Universitat Politècnica de València

Departamento de Sistemas Informáticos y Computación

Tesis de Doctorado en Informática

Bilal Ghanem

On the Detection of False Information: From Rumors to Fake News

Directores de Tesis
Paolo Rosso
Universitat Politècnica de València, Spain
Francisco Rangel
Symanto Research, Germany

July, 2020

Abstract

In the recent years, the development of social media and online news agencies has brought several challenges and threats to the Web. These threats have taken the attention of the Natural Language Processing (NLP) research community as they are polluting the online social media platforms. One of the examples of these threats is false information, in which false, inaccurate, or deceptive information is spread and shared by online users. False information is not limited to verifiable information, but it also involves information that is used for harmful purposes. Also, one of the challenges that researchers have to face is the massive number of users in social media platforms, where detecting false information spreaders is not an easy job.

Previous work that has been proposed for limiting or studying the issue of detecting false information has focused on understanding the language of false information from a linguistic perspective. In the case of verifiable information, approaches have been proposed in a monolingual setting. Moreover, detecting the sources or the spreaders of false information in social media has not been investigated much.

In this thesis we study false information from several aspects. First, since previous work focused on studying false information in a monolingual setting, in this thesis we study false information in a cross-lingual one. We propose different cross-lingual approaches and we compare them to a set of monolingual baselines. Also, we provide systematic studies for the evaluation results of our approaches for better understanding. Second, we noticed that the role of affective information was not investigated in depth. Therefore, the second part of our research work studies the role of the affective information in false information and shows how the authors of false content use it to manipulate the reader. Here, we investigate several types of false information to understand the correlation between affective information and each type (Propaganda, Hoax, Clickbait, Rumor, and Satire). Last but not least, in an attempt to limit its spread, we also address the problem of detecting false information spreaders in social media. In this research direction, we focus on exploiting several text-based features extracted from the online profile messages of those spreaders. We study different feature sets that can have the potential to help to identify false information spreaders from fact checkers.

Resumen

En tiempos recientes, el desarrollo de las redes sociales y de las agencias de noticias han traído nuevos retos y amenazas a la web. Estas amenazas han llamado la atención de la comunidad investigadora en Procesamiento del Lenguaje Natural (PLN) ya que están contaminando las plataformas de redes sociales. Un ejemplo de amenaza serían las noticias falsas, en las que los usuarios difunden y comparten información falsa, inexacta o engañosa. La información falsa no se limita a la información verificable, sino que también incluye información que se utiliza con fines nocivos. Además, uno de los desafíos a los que se enfrentan los investigadores es la gran cantidad de usuarios en las plataformas de redes sociales, donde detectar a los difusores de información falsa no es tarea fácil.

Los trabajos previos que se han propuesto para limitar o estudiar el tema de la detección de información falsa se han centrado en comprender el lenguaje de la información falsa desde una perspectiva lingüística. En el caso de información verificable, estos enfoques se han propuesto en un entorno monolingüe. Además, apenas se ha investigado la detección de las fuentes o los difusores de información falsa en las redes sociales.

En esta tesis estudiamos la información falsa desde varias perspectivas. En primer lugar, dado que los trabajos anteriores se centraron en el estudio de la información falsa en un entorno monolingüe, en esta tesis estudiamos la información falsa en un entorno multilingüe. Proponemos diferentes enfoques multilingües y los comparamos con un conjunto de baselines monolingües. Además, proporcionamos estudios sistemáticos para los resultados de la evaluación de nuestros enfoques para una mejor comprensión. En segundo lugar, hemos notado que el papel de la información afectiva no se ha investigado en profundidad. Por lo tanto, la segunda parte de nuestro trabajo de investigación estudia el papel de la información afectiva en la información falsa y muestra cómo los autores de contenido falso la emplean para manipular al lector. Aquí, investigamos varios tipos de información falsa para comprender la correlación entre la información afectiva y cada tipo (Propaganda, Trucos / Engaños, Clickbait y Sátira).

Por último, aunque no menos importante, en un intento de limitar su propagación, también abordamos el problema de los difusores de información falsa en las redes sociales. En esta dirección de la investigación, nos enfocamos en explotar varias características basadas en texto extraídas de los mensajes de perfiles en línea de tales difusores. Estudiamos diferentes conjuntos de características que pueden tener el potencial de ayudar a discriminar entre difusores de información falsa y verificadores de hechos.

Resum

En temps recents, el desenvolupament de les xarxes socials i de les agències de notícies han portat nous reptes i amenaces a la web. Aquestes amenaces han cridat l'atenció de la comunitat investigadora en Processament de Llenguatge Natural (PLN) ja que estan contaminant les plataformes de xarxes socials. Un exemple d'amenaça serien les notícies falses, en què els usuaris difonen i comparteixen informació falsa, inexacta o enganyosa. La informació falsa no es limita a la informació verificable, sinó que també inclou informació que s'utilitza amb fins nocius. A més, un dels desafiaments als quals s'enfronten els investigadors és la gran quantitat d'usuaris en les plataformes de xarxes socials, on detectar els difusors d'informació falsa no és tasca fàcil.

Els treballs previs que s'han proposat per limitar o estudiar el tema de la detecció d'informació falsa s'han centrat en comprendre el llenguatge de la informació falsa des d'una perspectiva lingüística. En el cas d'informació verificable, aquests enfocaments s'han proposat en un entorn monolingüe. A més, gairebé no s'ha investigat la detecció de les fonts o els difusors d'informació falsa a les xarxes socials.

En aquesta tesi estudiem la informació falsa des de diverses perspectives. En primer lloc, atès que els treballs anteriors es van centrar en l'estudi de la informació falsa en un entorn monolingüe, en aquesta tesi estudiem la informació falsa en un entorn multilingüe. Proposem diferents enfocaments multilingües i els comparem amb un conjunt de baselines monolingües. A més, proporcionem estudis sistemàtics per als resultats de l'avaluació dels nostres enfocaments per a una millor comprensió. En segon lloc, hem notat que el paper de la informació afectiva no s'ha investigat en profunditat. Per tant, la segona part del nostre treball de recerca estudia el paper de la informació afectiva en la informació falsa i mostra com els autors de contingut fals l'empren per manipular el lector. Aquí, investiguem diversos tipus d'informació falsa per comprendre la correlació entre la informació afectiva i cada tipus (Propaganda, Trucs / Enganys, Clickbait i Sàtira).

Finalment, però no menys important, en un intent de limitar la seva propagació, també abordem el problema dels difusors d'informació falsa a les xarxes socials. En aquesta direcció de la investigació, ens enfoquem en explotar diverses característiques basades en text extretes dels missatges de perfils en línia de tals difusors. Estudiem diferents conjunts de característiques que poden tenir el potencial d'ajudar a discriminar entre difusors d'informació falsa i verificadors de fets.

