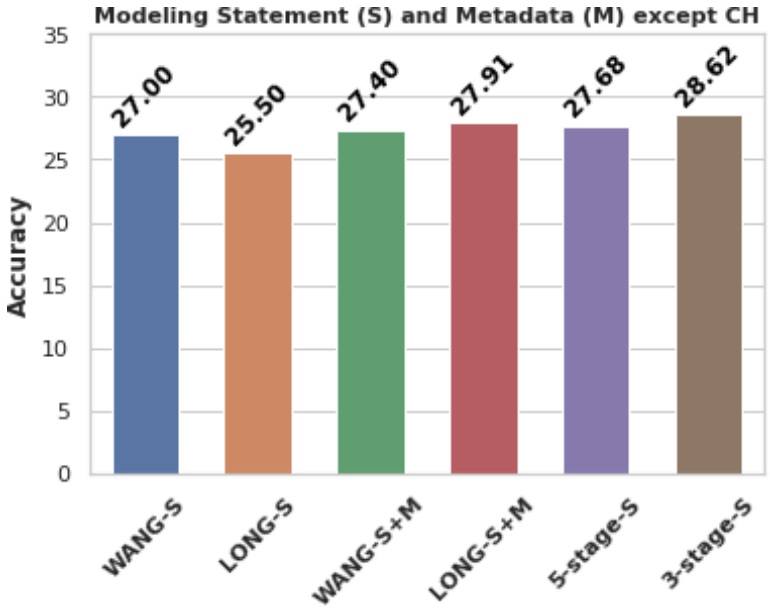

Figure 5.7: Performance comparison on statement

the best overall accuracy although word-counts seem to be slightly more important than other type of features.

Model Ablation

We investigate the ability of multistage (e.g. five and three) feature-assisted neural models to discriminate the fine-grained classes using statement as an input. We conduct experiments on using feature-assisted GRU (in each sub-stage of both hierarchies) as shown in Figure 5.3 but without justification. We only utilize all word-count features and 2000 most frequent BoW extracted directly from the statement. As a result, Figure 5.7 shows that modeling only the statement with five-stage and three-stage classification approaches outperforms the prior approaches including (Wang, 2017) and (Long et al., 2017). They also perform better than hybrid approaches (e.g. statements and metadata except credit history) (Wang, 2017; Long et al., 2017) due to the multistage adoption. In other words, multistage models proved to be effective without using the profile information of the speaker and the justification.

We further perform an ablation study for all possible combinations of the feature-assisted neural model to get insights into how important each component is for a better

Table 5.5: Ablation Study - Statement (S) Justification (J)

Models	Accuracy(%)	F1-score(%)
Five-stage models		
Single S+J	38.86	38.49
&Features	39.80	39.34
Dual-S&J	39.01	38.50
&Features	42.46	41.98
&Gaussian	45.04	45.02
Three-stage models		
Single S+J	39.01	38.00
&Features	41.05	40.86
Dual-S&J	40.58	39.53
&Features	42.22	41.14
&Gaussian	46.13	45.31

prediction. As illustrated in Table 5.5, we first examine the influence of either concatenating the statement and the justification into a **single** GRU layer or passing the two representations through **dual** GRU layers. We see that the dual GRU architecture gives us a slight improvement in both of the multistage settings. We also observe that adding lexical-overlap features into dual GRU layers leads to a slightly better performance compared to the single GRU layer. On the other hand, the overall performance is largely improved when the Gaussian Noise layer is introduced into the multistage setting in order to prevent overfitting. Overall, we find that utilizing the dual GRUs, lexical-features and Gaussian Noise layer produces higher performance compared to other combinations in both multistage models.

5.5 Discussion

We start by investigating the confusion matrices of three-stage model with and without credibility for the purpose of detecting the fine-grained classes of Political Fake Statement Detection. All models can distinguish between *true* and *pants-on-fire* classes with few errors as illustrated in Figure 5.8. We can see that all models misclassify the largest number of *pants-on-fire* instances as *false*. We observe that there is some misclassifications

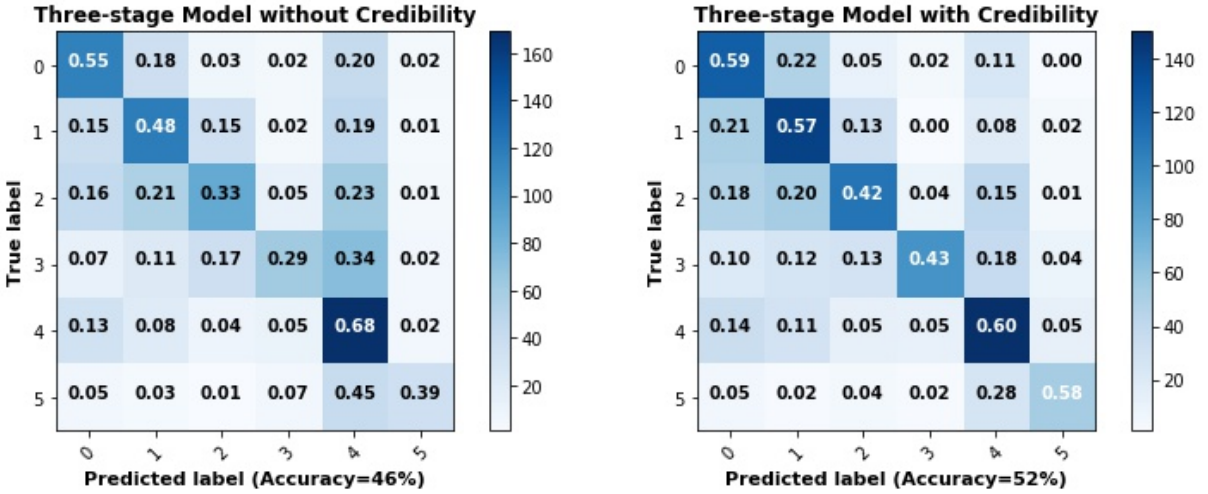

Figure 5.8: Three-stage Feature-assisted Neural confusion matrices on LIAR (*true-0*, *mostly-true-1*, *half-true-2*, *barely-true-3*, *false-4*, *pants-on-fire-5*)

between all *true* related classes (*true*, *mostly-true*, *half-true* and *barely-true*) and *false*, although the *false* class misclassifications towards *all-true* classes are minimal compared to the other way around. It is difficult to classify between *true* and *mostly-true*, and also our models struggle to separate *half-true* from *true* class. The model with credibility history improves the accuracy of all classes except the *false* category which the model still predicts the highest accuracy of all.