Acknowledgments

I owe my deepest gratitude to my supervisor Paolo Rosso (El Jefe). His support, guidance and funding during my study enabled me to complete this thesis. I am truly indebted to my second supervisor Francisco Rangel (El Jefazo) who helped and guided me during these years. Also, I would like to thank their families for being nice and for their warm welcome.

I am grateful to the people who advised me during their research visits to our lab or during my internships: Manuel Montes-y-Gómez, Davide Buscaldi, Simone Paolo Ponzetto, Marc Franco-Salvador, and Yassine Benajiba; my research colleagues Simona, Gretel, Alesandra, Anastasia, and Javier; and finally the reviewers of this thesis: Leo Wanner, Damiano Spina, and Preslav Nakov.

I would like to thank Universitat Politècnica de València and its employees who helped me to solve my issues and answer my inquiries with smiles and care. Special thanks to Elsa Cubel, DSIC secretary staff, and that lovely lady in the reception who was always showing the smiley face.

Finally, I would like to thank my great friends who supported me with their love and care and were always available and supportive when in need. I am eternally indebted to my parents Amne and Hisham, for the sacrifices that they have made on my behalf. I could not have finished this thesis without their support and prayers. I would also like to thank my all siblings for their encouragement and motivation.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

1.1 Problem Description

The emergence of the World Wide Web has a key role in the development of many fields. The development flourished with the merit of knowledge sharing. This development was evident in many sectors, such as trade and industry, and reflected on the individual level. The free, easy, and unlimited access to the Web made individuals able to make use of online texts and media as a way to extend their knowledge. Based on a statistical report from *EMC.com* [67], the amount of online information is increasing with a ratio of 40% per year. The reporters expected that by the end of 2020, the amount of online information would be around 44 zettabytes (trillion gigabytes), where in 2013 it was approximately 4.4 zettabytes. Despite the vast benefit that we are receiving, this massive increase exposes us to risk. The free online information is editable by anyone and at any time. This openness became a double-edged sword. On one hand, humans take advantage of the available information to evolve in all life aspects. On the other hand, people became exposed to manipulation by others.

According to *Pew* Research Center [200], statistics showed that two-thirds of Americans use social media and online Websites as news sources. Differently from the TV and the radio, in the World Wide Web, with the advocacy of social media platforms, unverified sources can share information with others without restrictions. This openness caused thousands of cases around the world where online users were affected by unreliable information. These cases appeared because many news consumers believe what they read without critically examining the events or their sources. For example, many online false information posts led to a decrease of measles vaccination rates in Romania [114], causing in 2017 one of the worst measles outbreak in decades; more than 32 children died by a disease that was almost eradicated.

Recently, a research study carried out at the Massachusetts Institute of Technology (MIT) and published in Science [229], studied the worldwide

concern over false information and the possibility that it can influence political, economic, and social well-being. Some of the results obtained by the authors are that falsehoods diffuse significantly faster, farther, deeper and broader than the truth in all categories of information, and especially in social media. In essence, fake news has 70% higher probability to be retweeted than truthful news. The degree of novelty and the emotional reactions of recipients seemed to have a crucial role in the spreading process. Also, they showed that the effects were more pronounced for false political news than for false news about terrorism, natural disasters, science, urban legends or financial information.

The emersion of social media platforms, e.g., Twitter and Facebook, has led to the fast spread of false information among the public. The diffusion of false information has been done with the help of online users that have, in some cases, ideological agendas to spread falsehoods. In a study conducted by a group of researchers from the University of Southern California and Indiana University [223], they found that up to 15% of Twitter accounts are in fact bots rather than people. Unfortunately, due to the anonymity option that most of the social media platforms provide, most of these accounts cannot be identified easily by other users. And often, their published content is shared by others without being verified.

In this thesis, we aim to explore the problem of the detection of false information from several perspectives. We focus, in particular, on a set of false information types that are well known in social media (e.g., rumors, clickbaits, etc.) and online news articles (e.g., propaganda, hoax, etc.). We propose to address the problem of false information as a classification task to discriminate between truthful and false information. We propose several text classification approaches, focusing mainly on using affective information from text. Furthermore, we analyze false information to show the reader how it exploits the emotional information to affect her opinions towards specific events. Our research scope was not limited to false information, but it also includes studying suspicious online users that have a significant role in spreading false information.

This chapter introduces some essential concepts of false information and suspicious online users. We first define false information and describe what are its different types. Also, we define what suspicious online users are and why it is important to detect them. In the last section, we present the research questions and the contributions of this thesis.

1.2 False Information

False information is an old problem, appeared before the Web. In 1890, the journalists Joseph Pulitzer and William Hearst competed over the audience by writing false content full of exaggeration in the newspapers; this kind of

Figure 1.1: False information types.

practice was called later as *yellow journalism*. Later on, the rumors that were spreaded by those two journalists played an important role in leading US into the American-Spanish 1898 war [188]. The journalists' motivations behind spreading such false information were to sell more newspapers by putting eye-catching content that attracts the audience attentions.

The concept "false information" has been replaced by "fake news" [1] in some research work [202]. Experts recommend to use the former since "fake news" is closely associated with politics, and this association can narrow its focus [36]. False information can be categorized based on the harm intention or the interest behind. According to [242], false information is categorized into eight different types[2]. Some of these types intend to harm where others do not. Accordingly, we can classify them into two main categories - misinformation and disinformation - where misinformation considers false information that is published without the intent to harm (e.g., satire and clickbait), whereas disinformation can be seen as a specific kind of false information with the aim to mislead and harm the reader (e.g., hoax, propaganda, and rumor) [3].

In this thesis, we focus on five main types, namely: **hoax**, **propaganda**, **clickbait**, **rumor**, and **satire** (see Figure 1.1). **Propaganda** is a biased or exaggerated story spread to manipulate readers and to harm the interest of

[1] The nowadays fashionable term *fake news* is often used instead of disinformation (i.e., false harmful information).

[2] Types of false information: Fabricated, Propaganda, Conspiracy Theories, Hoax, Biased or one-sided, Rumor, Clickbait, and Satire.

[3] In this PhD thesis we use *false information* to refer to both misinformation and disinformation.

a particular party. **Hoax** is a paranoia-fueled story [186] or a story sought to convince people of a positive event which had not happened. Unlike propaganda, hoax stories do not aim to manipulate readers' opinions, but to convince them of their veracity. **Clickbait** is another type of misinformation that refers to the deliberate use of misleading headlines, thumbnails, or stories' snippets to redirect attention (for traffic attention). **Rumor** refers to stories whose truthfulness is ambiguous or never confirmed [166]. Rumors are well known in online social media platforms. Lastly, **Satire** is the only type of misinformation, where the author's main purpose is not to mislead the reader, but rather to deliver the story, for instance, in an ironic way. Satire news uses figurative language e.g. humor, irony, sarcasm, etc., to criticize people's foolishness or vices, particularly in the context of contemporary politics. In this thesis, we consider satire as an umbrella term employing frequently figurative language, precisely irony [149]. Nonfactual information can make use of irony, for instance, when satire is used in the news [244]. Although a satirical news has not the aim to mislead the reader with deceptive content but to entertain her being a sarcastic version of reality, it could be mistaken for a real news and propagated as such. Given the previous types, we define false information as inaccurate information that needs to be fact-checked.

1.2.1 Fact-Checking False Information

The huge amount of shared information online [67] made users unaware of the truthfulness of the shared data. This issue has received the attention of expert journalists to halt the propagation of rumors, where non-expert users are unable to know how accurate online information is. To this end, several fact-checking sites (i.e., Politifact[4], Leadstories[5], Snopes[6], etc.) have been created by those expert journalists. In addition to the aim of fact-checking online rumors, the other role of these sites is to raise awareness among online users to know how to reject false information.

Nonetheless, the manual verification of false online information is time-consuming and requires enormous efforts. To keep up with the scale and speed at which false information spreads, we need tools to automate this verification process. Thereupon, many fact-checking systems have been proposed to fill the gap. Previous work [171, 117, 82, 172] incorporated external evidence extracted from search engines to fact-check claims. Other work, like [232, 186, 88], trained classifiers using handcrafted features on labeled claims from the PolitiFact website without using any external knowledge. Unlike previous attempts, the work [6] studied the potential of having a fact-checking system that is optimized based on two objectives: claims ve-

[4] https://www.politifact.com/
[5] https://leadstories.com/
[6] https://www.snopes.com

racity prediction and generation of fact-checking explanation. However, the focus of previous work was on proposing monolingual fact-checking systems, without addressing the problem from a cross-lingual perspective.

1.2.2 The Role of Online Users

Although false information has existed for a long time, the existence of social media has created a proper environment for their propagation. The work [229] has emphasized this fact by showing that false information diffuses significantly faster in social media. In social media, online users play a primary role in all the phases of spreading false information by creating, publishing, and sharing them with more users. However, the sharing step is not always intentional; many users share false information after they get deceived by it.

The last 2016 US elections presented to the public online users, namely "trolls" [26], that had interfered in the elections' results. In May 2018, the democratic representatives from the US House Permanent Select Committee on Intelligence (USHPSCI) made public some findings regarding Russian interference in the 2016 elections. The investigation by the USHPSCI has presented that many Twitter accounts controlled by a Russian agency called "Internet Research Agency" (IRA) were behind the spread of a massive amount of fake news. According to [29], these accounts were using a set of advertised topics to seed discord among individuals.

The authors of [46] have classified suspicious online users based on their content spreading behaviour into three main categories: robots (automated accounts that have a strictly limited vocabulary), cyborgs (accounts exhibit human-like behavior and content through loosely structured, generic, automated messages and from borrowed content copied from other sources), and human spammers (legitimate accounts that abuse an algorithm to post a burst of almost indistinguishable tweets that may differ by a character in order to fool Twitter's spam detection protocols). In our work, we classify suspicious online users into two main types: trolls and bots. Trolls are online accounts controlled by a human, and used to distract and sow discord by posting inflammatory, digressive, and extraneous content[7]. There are different reasons behind the behaviour of the trolls. One reason is to spread their infected or false information to affect others by their ideologies, which is the case of the US 2016 elections. Another is that they are people that may suffer from depression, attention-starved, sadism, or psychopathy [30]. Bots are online accounts that are controlled by computer programs to spread advertisements of goods and services [50]. Also, they have been used simultaneously with trolls to retweet trolls' contents and to make them reach a broader audience.

[7] https://en.wikipedia.org/wiki/Internet_troll

1.3 Relevant Works

In an attempt to detect false information, a lot of research have been proposed. It is worthy to mention at this point that a lot of work in the literature did not focus on the fine-grained types of false information, and they used the terms *fake* or *unreliable* instead (e.g., fake vs real, as a binary classification task). The work in this line varies in terms of the used models, from simple handcrafted based systems to models based on deep learning [165, 186, 34, 201].

In the following subsections, we focus on presenting some of the relevant work for each false information type. Finally, we show how the literature work approached the online suspicious users.

1.3.1 Propaganda

According to [145], propaganda can be identified by its persuasive function, sizable target audience, the representation of a specific group's agenda, and the use of faulty reasoning or emotional appeal. Based on that, the authors of [53] defined 18 different types of propaganda and built a system that uses a Bidirectional Encoder Representations from Transformers (BERT) [59] to predict the type of propaganda. The results showed that their proposed system was able to detect propaganda techniques with a value of 0.23 using an edited version of an F1 score. It is worthy to mention that this work is the only one that addresses propaganda identification at sentence-level instead of article-level. The authors of [186] created a corpus of news articles that was labeled with the main false information types, including propaganda. A model based on word ngrams of length one to three has been used with a Max-Entropy classifier. The model achieved macro F1 values of 0.91 and 0.65 on in-domain and out-of-domain test sets, respectively. The work [16] experimented with a binarized version of the corpus (propaganda vs the rest of the labels) was introduced in [186], using an approach based on more than 140 different features. The proposed approach achieved an F1 value of 0.97. The work [16] was part of a project called "Propaganda Analysis" [51] that aims to help readers to detect the use of propaganda and analyze how propaganda spreads online. In addition to the mentioned work, a shared task on propaganda detection has been organized [52] using the corpus described in [53].

1.3.2 Hoax

Less work was done to approach hoaxes compared to the other false information types. The authors of [214] studied the diffusion of hoaxes and in particular, how the availability of debunking information may contain their diffusion. The work [56] proposed a hoax detection method that combines

news content and social context features. The model outperformed several baselines and achieved an accuracy value of 80.2%. On the basis of who "liked" hoxes, the author of [213] proposed two classification techniques in which one exceeds an accuracy value of 99%. All of the previous work used data extracted from Facebook. As Wikipedia became a primary source of information, the work [128] proposed a set of features to detect Wiki hoxes (e.g., length of text, text-to-markup ratio, number of prior edits, etc.) and the overall model obtained 92% value of accuracy.

1.3.3 Clickbait

Given the large amount of news online and in social media, some people use a misleading way to convince more audience to read their news by using eye-catching words in the headlines of the news articles. In [39] the authors used a large set of textual features to detect clickbaits. An interesting feature proposed in the work is modeling of the hyperbolic words, that is, words with high positive or negative sentiment, e.g., terrifying, awe-inspiring, etc. The proposed approach achieved an F1 score of 0.93. The authors of [41] examined several methods to detect clickbaits. The work investigated text features, like syntactic relations, and proposed to integrate the use of images since it is a way to interest users. In [174] the authors proposed a clickbait detector for Twitter data. The authors used features based on tweets' texts, linked URLs, and tweets' meta information like text ngrams, is retweeted, etc. The results showed that features like *Tweet starts with number* or the word *You* are very informative to discriminate clickbaits from genuine tweets. The overall result of the proposed model reached an ROC-AUC value of 0.79. Given a news article headline with its content text, in [24] the authors proposed an approach that extracts features from both the headline and the news text. The most interesting idea in the mentioned work is about the formality of the news article; the existence of an informal news text in an article is a strong indicator of having a clickbait headline. Several features to model the informality of the text have been investigated, with measures to calculate the similarity between the headlines and the news texts. The approach achieved an 0.75 F1 score for detecting clickbaits.