To explore the reasons for these misclassifications, we select a three-stage model with credibility history since it has the best overall Accuracy and F1-score. We take a look at several examples where we analyze the statement, justification and their predicted label as illustrated in Table 5.6. A large number of claim/justification pairs from *true* samples have a high number of misclassifications by being *mostly-true* because of the model's inability to discern the numerical values. As shown in the first two examples of Table 5.6, 31 out 47 (22%) misclassified pairs refer to numerical values which our model is not able to capture the numbers, dates, percentages and money information as they appear very frequently within the claims and justifications. The presence of refuting words (e.g. "does not", "doesn't", "impossible", "without", "fake") in the textual content does not necessarily imply disagreement between the pairs. With respect to third and fourth misclassification examples, our model does not understand that the refuting words

are merely in the discussion without taking any position, so the model predicts *false* instead of the correct category *"true"*. On the other hand, the last pair of Table 5.6 hold a *"pants-on-fire"* veracity on "If you want to bring America down", but affected by the word "fake", the predicted veracity of this pair is *false*. Finally, the fifth misclassification entry was wrongly classified as *false* instead of *pants-on-fire* class due to the presence of over-exaggeration words. For example, the content may already give the model a clue that over-exaggeration such as "years and years, I mean, hundreds of thousands of years" indicates *"pants-on-fire"* category but the model cannot resolve that and predicts the pair as *false*.

5.6 Conclusion

In this chapter, we study the problem of fine-grained Political Fake Statement Detection. We explore novel multistage approaches based on dual layer of neural models with lexical-overlap features extracted from statement and justification. We relate the problem to a stance detection task with the aim to improve the state-of-the-art performance on the LIAR dataset. Inspired by multistage classification approaches, we divide the problem into different sub-problems and construct five-stage and three-stage models to categorize the fine-grained labels of fake news. The extensive experimental study indicates that this approach can effectively classify fake news. We also conclude that using credit history boosts the performance by more than 6% compared against the state-of-the-art. In the next chapter, we provide a summary of the thesis contributions and possible lines of research which can be further investigated in the future studies.

Table 5.6: Examples of incorrect predictions

***Claim**: The Fed created $1.2 trillion out of nothing, gave it to banks, and some of them foreign banks, so that they could stabilize their operations.*
Veracity: *true* - ***Model predicts***: *mostly-true*
Justification: Two of the several foreign banks getting help were Royal Bank of Scotland Plc at $84.5 billion and UBS AG of Switzerland at $77.2 billion. [...] So while Kucinich's comment that "the Fed created $1.2 trillion out of nothing, , gave it to banks and some of them foreign banks" [...]
***Claim**: Says that in 2015, illegal immigrants accounted for 75 percent of federal drug possession convictions and 5 percent to 30 percent of convictions for murder and kidnapping plus two other crimes.*
Veracity: *true* - ***Model predicts***: *mostly-true*
Justification: Hannity's slide states that in fiscal 2015, "illegal immigrants" represented 5 percent to 75 percent of U.S. residents convicted of five federal crimes ranging from murder to simple drug possession. [...]
***Claim**: John McCain says it's okay with him if the U.S. spends the next thousand years in Iraq.*
Veracity: *true* - ***Model predicts***: *false*
Body: VoteVets has joined a coalition of liberal-leaning groups such as MoveOn.org that will be running ads against McCain.The VoteVets ad does not make the mistake Obama does.It says, "John McCain says it's okay with him if the U.S. spends the next thousand years in Iraq." That careful wording might imply the war might drag on that long, but it does not say it explicitly and we find the words are a fair summary of McCain's remarks.
***Claim**: The U.S. doesnt make television sets anymore.*
Veracity: *true* - ***Model predicts***: *false*
Body: Trump said the U.S. "doesn't make television sets anymore." An expert told us it's "impossible" to build a TV in the United States without relying heavily on imported components. Trump's statement ignores three companies that are, to varying degrees, assembling TVs in the United States.Two of them manufacture expensive niche products such as outdoor televisions.[...]
***Claim**: The Taliban has been there for years and years, I mean, hundreds of thousands of years.*
Veracity: *pants-on-fire* - ***Model predicts***: *false*
Justification: Meek clearly made a mid-debate stumble.He could have just stopped after saying that "the Taliban has been there for years and years" rather than adding "I mean, hundreds of thousands of years."
***Claim**: Says Joseph Stalin said if you want to bring America down you, have to undermine three things: our spiritual life, our patriotism and our morality.*
Veracity: *pants-on-fire* - ***Model predicts***: *false*
Body: Carson quoted Stalin as saying, "If you want to bring America down, you have to undermine three things: our spiritual life, our patriotism and our morality." All signs point to this being a fake quote that has made its way around the Internet.

CHAPTER 6

CONCLUSIONS

The main goal of this dissertation is to apply feature-assisted DL models in a multiclass or multistage classification setting for fake news detection. In Chapter 1, we first discuss the motivation behind the research project, we next layout the existing challenges and we also define the problems to be addressed in this manuscript. Finally, we wrap-up the potential contributions of our research. In Chapter 2, we present the literature background of fake news detection and its tasks, we further discuss the related work of this thesis and we finally explore the details of the proposed approaches. Overall, the contributions of this thesis are based on the following threefold solutions presented in Chapters 3, 4 and 5.

Chapter 3 presents an empirical assessment of different engineered statistical and similarity features extracted from headlines and article bodies to boost the performance on **Multiclass Stance Detection**. In addition, we explore text augmentation methods to create new training instances for minority classes on news-stance detection. We also present feature-assisted deep neural model and feature-based models for discriminating multiclass stance classification and we show the performances of different embeddings and regularization techniques used in order to improve the Accuracy and F1-score of the task. Finally, we show the effectiveness of our feature-assisted deep neural model over

different fake-news stance detection datasets (e.g. FNC-1 and ARC) and demonstrate that it outperforms the state-of-the-art. This work is published at SLSP 2019 (Hassan and Lee, 2019) as a regular Conference Paper.

Chapter 4 extends the study of previous chapter by proposing two **Multistage Stance Detection** models designed to merge the benefits of classical and neural classifiers. We present the first approach which is based on a **two-stage classification**: (1) a feature-based relevance classifier to determine if the news document is related or unrelated to a target headline. (2) a feature-assisted neural model with a GloVe word embeddings to predict the multiclass categories of agree, disagree and discuss. We also explore the second approach which is a **three-stage classification**: (1) 1st-stage - adopted from the two-stage classification (2) 2nd-stage - adopted from the two-stage classification but this time determines binary classes of discuss and agreement. (3) 3rd-stage - this stage aims to classify document/claim pairs into agree and disagree by using a feature-assisted neural model encoded with sentence representations generated from Universal Sentence Encoder (USE) vectors instead of GloVe word embeddings. We finally show that the proposed models achieve higher Accuracy and F1-score on the FNC-1 dataset for class-wise and overall predictions when compared against the state-of-the-art. The results of this work have been published at SLSP 2019 as a Poster and CISIS 2020 (Hassan and Lee, 2020a) as a regular Conference Paper.

In Chapter 5, we present two **Multistage Fake News Detection** models by considering pairs of claims and justifications as an input instead of metadata with the aim to improve the performance of the task. The chapter explores a five-level hierarchy prediction model with binary classifiers that uses feature-assisted neural-network in every level to individually tune and improve the overall performance of the fine-grained task. We also present a three-level model that utilizes feature-assisted DL in a multistage setting: where the first level a four-class classifier is trained to classify the pairs of text (statement and justification) as factual, incomplete, manipulative and hoax. In the second level, we further classify the factual and hoax samples into binary categories using

two feature-assisted neural models. We show a significant improvement over the state-of-the-art multiclass classification DL architectures on the LIAR dataset after applying multistage classification approaches. This work is published at ISI-IEEE 2020 (Hassan and Lee, 2020b) as a regular Conference Paper.