1.3.4 Satire

A lot of work has been done to detect satire in the news. In [110] the authors proposed a broad set of content-based features, including readability, stylistic, and psycholinguistic features. When these features are fed to a Support Vector Machine (SVM) classifier, they showed a high value of accuracy in the task of differentiating satire from real news (91% accuracy), but somewhat less for differentiating fake news from real (78% accuracy). The work also concluded that fake news in most cases are more similar to satire than to real

news. The authors of [193] examined satirical news in contrast with their legitimate news counterparts in 12 different news topics in 4 domains. They proposed a model that uses absurdity, humor, grammar, negative affect, and punctuation features. The model detected satirical news with an F1 value of 0.87. The work [239] proposed a 4-level hierarchical neural network that incorporates attention mechanism to reveal paragraph-level satirical cues. The model's performance was compared to several baselines and proved its effectiveness with an F1 value of 0.91 for detecting satirical news. The work [70] targeted Twitter ironic messages instead of news articles. The authors proposed a combination of affective and structural features to detect ironic messages on six different datasets. The proposed model outperforms the state of the art on almost all datasets. Finally, the authors of [207] studied a set of linguistic features to understand how different are truthful news comparing to satirical ones that use humorous language. The study highlighted that there are sentiment, syntactic, text cohesion differences. The authors trained a Logistic Regression classifier on the data, and they achieved an accuracy of 65%.

1.3.5 Rumors

The rumors verification task is usually combined with the stance detection task in the literature. Actually, the work on rumors verification have two main lines: stance-based and without stance. The hypothesis behind the former one is that, given a rumor message (R) from social media with its replies, R can be verified by examining the stances of its replies. For instance, when the R message has many denies or queries (questions), then R is likely a rumor. A lot of work modeled the replies' stances to infer the veracity of the rumors in social media [221, 137, 64, 169, 252]. The work [221] proved this hypothesis by proposing a BERT-based model that has two main branches: one models the stances, and the other infers rumor veracity using the texts of their replies. The results demonstrated that employing the stances in the proposed model improves the results clearly, with 6% as an average difference in terms of macro F1 score. Recently, two editions of a task on rumors verification and stance detection in rumors have been organized (RumorEval 2017 [58] and 2019 [95]). For the latter, the proposed work mainly can be divided into using deep learning techniques [75, 138] and manual features -based [228] models, in which word embeddings or handcraft features (e.g. ratio of negation terms in the comments) are used to verify the rumors.

1.3.6 Suspicious Online Users

The misbehaviour of the suspicious online accounts attracted the research community attention. Trolls post false information content that has potentially affected many people. After the 2016 US elections, Twitter released

a dataset[8] of online trolls (IRA trolls) that spreaded false rumors. Due to this event, an emerging research work on the Russian troll accounts has appeared [29, 241, 113, 94, 8]. The research studied IRA trolls from several perspectives, although most of the work focused on analyzing them instead of building a detection model. In [94] the authors studied the links' domains that were mentioned by IRA trolls and how much they overlap with other links used in tweets related to "Brexit". Besides, they compare "Left" and "Right" ideological trolls in terms of the number of retweets, number of followers, etc., and the online propaganda strategies they used. The authors of [29] analyzed the IRA campaign in both Twitter and Facebook, and they focused on the evolution of IRA paid advertisements on Facebook before and after the US presidential elections from a topic perspective, e.g., what topics IRA trolls targeted to seed discord among the public. In addition to the IRA trolls dataset, Twitter also released another dataset of thousands of accounts originated in United Arab Emirates, Egypt, and Saudi Arabia[9]. These accounts were engaged in a "multi-faceted information operation" targeting Qatar and Iran while amplifying messages supportive of the Saudi government. Another similar case also detected in which accounts from the Republic of China attempted to sow political discord in Hong Kong[10].

Online social bots have been a source of nuisance for the social media users for their suspicious behaviour in retweeting duplicated tweets or boosting advertisement tweets. In [130] the authors studied a large portion of Twitter bots collected during a study of seven months. The authors studied the behaviour of these bots, and they grouped them into a set of categories, e.g., duplicate spammers, malicious promoters, friend infiltrators. The focus of the previous works to limit the fake content was by addressing the task as a classification problem, where information spread by trolls or bots is categorized into genuine or fake. The work of [161] looked at the problem from a different perspective where a Multi-Criteria Decision Making approach is proposed, aiming to assess the credibility of spread content based on prior domain knowledge. Recently, a shared task on bots profiling in Twitter [184] has been organized at PAN-2019 Lab targeting both Spanish and English languages. The best performing system [115] for the English language achieved an accuracy of 96%. The system is based on stylistic features such as terms occurrence, tweets length, number of capitalized words, etc., and employed a Random Forest classifier.

[8] https://about.twitter.com/en_us/values/elections-integrity.html

[9] https://blog.twitter.com/en_us/topics/company/2019/info-ops-disclosure-data-september-2019.html

[10] https://blog.twitter.com/en_us/topics/company/2019/information_operations_directed_at_Hong_Kong.html

1.4 Motivation and Objectives

Social media and online news sources become the default channel for people to access information and express ideas and opinions. TV, radio, and newspapers are in the way to extinction. The most noticeable effect is the democratization of information and knowledge and the capacity of influence on the public opinion. Instead of having specific media sources that control the content of the news, and inevitably introduce biased opinions, now with social media any user can contribute in sharing and discussing news with the public. In February 2017, the Facebook platform improved its safety crisis response tool by adding a community help feature. The new feature allows users to search through categorized news posts, and connect with news providers over the platform [69]. But there are also undesired effects of this democratization of knowledge that are more and more present and relevant. One of them is the spread of false information: instead of helping other people, especially during a crisis, false information spreads horror and fear. Recently, with the emergence of Coronavirus (COVID-19), a wave of false information on COVID-19 floods social media, forcing the Indian government to issue advisories to educate people about the prevention methods [178]. Another critical issue is the "echo chamber" phenomenon. People by their nature like to be together with others that have the same opinions (e.g., social network communities). This kind of grouping creates the effect of the echo chamber whereby users with similar views believe the information they receive from members of the same community without verifying its credibility [42]. Therefore, social networks contribute, paradoxically, to the spread of false information and polarization of society, as we have recently witnessed in the last presidential elections in the US or the Brexit referendum.

The use of affective information, like emotions, improved modeling the text sentiment more accurately and produced significant improvements in NLP tasks (e.g., irony detection [70], success in books [140], misogyny detection [73], and stance detection [159]). However, there is room for improvement in the false information detection task. Most of the previous approaches did not to take into account affective information when building false information detectors, although, previous analysis showed that false information exploits emotions to affect the public sentiment.

Considering what we mentioned above, this research has the following objectives:

- To study the potential of affective information in the detection of false information and suspicious online users.
- To develop detection models for the false information types from an NLP perspective.
- To address false information detection from a cross-lingual perspective.

- To study false information spreaders in social media from several perspectives and to compare them to truthful news spreaders and fact checkers.

1.5 Research Questions

The research questions we aim to answer in this thesis are:

- **RQ1** *Can we detect false information from a cross-lingual perspective?* Several approaches have been proposed to detect false information for the English language. Other low-resource languages, such as Arabic, have not gained the same research effort as to the English language. Thus, in this thesis, we study false information from a cross-lingual perspective and investigate also the performance of cross-lingual approaches.

- **RQ2** *Can affective information help in the detection of false information?* Previous work have not investigated much the role of affective information as features to detect false information. To this end, we propose several models to detect false information by taking into account affective information. Overall, we focus on combining the affective information with deep learning models.

- **RQ3** *Can we detect spreaders of false information from a textual perspective?* It has been proved that one of the main reasons for the rapid propagation of false information is suspicious users. Thus, we study the potential of detecting false information spreaders in an attempt to prevent further dissemination.

1.6 Contributions of this Thesis

In this section, we briefly summarize the main contributions of this thesis.

We show that false information can be detected from a cross-lingual perspective. We study the importance of both handcraft features and word embeddings -based models for detecting false information to measure to what extent false information is language-dependent. Our results are encouraging and open the door to false information detection in languages that lack annotated data. We also study, for the first time, the potentials of using a cross-lingual approach to verify false information across languages. The results show that our proposed approach achieves better results than other approaches that are language-dependent.

For the affective information, we prove that taking into account emotions helps the detection of false information. We evaluated the proposed emotionally-infused neural models on several false information datasets to

compare with state of the art. Besides, we study the importance of affective features in each type of false information.

Finally, regarding the suspicious online users, we propose several models to detect them. We study the effect of many features such as emotions, personality traits, linguistic patterns, etc. Also, we show a comprehensive analysis over those users to give the readers a better understanding of their behaviour and language. We compare the performance of our proposed models to a set of baselines and approaches to evaluate them.

1.7 Structure of the Thesis

This PhD thesis is presented as a compendium of research articles which were published during the study phase of the author's PhD. Since we study false information from several perspectives, we split this thesis into 3 main parts to answer our research questions separately (see Section 1.5). Next, we briefly present the content of the parts:

Part I: False Information and Figurative Language

2) Irony Detection in a Multilingual Context.
3) UPV-UMA at CheckThat! Lab: Verifying Arabic Claims using a Cross Lingual Approach.

In this part, we present two research papers, that we have published at the ECIR international conference and at the CheckThat! lab at the CLEF conference, respectively. The main focus of this part is on using cross-lingual approaches for the detection of false information and irony. The first work proposes the first multilingual irony detection system. In this work we show that monolingual models trained separately on different languages using a multilingual word representation or text-based features can open the door to irony detection in languages that lack annotated data. In the second work, we present a cross-lingual approach we proposed for a fact-checking shared task for the Arabic language, to verify the factuality of claims. The model achieves the best results in the task, also in comparison with other monolingual approaches.

Part II: False Information and Emotions

4) UPV-28-UNITO at SemEval-2019 Task 7: Exploiting Post's Nesting and Syntax Information for Rumor Stance Classification.
5) An Emotional Analysis of False Information in Social Media and News Articles.

This part introduces two pieces of work that take into account affective information to improve the detection of false information. These two research works have been published at the SemEval workshop, and in the international TOIT journal, respectively. In the first one, we present our participation at the RumorEval shared task for rumors stance detection and verification. The analysis shows that emotions and sentiment contribute to the overall results. In the second work, we study the role of emotions in detecting different types of false information. The results show that emotions improve the detection of false information comparing to several baselines on two different datasets (Twitter and new articles). Moreover, we analyze from an emotional perspective each type of false information to give a clearer view of how false information uses emotions to deceive the reader.

Part III: False Information Spreaders

6) FacTweet: Profiling Fake News Twitter Accounts.

Some suspicious accounts in Twitter have specific agendas to spoil the reputation of organizations or individuals. This kind of accounts tries to gain the trust of the social media audience to make its news reach as many people as possible. Therefore, such type of accounts mixes fake with real news to hide their intentions. The detection of these sources is the first step towards preventing the spreading of false information. In this part, we investigate a new way for the detection of fake news spreaders in social media. Unlike previous attempts, where the sequential order of tweets in suspicious accounts' streams was discarded, here we propose an approach to detect suspicious Twitter accounts by treating post streams as a sequence of tweets' chunks. We test several semantic and dictionary-based features together with a sequential neural model.

Part IV: Summary

7) Discussion of the Results.
8) Conclusions and Future Work.

In this part, we discuss some of the results that have been obtained in the previous parts, we answer the research questions that we introduced in the introduction, and we draw the main conclusions of this thesis. Moreover, we complement our study with some further experiments in order to give additional insights. In Chapter 7, we extend our experiments for the previous parts. Precisely, for Part I, we conduct further experiments for the fact-checking task but from a monolingual perspective. We also study the issue of claims veracity together with stance detection task. Regarding Part II, we extend our experiments

for detecting false information by proposing a model that takes into account the chronological order of affective information in texts. Last but not least, we further study false information spreaders, as in Part III, but using different types of features, such as psychological traits, linguistic features, and emotions. Besides, we compare false information spreaders to news fact checkers, and we investigate some of their specific types (e.g., trolls). At the end, in Chapter 8, we list our scientific contributions that were disseminated in the form of publications and we comment on open research lines for possible future works.

Part I

False Information and Figurative Language

To verify whether the information is factual, it is important to check the veracity of a claim. In this first part we address both problems, the detection of irony and the verification of the veracity of news claims, both from a cross-lingual perspective.

In Chapter 2 we study the potential of applying cross-lingual approaches for the detection of irony messages. We target a set of languages, namely Arabic, English, and French. We compare the performance of monolingual approaches to cross-lingual ones. First, we use the monolingual systems that are based on manual features and deep learning networks to validate their performance in a monolingual setup. And then, we use them in cross-lingual configurations and study the similarities between the languages.

In Chapter 3 we present a cross-lingual approach for verifying the veracity of news claims in Arabic. The approach consists of a set of steps to retrieve and rank evidences, and apply a text-entailment process. We compare our proposed approach to a set of models in the context of a shared task, where it showed superior results comparing to a set of monolingual systems. Also, we study the causes of errors that the approach made for future improvements.