6.1 Research Questions Revisited

We revisit the research questions introduced in Chapter 1 and how our research contributions addressed them.

6.1.1 Multiclass Stance Detection

The first problem that this thesis has addressed was predicting stance of news articles regarding claims (or headlines) into one of these four classes: *agree*, *disagree*, *discuss* and *unrelated*, bearing in mind the existing challenge of unbalanced distribution of the dataset.

- **Research Question 1:** To what extent can the use of lexical and similarity feature representations influence the outcome of feature-based stance detection classifier? In addition, can a neural model be improved through the use of regularization techniques and lexical-overlap features for stance detection? And what is the effect of text augmentation for minority classes on this task?

The work presented in Chapter 3 has concluded that feature-based models have an important contribution towards stance detection. A conducted feature ablation study showed the influence of each feature using LightGBM classifier but this model struggled to predict the *disagree* class due to class imbalance problem. However, a feature-based shallow MLP classifier improved the class-wise and overall performance as this model demonstrated how important DL models can be if we combine these lexical and similarity features. By exploring towards the DL direction, we showed that neural model alone cannot perform for this particular problem due to overfitting on a smaller dataset. We also showed how the

feature-assisted neural model optimized with regularization layers outperformed against the state-of-the-art complex DL models for this domain by improving class-wise and overall performance.

Moreover, we demonstrated how this model can perform to a different domain dataset with similar setup in order to test its generalizability and it has shown better performance compared to the state-of-the-art. Unfortunately, the performance improvement towards the minority represented classes is limited although it is the highest according to the literature but the detection of those stances (e.g. *agree* and *disagree*) are important for fact-checking. When generated augmented instances for minority classes, our experiments illustrated that the feature-based classifiers and a feature-assisted neural classifier trained with expanded dataset using data augmentation performed better than models without augmentation on minority classes. We concluded that multiclass classifiers can easily be influenced towards predicting more on the majority classes but learning the fine-grained categories separately in cascaded classifiers may improve the predictions of the minority classes.

6.1.2 Multistage Stance Detection

The second problem related to the possibility of multistage classification which can address the challenges of multiclass stance detection.

- **Research Question 2:** Given the success of feature-based ML and feature-assisted neural classifiers, to what extent does a multistage classification affect the class-wise and overall performance of stance detection?

We confirmed that multistage classification approaches produce better results compared to multiclass classifications. We divided the multiclass categories into two-stage classification: relevance and stance, as well as three-stage classification: relevance, stance and agreement. We built a feature-based traditional classifier with keyword overlap for relevance and feature-assisted neural classifiers for stance and agreement.

The results of Chapter 4 ultimately satisfied our assumption to separate the tasks into smaller sub-tasks that can easily be determined by their discriminative semantic information and built a feature-specific sub-stage classifier in a multistage setting. We observed that both of these multistage models significantly improved the results for minority classes as well as the overall Accuracy and F1-score which did not come at the expense of majority classes.

6.1.3 Multistage Political Fake Statement Detection

The final research problem explored the potential application of a multistage feature-assisted neural model given that the multiclass DL had the challenge to better classify the fine-grained political fake statement, considering a single statement could belong to classes that are partially true and false at the same time.

- **Research Question 3:** Considering statement and justification pairs as a stance detection task, can neural models benefit from the inclusion of lexical features in a multistage classification hierarchy for this task?

The work presented in Chapter 5 has demonstrated the ability of a multistage feature-assisted neural model in the domain of Political Fake Statement Detection. Adding lexical features to a DL model proved their effectiveness in capturing more discriminative features to improve the performance of the task in a multistage setting without using the credibility history.

In addition, we showed that incorporating the credibility history into the multistage classification significantly improved upon the accuracy of the task, well beyond 6% improvement against the state-of-the-art. By comparing the results of our multistage models against the multiclass DL classifiers in the literature, it turns out that simple feature-assisted neural model and the sub-dividing of the task into multiple classifiers can effectively classify the fine-grained categories (e.g. degree of truthfulness).

6.2 Future Work

Although we have addressed the research questions of this study, there are possible lines of research that can be further investigated in the future.

6.2.1 Check-worthy Claim Detection

The future work should explore models that can be able to detect factual or check-worthy claims in social networks where most of the misinformation perpetrators disseminate inaccurate stories. Researchers in this area still face the challenge to better identify check-worthy claims because, they have to consider not only the textual content but also the contextual information[1] (speaker, time, place, etc,.) when detecting factual claims. To date, most of the check-worthy claim detection research focuses on recognizing factual claims that make-up only one sentence however, detecting claims from a body of text, over more than one sentence, deserve more research to be carried-out.

6.2.2 Relevant Document Discovery

A domain-specific question answering system is needed to convert the check-worthy claim sentences into relevant questions for a search through the web documents. The form of the question should be considered when querying the Internet databases in order to get the relevant articles (or relevant text snippets) of interest. However, the existing question generation and answering tools could not handle this type of problem as Jimenez (2017) has proved hence, it is an open problem that needs researchers attention.

6.2.3 Objectivity Detection (Fake News Classification)

The major problem for this domain is lack of sufficient training dataset and that is why the likes of traditional and neural models have an average accuracy score, therefore, it is

[1] https://fullfact.org/blog/2016/aug/automated-factchecking/

important to create large dataset. Feature-assisted multistage approach can be applied for better modeling and this enables to build a sub-stage model that pays attention to category-specific discriminative semantic information, hence, further research is required.

6.2.4 Stance Detection

Our stance detection methodologies resulted promising performance when dealing with highly imbalanced datasets and these methodologies can be replicated in fake news detection and other text classification tasks respectively. Despite the state-of-the-art performance, there is still room for future research. Future studies should address the lack of large training datasets in fake news detection domain in general and the imbalanced nature of the stance detection datasets in particular. In line with a multistage setting, we should also investigate more sophisticated deep language models (e.g. BERT) that have a deeper semantic understanding of the text.

6.2.5 Fact-checking Pipeline (Claim Validation)

Over the last few years, fact-checking has emerged as one of the most active areas in Computational Journalism. Researchers in this area are now focused to accomplish the computational dream regarding the automated fact-checking. Vlachos and Riedel (2014) introduced the task of fact-checking as they gave details about how to build a dataset and possible approaches that can be used to tackle the task. As a future work, it is important to develop a real-time system that can automatically fact-check the veracity of a claim. The job of this kind of system should start with firstly classifying claim sentences into being check-worthy or non-check-worthy (Hassan et al., 2015b; Gencheva et al., 2017). Then, it should formulate a query from the factual sentences to search relevant articles (Hassan et al., 2017; Popat et al., 2017), objectively reported (Nakashole and Mitchell, 2014; Popat et al., 2017), through the Internet and should then assess the reliability of the source (Popat et al., 2017; Zhi et al., 2017; Choudhary et al., 2018). Finally, it should

return a relevant evidence, as a document, in order to detect the stance whether the evidence supports or denies in relation to the claim (Hanselowski et al., 2018; Zhi et al., 2017) and then the system should determine if the factual claim is true or not.