Chapter 2

Irony Detection in a Multilingual Context

Abstract. This paper proposes the first multilingual (French, English and Arabic) and multicultural (Indo-European languages vs. less culturally close languages) irony detection system. We employ both feature-based models and neural architectures using monolingual word representation. We compare the performance of these systems with state-of-the-art systems to identify their capabilities. We show that these monolingual models trained separately on different languages using multilingual word representation or text-based features can open the door to irony detection in languages that lack annotated data for irony.

Published in:

- **Ghanem, B.**, Karoui, J., Benamara, F., Rosso, P., and Moriceau, V. (2020). Irony Detection in a Multilingual Context. In Advances in Information Retrieval. ECIR 2020. Lecture Notes in Computer Science, vol 12036, (pp. 141-149). Springer, Cham. (Core A)

2.1 Motivation

Figurative language makes use of figures of speech to convey non-literal meaning [100, 7]. It encompasses a variety of phenomena, including metaphor, humor, and irony. We focus here on irony and we use it as an umbrella term that covers satire, parody and sarcasm.

Irony detection (ID) has gained relevance recently, due to its importance to extract information from texts. For example, to go beyond the literal matches of user queries, Veale enriched information retrieval with new operators to enable the non-literal retrieval of creative expressions [224]. Also, the performance of sentiment analysis systems drastically decrease when applied to ironic texts [19, 70]. Most related work concern English [106, 112] with some efforts in French [120], Portuguese [33], Italian [92], Dutch [133], Hindi [212], Spanish variants [158] and Arabic [119, 80]. Bilingual ID with one model per language has also been explored, like English-Czech [177] and English-Chinese [215], but not within a cross-lingual perspective.

In social media, such as Twitter, specific hashtags (#irony, #sarcasm) are often used as gold labels to detect irony in a supervised learning setting. Although recent studies pointed out the issue of false-alarm hashtags in self-labeled data [111], ID via hashtag filtering provides researchers positive examples with high precision. On the other hand, systems are not able to detect irony in languages where such filtering is not always possible. Multilingual prediction (either relying on machine translation or multilingual embedding methods) is a common solution to tackle under-resourced languages [23, 195]. While multilinguality has been widely investigated in information retrieval [134, 198] and several NLP tasks (e.g., sentiment analysis [12, 15] and named entity recognition [155]), no one explored it for irony.

We aim here to bridge the gap by tackling ID in tweets from both multilingual (French, English and Arabic) and multicultural perspectives (Indo-European languages whose speakers share quite the same cultural background vs. less culturally close languages). Our approach does not rely either on machine translation or parallel corpora (which are not always available), but rather builds on previous corpus-based studies that show that irony is a universal phenomenon and many languages share similar irony devices. For example, Karoui et. al [121] concluded that their multi-layer annotated schema, initially used to annotate French tweets, is portable to English and Italian, observing relatively the same tendencies in terms of irony categories and markers. Similarly, Chakhachiro [38] studies irony in English and Arabic, and shows that both languages share several similarities in the rhetorical (e.g., overstatement), grammatical (e.g., redundancy) and lexical (e.g., synonymy) usage of irony devices. The next step now is to show to what extent these observations are still valid from a computational point of view. Our contributions are:

I. *A new freely available corpus of Arabic tweets* manually annotated for irony detection[1].

II. *Monolingual ID*: We propose both feature-based models (relying on language-dependent and language-independent features) and neural models to measure to what extent ID is language-dependent.

III. *Cross-lingual ID*: We experiment using cross-lingual word representation by training on one language and testing on another one to measure how the proposed models are culture-dependent. Our results are encouraging and open the door to ID in languages that lack annotated data for irony.

2.2 Data

Arabic dataset (AR=11,225 tweets). Our starting point was the corpus built by [119] that we extended to different political issues and events related to the Middle East and Maghreb that hold during the years 2011 to 2018. Tweets were collected using a set of predefined keywords (which targeted specific political figures or events) and containing or not Arabic ironic hashtags (#سخرية, #مسخرة, #تهكم, #استهزاء) [2]. The collection process resulted in a set of 6,809 ironic tweets (I) vs. 15,509 non ironic (NI) written using standard (formal) and different Arabic language varieties: Egyptian, Gulf, Levantine, and Maghrebi dialects.

To investigate the validity of using the original tweets labels, a sample of 3,000 I and 3,000 NI was manually annotated by two Arabic native speakers which resulted in 2,636 I vs. 2,876 NI. The inter-annotator agreement using Cohen's Kappa was 0.76, while the agreement score between the annotators' labels and the original labels was 0.6. Agreements being relatively good knowing the difficulty of the task, we sampled 5,713 instances from the original unlabeled dataset to our manually labeled part. The added tweets have been manually checked to remove duplicates, very short tweets and tweets that depend on external links, images or videos to understand their meaning.

French dataset (FR=7,307 tweets). We rely on the corpus used for the DEFT 2017 French shared task on irony [19], which consists of tweets relative to a set of topics discussed in the media between 2014 and 2016 and contains topic keywords and/or French irony hashtags (#ironie, #sarcasme). Tweets have been annotated by three annotators (after removing the original labels) with a reported Cohen's Kappa of 0.69.

English dataset (EN=11,225 tweets). We use the corpus built by [177] which consists of 100,000 tweets collected using the hashtag #sarcasm. It

[1]The corpus is available at https://github.com/bilalghanem/multilingual_irony

[2]All of these words are synonyms where they mean "Irony".

was used as benchmark in several works [84, 107]. We sliced a subset of approximately 11,200 tweets to match the sizes of the other languages' datasets.

Table 2.1 shows the tweet distribution in all corpora. Across the three languages, we keep a similar number of instances for train and test sets to have fair cross-lingual experiments as well (see Section 2.4). Also, for French, we use the original dataset without any modification, keeping the same number of records for train and test to better compare with state-of-the-art results. For the class distribution (ironic vs. non ironic), we do not choose a specific ratio, but we use the resulted distribution from the random shuffling process.

Table 2.1: Tweet distribution in all corpora.

	# **Ironic**	# **Not-Ironic**	**Train**	**Test**
Ar	6,005	5,220	10,219	1,006
Fr	2,425	4,882	5,843	1,464
En	5,602	5,623	10,219	1,006

2.3 Monolingual Irony Detection

It is important to note that our aim is not to outperform state-of-the-art models in monolingual ID, but to investigate which of the monolingual architectures (neural or feature-based) can achieve comparable results with existing systems. The result can show which kind of features works better in the monolingual settings and can be used to detect irony in a multilingual setting. In addition, it can show us to what extend ID is language dependent by comparing their results to multilingual results. Two models have been built, as explained below. Prior to learning, basic preprocessing steps were performed for each language (e.g., removing foreign characters, ironic hashtags, mentions, and URLs).

Feature-based models. We used state-of-the-art features that have shown to be useful in ID: some of them are language-independent (e.g., punctuation marks, positive and negative emoticons, quotations, personal pronouns, tweet's length, named entities) while others are language-dependent relying on dedicated lexicons (e.g., negation, opinion lexicons, opposition words). Several classical machine learning classifiers were tested with several feature combinations, among them Random Forest (RF) achieved the best result with all features.

Neural model with monolingual embeddings. We used Convolutional Neural Network (CNN) whose structure is similar to the one proposed by [124]. For the embeddings, we relied on *AraVec* [209] for Arabic, Fast-

Text [99] for French, and Word2vec Google News [144] for English [3]. For the three languages, the size of the embeddings is 300 and the embeddings were fine-tuned during the training process. The CNN network was tuned with 20% of the training corpus using the *Hyperopt*[4] library.

Results. Table 2.2 shows the results obtained when using train-test configurations for each language. For English, our results, in terms of macro F-score (F), were not comparable to those of [177, 218], as we used 11% of the original dataset. For French, our scores are in line with those reported in state of the art (cf. best system in the irony shared task achieved $F = 78.3$ [19]). They outperform those obtained for Arabic ($A = 71.7$) [119] and are comparable to those recently reported in the irony detection shared task in Arabic tweets [80, 81] ($F = 84.4$). Overall, the results show that semantic-based information captured by the embedding space is more productive compared to standard surface and lexicon-based features.

Table 2.2: Results of the monolingual experiments (in percentage) in terms of accuracy (A), precision (P), recall (R), and macro F-score (F).

	Arabic				French				English			
	A	P	R	F	A	P	R	F	A	P	R	F
RF	68.0	67.0	82.0	68.0	68.5	71.7	87.3	61.0	61.2	60.0	70.0	61.0
CNN	**80.5**	79.1	84.9	**80.4**	**77.6**	68.2	59.6	**73.5**	**77.9**	74.6	84.7	**77.8**

2.4 Cross-lingual Irony Detection

We use the previous CNN architecture with bilingual embedding and the RF model with surface features (e.g., use of personal pronoun, presence of interjections, emoticon or specific punctuation)[5] to verify which pair of the three languages: (a) has similar ironic pragmatic devices, and (b) uses similar text-based pattern in the narrative of the ironic tweets. As continuous word embedding spaces exhibit similar structures across (even distant) languages [143], we use a multilingual word representation which aims to learn a linear mapping from a source to a target embedding space. Many methods have been proposed to learn this mapping such as parallel data supervision and bilingual dictionaries [143] or unsupervised methods relying on monolingual corpora [48, 5, 230]. For our experiments, we use Conneau et al 's approach as it showed superior results with respect to the literature [48]. We perform several experiments by training on one language ($lang_1$) and testing on another one ($lang_2$) (henceforth $lang_1 \rightarrow lang_2$). We get 6 configurations, plus two others to evaluate how irony devices are expressed cross-culturally, i.e.

[3] Other available pretrained embeddings models have also been tested.

[4] https://github.com/hyperopt/hyperopt

[5] To avoid language dependencies, we rely on surface features only discarding those that require external semantic resources or morpho-syntactic parsing.

in European vs. non European languages. In each experiment, we took 20% from the training to validate the model before the testing process. Table 2.3 presents the results.

Table 2.3: Results of the cross-lingual experiments.

	CNN				RF			
Train→Test	A	P	R	F	A	P	R	F
Ar→Fr	60.1	37.2	26.6	**51.7**	47.03	29.9	43.9	46.0
Fr→Ar	57.8	62.9	45.7	**57.3**	51.11	61.1	24.0	54.0
Ar→En	48.5	26.5	17.9	34.1	49.67	49.7	66.2	**50.0**
En→Ar	56.7	57.7	62.3	**56.4**	52.5	58.6	38.5	53.0
Fr→En	53.0	67.9	11.0	42.9	52.38	52.0	63.6	**52.0**
En→Fr	56.7	33.5	29.5	50.0	56.44	74.6	52.7	**58.0**
(En/Fr)→Ar	62.4	66.1	56.8	**62.4**	55.08	56.7	68.5	62.0
Ar→(En/Fr)	56.3	33.9	09.5	42.7	59.84	60.0	98.7	**74.6**

From a semantic perspective, despite the language and the cultural differences between Arabic and French languages, CNN results show high performance comparing to the other language pairs when we train on each of these two languages and test on the other one. Similarly, for the French and the English pair, but when we train on French they are quite lower. We have a similar case when we train on Arabic and test on English. We can justify that by, the language presentation of the Arabic and the French tweets are quite informal and have many dialect words that may not exist in the pretrained embeddings we used comparing to the English ones (lower embeddings coverage ratio), which become harder for the CNN to learn a clear semantic pattern. Another point is the presence of Arabic dialects, where some dialect words may not exist in the multilingual pretrained embedding model that we used. On the other hand, from the text-based perspective, the results show that the text-based features can help in the case when the semantic aspect shows weak detection; this is the case for the $Ar \longrightarrow En$ configuration. It is worthy to mention that the highest result we get in this experiment is from the En→Fr pair, as both languages use Latin characters. Finally, when investigating the relatedness between European vs. non European languages (cf. (En/Fr)→Ar), we obtain similar results to those obtained in the monolingual experiment (macro F-score 62.4 vs. 68.0) and best results are achieved by Ar →(En/Fr). This shows that there are pragmatic devices in common between both sides and, in a similar way, similar text-based patterns in the narrative way of the ironic tweets.

2.5 Discussions and Conclusion

This paper proposes the first multilingual ID in tweets. We show that simple monolingual architectures (either neural or feature-based) trained sep-

arately on each language can be successfully used in a multilingual setting providing a cross-lingual word representation or basic surface features. Our monolingual results are comparable to the state of the art for the three languages. The CNN architecture trained on cross-lingual word representation shows that irony has a certain similarity between the languages we targeted despite the cultural differences, which confirms that irony is a universal phenomenon, as already shown in previous linguistic studies [205, 121, 47]. The manual analysis of the common misclassified tweets across the languages in the multilingual setup, shows that classification errors are due to three main factors. (1) First, the *absence of context* where writers did not provide sufficient information to capture the ironic sense even in the monolingual setting, as in نبدا تاني يسقط يسقط حسني مبارك !! (*Let's start again, get off get off Mubarak!!*), where the writer mocks the Egyptian revolution, as the actual president "Sisi" is viewed as Mubarak's fellows. (2) Second, the presence of *out of vocabulary (OOV) terms* because of the weak coverage of the mutlilingual embeddings, which make the system fails to generalize when the OOV set of unseen words is large during the training process. We found tweets in all the three languages written in a very informal way, where some characters of the words were deleted, duplicated or written phonetically (e.g., *phat* instead of *fat*). (3) Another important issue is the difficulty to *deal with the Arabic language*. Arabic tweets are often characterized by non-diacritised texts, a large variations of unstandardized dialectal Arabic (recall that our dataset has 4 main varieties, namely Egypt, Gulf, Levantine, and Maghrebi), presence of transliterated words (e.g., the word *table* becomes طابلة (*tabla*)), and finally linguistic code switching between Modern Standard Arabic and several dialects, and between Arabic and other languages like English and French. We found some tweets contain only words from one of the varieties and most of these words do not exist in the Arabic embeddings model. For example in مبارك بقاله كام يوم مامتش .. هو عيان ولاه ايه #مصر (*Since many days Mubarak didn't die .. is he sick or what? #Egypt*), only the words يوم (day), مبارك (Mubarak), and هو (he) exist in the embeddings. Clearly, considering only these three available words, we are not able to understand the context or the ironic meaning of the tweet.