We can conclude that there are no gold-standard datasets for fact-checking and that is why every research has introduced its own unified task while experimenting on a proprietary dataset. To evaluate different approaches and solutions proposed for this area, it should be set out shared criteria and should be used common pipeline with a united-front to tackle against the problem of fact-checking.

REFERENCES

Amol Agrawal. Clickbait detection using deep learning. In *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, pages 268–272. IEEE, 2016.

Ahmet Aker, Leon Derczynski, and Kalina Bontcheva. Simple open stance classification for rumour analysis. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 31–39, 2017.

Essam Al Daoud. Comparison between xgboost, lightgbm and catboost using a home credit dataset. *International Journal of Computer and Information Engineering*, 13(1): 6–10, 2019.

César Alfaro, Javier Cano-Montero, Javier Gómez, Javier M Moguerza, and Felipe Ortega. A multi-stage method for content classification and opinion mining on weblog comments. *Annals of Operations Research*, 236(1):197–213, 2016.

Tariq Alhindi, Savvas Petridis, and Smaranda Muresan. Where is your evidence: Improving fact-checking by justification modeling. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 85–90, 2018.

Emily Allaway and Kathleen McKeown. Zero-shot stance detection: A dataset and model using generalized topic representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8913–8931, 2020.

Ankesh Anand, Tanmoy Chakraborty, and Noseong Park. We used neural networks to detect clickbaits: You won't believe what happened next! In *European Conference on Information Retrieval*, pages 541–547. Springer, 2017.

Pepa Atanasova, Alberto Barron-Cedeno, Tamer Elsayed, Reem Suwaileh, Wajdi Zaghouani, Spas Kyuchukov, Giovanni Da San Martino, and Preslav Nakov. Overview of the clef-2018 checkthat! lab on automatic identification and verification of political claims. task 1: Check-worthiness. *arXiv preprint arXiv:1808.05542*, 2018.

Pepa Atanasova, Preslav Nakov, Georgi Karadzhov, Mitra Mohtarami, and Giovanni Da San Martino. Overview of the clef-2019 checkthat! lab: Automatic identification

and verification of claims. task 1: Check-worthiness. In *In Working Notes of CLEF 2019 - Conference and Labs of the Evaluation Forum*, 2019a.

Pepa Atanasova, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, Georgi Karadzhov, Tsvetomila Mihaylova, Mitra Mohtarami, and James Glass. Automatic fact-checking using context and discourse information. *Journal of Data and Information Quality (JDIQ)*, 11(3):1–27, 2019b.

Giuseppe Attardi, Antonio Carta, Federico Errica, Andrea Madotto, and Ludovica Pannitto. Fa3l at semeval-2017 task 3: A three embeddings recurrent neural network for question answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 299–304, 2017.

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. Stance detection with bidirectional conditional encoding. *arXiv preprint arXiv:1606.05464*, 2016.

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 759–760, 2017.

Hareesh Bahuleyan and Olga Vechtomova. Uwaterloo at semeval-2017 task 8: Detecting stance towards rumours with topic independent features. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 461–464, 2017.

Sean Baird, Doug Sibley, and Yuxi Pan. Talos targets disinformation with fake news challenge (2017). https://blog.talosintelligence.com/2017/06/talos-fake-news-challenge.html, 2017.

Rushlene Kaur Bakshi, Navneet Kaur, Ravneet Kaur, and Gurpreet Kaur. Opinion mining and sentiment analysis. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 452–455. IEEE, 2016.

Timothy Baldwin, Huizhi Liang, Bahar Salehi, Doris Hoogeveen, Yitong Li, and Long Duong. Unimelb at semeval-2016 task 3: Identifying similar questions by combining a cnn with string similarity measures. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 851–856, 2016.

Ramy Baly, Georgi Karadzhov, Dimitar Alexandrov, James Glass, and Preslav Nakov. 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, 2018.

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 747–754, 2017.

Adrian Benton and Mark Dredze. Using author embeddings to improve tweet stance classification. In *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, pages 184–194, 2018.

Gaurav Bhatt, Aman Sharma, Shivam Sharma, Ankush Nagpal, Balasubramanian Raman, and Ankush Mittal. Combining neural, statistical and external features for fake news stance identification. In *Companion Proceedings of the The Web Conference 2018*, pages 1353–1357. International World Wide Web Conferences Steering Committee, 2018.

Prakhar Biyani, Kostas Tsioutsiouliklis, and John Blackmer. "8 amazing secrets for getting more clicks": detecting clickbaits in news streams using article informality. In *Thirtieth AAAI conference on artificial intelligence*, 2016.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

Dasha Bogdanova, Jennifer Foster, Daria Dzendzik, and Qun Liu. If you can't beat them join them: Handcrafted features complement neural nets for non-factoid answer reranking. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 121–131, 2017.

Luís Borges, Bruno Martins, and Pável Calado. Combining similarity features and deep representation learning for stance detection in the context of checking fake news. *Journal of Data and Information Quality (JDIQ)*, 2019.

Peter Bourgonje, Julian Moreno Schneider, and Georg Rehm. From clickbait to fake news detection: an approach based on detecting the stance of headlines to articles. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 84–89, 2017.

Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 675–684, 2011.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018.

Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. Stop clickbait: Detecting and preventing clickbaits in online news media. In *2016 ieee/acm international conference on advances in social networks analysis and mining (asonam)*, pages 9–16. IEEE, 2016.

Tao Chen, Ruifeng Xu, Yulan He, and Xuan Wang. Improving sentiment analysis via sentence type classification using bilstm-crf and cnn. *Expert Systems with Applications*, 72:221–230, 2017.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

Yimin Chen, Niall J Conroy, and Victoria L Rubin. Misleading online content: recognizing clickbait as "false news". In *Proceedings of the 2015 ACM on workshop on multimodal deception detection*, pages 15–19, 2015.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

Nurendra Choudhary, Rajat Singh, Ishita Bindlish, and Manish Shrivastava. Neural network architecture for credibility assessment of textual claims. In *Proceedings of the 2018 International Conference on Computational Linguistics and Intelligent Text Processing*, 2018.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini. Computational fact checking from knowledge networks. *PloS one*, 10(6):e0128193, 2015.

Costanza Conforti, Mohammad Taher Pilehvar, and Nigel Collier. Towards automatic fake news detection: Cross-level stance detection in news articles. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 40–49, 2018.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*, 2017.

Nadia K Conroy, Victoria L Rubin, and Yimin Chen. Automatic deception detection: Methods for finding fake news. *Proceedings of the Association for Information Science and Technology*, 52(1):1–4, 2015.

Giovanni Da San Martino, Alberto Barron-Cedeno, and Preslav Nakov. Findings of the nlp4if-2019 shared task on fine-grained propaganda detection. In *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 162–170, 2019a.

Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeno, Rostislav Petrov, and Preslav Nakov. Fine-grained analysis of propaganda in news article. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5640–5650, 2019b.

Sanjiv R Das and Mike Y Chen. Yahoo! for amazon: Sentiment extraction from small talk on the web. *Management science*, 53(9):1375–1388, 2007.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *In Proceedings of ICWSM*, 2017.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga. Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 69–76, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *In Proceedings of NAACL*, 2018.

Kuntal Dey, Ritvik Shrivastava, and Saroj Kaushik. Twitter stance detection—a subjectivity and sentiment polarity inspired two-phase approach. In *2017 IEEE international conference on data mining workshops (ICDMW)*, pages 365–372. IEEE, 2017.

Kuntal Dey, Ritvik Shrivastava, and Saroj Kaushik. Topical stance detection for twitter: A two-phase lstm model using attention. In *European Conference on Information Retrieval*, pages 529–536. Springer, 2018.

Li Dong, Furu Wei, Yichun Yin, Ming Zhou, and Ke Xu. Splusplus: a feature-rich two-stage classifier for sentiment analysis of tweets. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 515–519, 2015.

Omar Enayet and Samhaa R El-Beltagy. Niletmrg at semeval-2017 task 8: Determining rumour and veracity support for rumours on twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 470–474, 2017.

Wei Fang, Moin Nadeem, Mitra Mohtarami, and James Glass. Neural multi-task learning for stance prediction. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)*, pages 13–19, 2019.

William Ferreira and Andreas Vlachos. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1163–1168, 2016.

Antigoni Maria Founta, Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Athena Vakali, and Ilias Leontiadis. A unified deep learning architecture for abuse detection. In *Proceedings of the 10th ACM Conference on Web Science*, pages 105–114, 2019.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

Pepa Gencheva, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, and Ivan Koychev. A context-aware approach for detecting worth-checking claims in political debates. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 267–276, 2017.

Bilal Ghanem, Paolo Rosso, and Francisco Rangel. Stance detection in fake news a combined feature representation. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 66–71, 2018.

Bilal Ghanem, Alessandra Teresa Cignarella, Cristina Bosco, Paolo Rosso, and Francisco Manuel Rangel Pardo. Upv-28-unito at semeval-2019 task 7: Exploiting post's nesting and syntax information for rumor stance classification. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 1125–1131, 2019.

Praveen Kumar Badimala Giridhara, Chinmaya Mishra, Reddy Kumar Modam Venkataramana, Syed Saqib Bukhari, and Andreas Dengel. A study of various text augmentation techniques for relation classification in free text. *ICPRAM*, 3:5, 2019.

Njagi Dennis Gitari, Zhang Zuping, Hanyurwimfura Damien, and Jun Long. A lexicon-based approach for hate speech detection. *International Journal of Multimedia and Ubiquitous Engineering*, 10(4):215–230, 2015.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

Richard Gunther, Paul A Beck, and Erik C Nisbet. Fake news may have contributed to trump's 2016 victory. ohio state university. https://www.documentcloud.org/documents/4429952-Fake-News-May-Have-Contributed-to-Trump-s-2016.html, 2018.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. *arXiv preprint arXiv:1708.01425*, 2017.

Sardar Hamidian and Mona T Diab. Rumor detection and classification for twitter data. In *The Fifth International Conference on Social Media Technologies Communication, and Informatics, SOTICS, IARIA*, pages 71–77, 2015.

Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

Andreas Hanselowski, Avinesh PVS, Benjamin Schiller, and Felix Caspelherr. Description of the system developed by team Athene in the FNC-1 (2017). https://medium.com/@andre134679/team-athene-on-the-fake-news-challenge-28a5cf5e017b, 2017.

Andreas Hanselowski, PVS Avinesh, Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M Meyer, and Iryna Gurevych. A retrospective analysis of the fake news challenge stance-detection task. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1859–1874, 2018.

Naeemul Hassan, Bill Adair, James T Hamilton, Chengkai Li, Mark Tremayne, Jun Yang, and Cong Yu. The quest to automate fact-checking. In *Proceedings of the 2015 Computation+ Journalism Symposium*, 2015a.

Naeemul Hassan, Chengkai Li, and Mark Tremayne. Detecting check-worthy factual claims in presidential debates. In *Proceedings of the 24th acm international on conference on information and knowledge management*, pages 1835–1838, 2015b.

Naeemul Hassan, Gensheng Zhang, Fatma Arslan, Josue Caraballo, Damian Jimenez, Siddhant Gawsane, Shohedul Hasan, Minumol Joseph, Aaditya Kulkarni, Anil Kumar Nayak, et al. Claimbuster: the first-ever end-to-end fact-checking system. *Proceedings of the VLDB Endowment*, 10(12):1945–1948, 2017.

Felix Hill, Kyunghyun Cho, and Anna Korhonen. Learning distributed representations of sentences from unlabelled data. *arXiv preprint arXiv:1602.03483*, 2016.

Sushant Hiray and Venkatesh Duppada. Agree to disagree: Improving disagreement detection with dual grus. In *2017 Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, pages 147–152. IEEE, 2017.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 1997.

Benjamin D Horne and Sibel Adali. This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. *arXiv preprint arXiv:1703.09398*, 2017.

Clayton J Hutto and Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth international AAAI conference on weblogs and social media*, 2014.

Mohammed Jabreel and Antonio Moreno. Target-dependent sentiment analysis of tweets using a bi-directional gated recurrent unit. In *WEBIST*, pages 80–87, 2017.

Israa Jaradat, Pepa Gencheva, Alberto Barrón-Cedeño, Lluís Màrquez, and Preslav Nakov. Claimrank: Detecting check-worthy claims in arabic and english. *NAACL HLT 2018*, page 26, 2018.

Damian Jimenez. Towards building an automated fact-checking system. In *Proceedings of the 2017 ACM International Conference on Management of Data*, pages 7–9, 2017.

Zhipeng Jin, Qiudan Li, Daniel Zeng, and Lei Wang. Filtering spam in weibo using ensemble imbalanced classification and knowledge expansion. In *2015 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pages 132–134. IEEE, 2015.

Zhiwei Jin, Juan Cao, Yongdong Zhang, and Jiebo Luo. News verification by exploiting conflicting social viewpoints in microblogs. In *Thirtieth AAAI conference on artificial intelligence*, 2016.

Karen Spärck Jones. Automatic summarising: The state of the art. *Information Processing & Management*, 43(6):1449–1481, 2007.

Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. In *International conference on machine learning*, pages 2342–2350, 2015.

Heejung Jwa, Dongsuk Oh, Kinam Park, Jang Mook Kang, and Heuiseok Lim. exbake: Automatic fake news detection model based on bidirectional encoder representations from transformers (bert). *Applied Sciences*, 9(19):4062, 2019.

Richard L Kaplan. *Politics and the American press: The rise of objectivity, 1865-1920*. Cambridge University Press, 2002.

Georgi Karadzhov, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, and Ivan Koychev. Fully automated fact checking using external sources. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 344–353, 2017.

Hamid Karimi, Proteek Roy, Sari Saba-Sadiya, and Jiliang Tang. Multi-source multi-class fake news detection. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1546–1557, 2018.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in neural information processing systems*, pages 3146–3154, 2017.

Johannes Kiesel, Maria Mestre, Rishabh Shukla, Emmanuel Vincent, Payam Adineh, David Corney, Benno Stein, and Martin Potthast. Semeval-2019 task 4: Hyperpartisan news detection. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 829–839, 2019.

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302, 2015.

Sosuke Kobayashi. Contextual augmentation: Data augmentation by words with paradigmatic relations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 452–457, 2018.

Elena Kochkina, Maria Liakata, and Isabelle Augenstein. Turing at semeval-2017 task 8: Sequential approach to rumour stance classification with branch-lstm. *arXiv preprint arXiv:1704.07221*, 2017.

Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. All-in-one: Multi-task learning for rumour verification. *arXiv preprint arXiv:1806.03713*, 2018.

Lev Konstantinovskiy, Oliver Price, Mevan Babakar, and Arkaitz Zubiaga. Towards automated factchecking: Developing an annotation schema and benchmark for consistent automated claim detection. *arXiv preprint arXiv:1809.08193*, 2018.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

Miroslav Kubat, Robert C Holte, and Stan Matwin. Machine learning for the detection of oil spills in satellite radar images. *Machine learning*, 30(2-3):195–215, 1998.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966, 2015.

Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, and Yajun Wang. Prominent features of rumor propagation in online social media. In *2013 IEEE 13th International Conference on Data Mining*, pages 1103–1108. IEEE, 2013.

Ju-Hong Lee, Sun Park, Chan-Min Ahn, and Daeho Kim. Automatic generic document summarization based on non-negative matrix factorization. *Information Processing & Management*, 45(1):20–34, 2009.

Cédric Lespagnol, Josiane Mothe, and Md Zia Ullah. Information nutritional label and word embedding to estimate information check-worthiness. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 941–944, 2019.

Quanzhi Li, Qiong Zhang, Luo Si, and Yingchi Liu. Rumor detection on social media: Datasets, methods and opportunities. In *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda. Association for Computational Linguistics*, pages 66–75, 2019a.

Xian Li, Weiyi Meng, and Clement Yu. T-verifier: Verifying truthfulness of fact statements. In *2011 IEEE 27th International Conference on Data Engineering*, pages 63–74. IEEE, 2011.

Yaliang Li, Jing Gao, Chuishi Meng, Qi Li, Lu Su, Bo Zhao, Wei Fan, and Jiawei Han. A survey on truth discovery. *ACM Sigkdd Explorations Newsletter*, 17(2):1–16, 2015.

Yukun Li, Zhenguo Yang, Xu Chen, Huaping Yuan, and Wenyin Liu. A stacking model using url and html features for phishing webpage detection. *Future Generation Computer Systems*, 94:27–39, 2019b.

Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.

Chao Liu, Xinghua Wu, Min Yu, Gang Li, Jianguo Jiang, Weiqing Huang, and Xiang Lu. A two-stage model based on bert for short fake news detection. In *International Conference on Knowledge Science, Engineering and Management*, pages 172–183. Springer, 2019.

Shigang Liu, Yu Wang, Jun Zhang, Chao Chen, and Yang Xiang. Addressing the class imbalance problem in twitter spam detection using ensemble learning. *Computers & Security*, 69:35–49, 2017.

Yunfei Long, Qin Lu, Rong Xiang, Minglei Li, and Chu-Ren Huang. Fake news detection through multi-perspective speaker profiles. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 252–256, 2017.

Marianela García Lozano, Hanna Lilja, Edward Tjörnhammar, and Maja Karasalo. Mama edha at semeval-2017 task 8: Stance classification with cnn and rules. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 481–485, 2017.

Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. Detecting rumors from microblogs with recurrent neural networks. 2016.

Jing Ma, Wei Gao, and Kam-Fai Wong. Detect rumor and stance jointly by neural multi-task learning. In *Companion Proceedings of the The Web Conference 2018*, pages 585–593, 2018a.

Yukun Ma, Haiyun Peng, and Erik Cambria. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm. In *Aaai*, pages 5876–5883, 2018b.

Bill MacCartney, Trond Grenager, Marie-Catherine de Marneffe, Daniel Cer, and Christopher D Manning. Learning to recognize features of valid textual entailments. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, page 41. Citeseer, 2006.

Razan Masood and Ahmet Aker. The fake news challenge: Stance detection using traditional machine learning approaches. In *Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (KMIS)*, pages 128–135, 2018.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR, abs/1301.3781*, 2013.

George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

Saif M Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. Stance and sentiment in tweets. *ACM Transactions on Internet Technology (TOIT)*, 17(3):1–23, 2017.

Mitra Mohtarami, Ramy Baly, James Glass, Preslav Nakov, Lluís Màrquez, and Alessandro Moschitti. Automatic stance detection using end-to-end memory networks. *arXiv preprint arXiv:1804.07581*, 2018.

Jonas Mueller and Aditya Thyagarajan. Siamese recurrent architectures for learning sentence similarity. In *thirtieth AAAI conference on artificial intelligence*, 2016.

Subhabrata Mukherjee and Gerhard Weikum. Leveraging joint interactions for credibility analysis in news communities. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 353–362, 2015.

Subhabrata Mukherjee, Akshat Malu, Balamurali Ar, and Pushpak Bhattacharyya. Twisent: a multistage system for analyzing sentiment in twitter. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2531–2534, 2012.

Moin Nadeem, Wei Fang, Brian Xu, Mitra Mohtarami, and James Glass. Fakta: An automatic end-to-end fact checking system. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 78–83, 2019.

Ndapandula Nakashole and Tom M Mitchell. Language-aware truth assessment of fact candidates. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1009–1019, 2014.

Ani Nenkova and Kathleen McKeown. A survey of text summarization techniques. In *Mining text data*, pages 43–76. Springer, 2012.

Dat Quoc Nguyen, Son Bao Pham, et al. A two-stage classifier for sentiment analysis. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 897–901, 2013.

Ray Oshikawa, Jing Qian, and William Yang Wang. A survey on natural language processing for fake news detection. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6086–6093, 2020.

Makbule Gulcin Ozsoy, Ferda Nur Alpaslan, and Ilyas Cicekli. Text summarization using latent semantic analysis. *Journal of Information Science*, 37(4):405–417, 2011.

Ji Ho Park and Pascale Fung. One-step and two-step classification for abusive language detection on twitter. *arXiv preprint arXiv:1706.01206*, 2017.

Ayush Patwari, Dan Goldwasser, and Saurabh Bagchi. Tathya: A multi-classifier system for detecting check-worthy statements in political debates. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 2259–2262, 2017.

Andraž Pelicon, Marko Pranjić, Dragana Miljković, Blaž Škrlj, and Senja Pollak. Zero-shot learning for cross-lingual news sentiment classification. *Applied Sciences*, 10(17): 5993, 2020.

James W Pennebaker, Martha E Francis, and Roger J Booth. Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, 71(2001):2001, 2001.

Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. Automatic detection of fake news. *arXiv preprint arXiv:1708.07104*, 2017.

Matthew E Peters, Sebastian Ruder, and Noah A Smith. To tune or not to tune? adapting pretrained representations to diverse tasks. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 7–14, 2019.

Francesco Pierri and Stefano Ceri. False news on social media: a data-driven survey. *ACM Sigmod Record*, 48(2):18–27, 2019.

Lahari Poddar, Wynne Hsu, Mong Li Lee, and Shruti Subramaniyam. Predicting stances in twitter conversations for detecting veracity of rumors: A neural approach. In *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 65–72. IEEE, 2018.

Dean Pomerleau and Delip Rao. Fake News Challenge (2017). http://www.fakenewschallenge.org/, 2017.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. Credibility assessment of textual claims on the web. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 2173–2178, 2016.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. Where the truth lies: Explaining the credibility of emerging claims on the web and social media. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1003–1012. International World Wide Web Conferences Steering Committee, 2017.

Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. Clickbait detection. In *European Conference on Information Retrieval*, pages 810–817. Springer, 2016.

Martin Potthast, Tim Gollub, Matthias Hagen, and Benno Stein. The clickbait challenge 2017: Towards a regression model for clickbait strength. In *In Proceedings of the Clickbait Challenge*, 2017a.

Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. A stylometric inquiry into hyperpartisan and fake news. *arXiv preprint arXiv:1702.05638*, 2017b.

Martin Potthast, Tim Gollub, Kristof Komlossy, Sebastian Schuster, Matti Wiegmann, Erika Patricia Garces Fernandez, Matthias Hagen, and Benno Stein. Crowdsourcing a large corpus of clickbait on twitter. In *Proceedings of the 27th international conference on computational linguistics*, pages 1498–1507, 2018.

Yujia Qin, Yankai Lin, Jing Yi, Jiajie Zhang, Xu Han, Zhengyan Zhang, Yusheng Su, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. Knowledge inheritance for pre-trained language models. *ArXiv*, abs/2105.13880, 2021.

Dragomir R Radev, Eduard Hovy, and Kathleen McKeown. Introduction to the special issue on summarization. *Computational linguistics*, 28(4):399–408, 2002.

Dragomir R Radev, Hongyan Jing, Małgorzata Styś, and Daniel Tam. Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6): 919–938, 2004.

Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2931–2937, 2017.

Benjamin Riedel, Isabelle Augenstein, Georgios P Spithourakis, and Sebastian Riedel. A simple but tough-to-beat baseline for the fake news challenge stance detection task. *arXiv preprint arXiv:1707.03264*, 2017.

Georgios Rizos, Konstantin Hemker, and Björn Schuller. Augment to prevent: short-text data augmentation in deep learning for hate-speech classification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 991–1000, 2019.

Gaetano Rossiello, Pierpaolo Basile, and Giovanni Semeraro. Centroid-based text summarization through compositionality of word embeddings. In *Proceedings of the MultiLing 2017 Workshop on Summarization and Summary Evaluation Across Source Types and Genres*, pages 12–21, 2017.

Natali Ruchansky, Sungyong Seo, and Yan Liu. Csi: A hybrid deep model for fake news detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 797–806, 2017.

Horacio Saggion and Thierry Poibeau. Automatic text summarization: Past, present and future. In *Multi-source, multilingual information extraction and summarization*, pages 3–21. Springer, 2013.

Tanik Saikh, Amit Anand, Asif Ekbal, and Pushpak Bhattacharyya. A novel approach towards fake news detection: deep learning augmented with textual entailment features. In *International Conference on Applications of Natural Language to Information Systems*, pages 345–358. Springer, 2019.

Mehdi Samadi, Partha Pratim Talukdar, Manuela M Veloso, and Manuel Blum. Claimeval: Integrated and flexible framework for claim evaluation using credibility of sources. In *AAAI*, pages 222–228, 2016.

Estela Saquete, David Tomas, Paloma Moreda, Patricio Martinez-Barco, and Manuel Palomar. Fighting post-truth using natural language processing: A review and open challenges. *Expert Systems with Applications*, 141:112943, 2020.

Timo Schick and Hinrich Schütze. Exploiting cloze questions for few-shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*, 2020.

Michael Schudson. *Discovering the news: A social history of American newspapers*. Basic Books, 1981.

Karishma Sharma, Feng Qian, He Jiang, Natali Ruchansky, Ming Zhang, and Yan Liu. Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(3):1–42, 2019.

Baoxu Shi and Tim Weninger. Discriminative predicate path mining for fact checking in knowledge graphs. *Knowledge-based systems*, 104:123–133, 2016.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 2017.

Umme Aymun Siddiqua, Abu Nowshed Chy, and Masaki Aono. Tweet stance detection using an attention based neural ensemble model. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1868–1873, 2019.

Craig Silverman. Emergent: A real-time rumor tracker. http://www.emergent.info/, 2015.

Vikram Singh, Sunny Narayan, Md Shad Akhtar, Asif Ekbal, and Pushpak Bhattacharyya. Iitp at semeval-2017 task 8: A supervised approach for rumour evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 497–501, 2017.

Dhanya Sridhar, James Foulds, Bert Huang, Lise Getoor, and Marilyn Walker. Joint models of disagreement and stance in online debate. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 116–125, 2015.

Ankit Srivastava, Georg Rehm, and Julian Moreno Schneider. Dfki-dkt at semeval-2017 task 8: Rumour detection and classification using cascading heuristics. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 486–490, 2017.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Jacopo Staiano and Marco Guerini. Depechemood: a lexicon for emotion analysis from crowd-annotated news. *arXiv preprint arXiv:1405.1605*, 2014.

Lukas Stappen, Fabian Brunn, and Björn Schuller. Cross-lingual zero-and few-shot hate speech detection utilising frozen transformer language models and axel. *arXiv preprint arXiv:2004.13850*, 2020.

Shiliang Sun, Chen Luo, and Junyu Chen. A review of natural language processing techniques for opinion mining systems. *Information fusion*, 36:10–25, 2017.

Angel Suseelan, S Rajalakshmi, B Logesh, S Harshini, B Geetika, S Dyaneswaran, S Milton Rajendram, and TT Mirnalinee. Techssn at semeval-2019 task 6: Identifying and categorizing offensive language in tweets using deep neural networks. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 753–758, 2019.

Noha S Tawfik and Marco R Spruit. Towards recognition of textual entailment in the biomedical domain. In *International Conference on Applications of Natural Language to Information Systems*, pages 368–375. Springer, 2019.

Matt Thomas, Bo Pang, and Lillian Lee. Get out the vote: Determining support or opposition from congressional floor-debate transcripts. *arXiv preprint cs/0607062*, 2006.

James Thorne and Andreas Vlachos. An extensible framework for verification of numerical claims. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 37–40, 2017.

James Thorne and Andreas Vlachos. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3346–3359, 2018.

James Thorne, Mingjie Chen, Giorgos Myrianthous, Jiashu Pu, Xiaoxuan Wang, and Andreas Vlachos. Fake news stance detection using stacked ensemble of classifiers. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 80–83, 2017.

Antonela Tommasel, Juan Manuel Rodriguez, and Daniela Godoy. Textual aggression detection through deep learning. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 177–187, 2018.

Fatemeh Torabi Asr and Maite Taboada. Big data and quality data for fake news and misinformation detection. *Big Data & Society*, 6(1):2053951719843310, 2019.

Sayan Unankard, Xue Li, and Mohamed A Sharaf. Emerging event detection in social networks with location sensitivity. *World Wide Web*, 18(5):1393–1417, 2015.

Slavena Vasileva, Pepa Atanasova, Lluís Màrquez, Alberto Barrón-Cedeño, and Preslav Nakov. It takes nine to smell a rat: Neural multi-task learning for check-worthiness prediction. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1229–1239, 2019.

Amir Pouran Ben Veyseh, Javid Ebrahimi, Dejing Dou, and Daniel Lowd. A temporal attentional model for rumor stance classification. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 2335–2338, 2017.

Andreas Vlachos and Sebastian Riedel. Fact checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 18–22, 2014.

Andreas Vlachos and Sebastian Riedel. Identification and verification of simple claims about statistical properties. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2596–2601. Association for Computational Linguistics, 2015.

Chenguang Wang, Yangqiu Song, Haoran Li, Yizhou Sun, Ming Zhang, and Jiawei Han. Distant meta-path similarities for text-based heterogeneous information networks. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1629–1638, 2017a.

Feixiang Wang, Man Lan, and Yuanbin Wu. Ecnu at semeval-2017 task 8: Rumour evaluation using effective features and supervised ensemble models. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 491–496, 2017b.

Liqiang Wang, Yafang Wang, Gerard De Melo, and Gerhard Weikum. Understanding archetypes of fake news via fine-grained classification. *Social Network Analysis and Mining*, 9(1):37, 2019a.

Sinong Wang, Han Fang, Madian Khabsa, Hanzi Mao, and Hao Ma. Entailment as few-shot learner. *arXiv preprint arXiv:2104.14690*, 2021.

William Yang Wang. "liar, liar pants on fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*, 2017.

William Yang Wang and Diyi Yang. That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using# petpeeve tweets. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 2557–2563, 2015.

Xingyou Wang, Weijie Jiang, and Zhiyong Luo. Combination of convolutional and recurrent neural network for sentiment analysis of short texts. In *Proceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers*, pages 2428–2437, 2016.

Xuezhi Wang, Cong Yu, Simon Baumgartner, and Flip Korn. Relevant document discovery for fact-checking articles. In *Companion Proceedings of the The Web Conference 2018*, pages 525–533, 2018.

Yangqian Wang, Hao Han, Ye Ding, Xuan Wang, and Qing Liao. Learning contextual features with multi-head self-attention for fake news detection. In *International Conference on Cognitive Computing*, pages 132–142. Springer, 2019b.

Jason Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6383–6389, 2019.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of human language technology conference*

and conference on empirical methods in natural language processing, pages 347–354, 2005.

Michael Wojatzki and Torsten Zesch. ltl. uni-due at semeval-2016 task 6: Stance detection in social media using stacked classifiers. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 428–433, 2016.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

Brian Xu, Mitra Mohtarami, and James Glass. Adversarial domain adaptation for stance detection. *arXiv preprint arXiv:1902.02401*, 2019.

Leiming Yan, Yuhui Zheng, and Jie Cao. Few-shot learning for short text classification. *Multimedia Tools and Applications*, 77(22):29799–29810, 2018.

Shenghui Yang, Haomin Zhang, et al. Comparison of several data mining methods in credit card default prediction. *Intelligent Information Management*, 10(05):115, 2018.

Jin-ge Yao, Xiaojun Wan, and Jianguo Xiao. Recent advances in document summarization. *Knowledge and Information Systems*, 53(2):297–336, 2017.

Wenpeng Yin, Katharina Kann, Mo Yu, and Hinrich Schütze. Comparative study of cnn and rnn for natural language processing. *arXiv preprint arXiv:1702.01923*, 2017.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. Qanet: Combining local convolution with global self-attention for reading comprehension. In *International Conference on Learning Representations*, 2018.

Dongxu Zhang and Zhichao Yang. Word embedding perturbation for sentence classification. *arXiv preprint arXiv:1804.08166*, 2018.

Jingqing Zhang, Piyawat Lertvittayakumjorn, and Yike Guo. Integrating semantic knowledge to tackle zero-shot text classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1031–1040, 2019a.

Qiang Zhang, Shangsong Liang, Aldo Lipani, Zhaochun Ren, and Emine Yilmaz. From stances' imbalance to their hierarchicalrepresentation and detection. In *The World Wide Web Conference*, pages 2323–2332, 2019b.

Xiang Zhang and Yann LeCun. Text understanding from scratch. *arXiv preprint arXiv:1502.01710*, 2015a.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657, 2015b.

Xichen Zhang and Ali A Ghorbani. An overview of online fake news: Characterization, detection, and discussion. *Information Processing & Management*, 57(2):102025, 2020.

Zhihua Zhang and Man Lan. Ecnu at semeval 2016 task 6: relevant or not? supportive or not? a two-step learning system for automatic detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 451–457, 2016.

Ziqi Zhang and Lei Luo. Hate speech detection: A solved problem? the challenging case of long tail on twitter. *Semantic Web*, 10(5):925–945, 2019.

Jiang Zhao, Man Lan, Zheng-Yu Niu, and Yue Lu. Integrating word embeddings and traditional nlp features to measure textual entailment and semantic relatedness of sentence pairs. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2015.

Shi Zhi, Yicheng Sun, Jiayi Liu, Chao Zhang, and Jiawei Han. Claimverif: a real-time claim verification system using the web and fact databases. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 2555–2558, 2017.

Yiwei Zhou, Alexandra I Cristea, and Lei Shi. Connecting targets to tweets: Semantic attention-based model for target-specific stance detection. In *International Conference on Web Information Systems Engineering*, pages 18–32. Springer, 2017.

Arkaitz Zubiaga, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Peter Tolmie. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PloS one*, 11(3):e0150989, 2016a.

Arkaitz Zubiaga, Elena Kochkina, Maria Liakata, Rob Procter, and Michal Lukasik. Stance classification in rumours as a sequential task exploiting the tree structure of social media conversations. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2438–2448, 2016b.

Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. Detection and resolution of rumours in social media: A survey. *ACM Computing Surveys (CSUR)*, 51(2):1–36, 2018a.

Arkaitz Zubiaga, Elena Kochkina, Maria Liakata, Rob Procter, Michal Lukasik, Kalina Bontcheva, Trevor Cohn, and Isabelle Augenstein. Discourse-aware rumour stance classification in social media using sequential classifiers. *Information Processing & Management*, 54(2):273–290, 2018b.