To conclude, our multilingual experiments confirmed that the door is open for multilingual approaches for ID. Furthermore, our results showed that ID can be applied to languages that lack annotated data. Our next step is to experiment with other languages such as Hindi and Italian.

Chapter 3

UPV-UMA at CheckThat! Lab: Verifying Arabic Claims using a Cross Lingual Approach

Abstract. In this paper we present our participation at CheckThat!-2019 Lab - Task 2 on Arabic claim verification. We propose a cross-lingual approach to detect the factuality of claims using three main steps, evidence retrieval, evidence ranking, and textual entailment. Our approach achieves the best performance in subtask-D, with a value of 0.62 as F1.

Published in:

- **Ghanem, B.**, Glavaš, G., Giachanou, A., Ponzetto, S. P., Rosso, P., and Rangel, F. (2019). UPV-UMA at CheckThat! Lab: Verifying Arabic Claims using a Cross Lingual Approach. In: L. Cappellato, N. Ferro, D. E. Losada and H. Müller (eds.) CLEF 2019 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings. CEUR-WS.org, vol. 2380.

3.1 Introduction

Rumours in news media and political debates may shape people's believes. Public opinion can be easily manipulated and this sometimes can lead to severe consequences including harming individuals, religions, and several other victims. For example, in 2016 a man opened fire on a Washington pizzeria because of a fake claim that reported that the pizzeria was housing young children as sex slaves as part of a child abuse ring led by the presidential candidate Hillary Clinton [206]. The spread of these claims is rapid and uncontrolled, which makes their verification hard and time-consuming. Thus, automated methods have been proposed to facilitate the process of their verification.

The Arabic language has a large number of speakers around the world. However, due to the language having a limited number of Natural Language Processing (NLP) resources for the Arabic language, there is an increasing gap between this language and other languages regarding the availability of NLP systems. Recently, there have been various research attempts on NLP tasks on Arabic, such as fact-checking [151] [66], author profiling [191] [175], and irony detection [122].

In this paper, we present our participation in the CheckThat! Lab - Task 2 [104] for detecting the factuality of Arabic claims in general news topics. Our approach is based on inferring the veracity by using a Natural Language Inference (NLI) system trained on the English language to predict if an Arabic pair of sentences entail each other. To do that, we use cross-lingual embeddings.

3.2 Related Work

Previous work on claims' factuality can be roughly split into two main approaches: external sources-based, and context-based. The external-sources-based approaches pass a claim to external search engines (e.g., Google, Bing), and then they build various features from the results. Ghanem et al. [82] proposed to pass the claims to Google and Bing search engines in order to retrieve evidence and then they extracted features like similarity between the claims and the snippets, as well as the Alexa rank[1] of the retrieved links. Finally, the authors used these features to train a Random Forest classifier. A similar approach was proposed by Karadzhov et al. [117] who computed the cosine similarity between the claim and the top N results to feed these similarities into a Long-Short Term Memory (LSTM).

On the other hand, the context-based approaches use a different way of inferring the factuality. Castillo et al. [35] used text characteristics, user-based, topic-based, and tweets propagation-based features. Similarly,

[1] https://www.alexa.com/siteinfo

Mukherjee and Weikum [150] proposed a continuous conditional random field model that exploits several signals of interaction between a set of features (e.g., language of the news, source trustworthiness, and users' confidence).

3.3 Task Description

Given a set of Arabic claims with their relevant documents (web pages), the goal of the task is to predict the factuality of these claims using the provided web pages. Task 2[2] has 4 different sub-tasks, but we decided to participate in two of them, namely subtasks B and D. Subtask B aims to predict how useful is a web page with respect to a claim, and the target labels are *very useful for verification*, *useful for verification*, *not useful* or *not relevant*. Subtask D aims to find the claim's factuality (*True* or *False*). This task is organized in 2 cycles; in cycle 1 the factuality should be estimated using the provided unlabeled web pages, whereas in cycle 2 using useful web pages (very useful and useful labels). The organizers provided the web pages in a real scenario, where the participants had to retrieve the evidence and then compared it to the claim.

Regarding the task data, the organizers provided 10 Arabic claims with their correspondent web pages with a number between 26 and 50 web pages results for each claim. These web pages were provided in their original form ($HTML$ format). For the test set, the organizers provided 59 claims to be verified.

3.4 Proposed Approach

We propose an approach that consists of the following three main steps: evidence retrieval, evidence ranking, and textual entailment. Figure 6.1 shows a schematic overview of our approach.

Evidence Retrieval: In the first step, we read the content of the articles and then we split it into sentences using *comma* (,) and *dot* (.) as delimiters following the previous literature work [131]. To obtain the best recall, we retrieve the top N similar sentences to the claim using cosine similarity over character n-grams. We use n-gram of length 5 and 6; we choose them experimentally. In addition, we tried to retrieve the most similar sentences using Named Entities (NEs), but we found that there are some sentences without named entities, like:

تخفف المشروبات الساخنة من نزلات البرد وتقلل من أعراضها

[2] https://sites.google.com/view/clef2019-checkthat/task-2-evidence-factuality

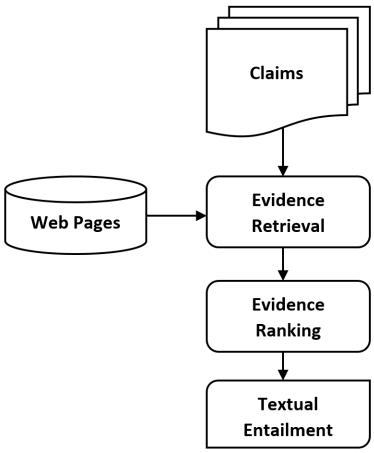


Figure 3.1: Overview of our approach.

Translation: *Cold drinks reduce colds and their symptoms*

In this step, we discard very short sentences[3]. Finally, we pass the top 20 sentences to the next step.

Evidence Ranking: For this step, we rank the top 20 sentences using word embeddings. For each claim-evidence pair, we measure their similarity and we rank the evidence based on the similarity values. For the word embeddings, we use Arabic $fastText$[4] pretrained model. We explore the following three different similarity techniques:

I. *Cosine over embeddings:* We calculate the average of the words embeddings of each sentence, and compute the cosine similarity.

II. *Cosine over weighted embeddings:* We calculate the average of the words' embeddings weighted by the Term Frequency Inverse Document Frequency (TF-IDF) weighting scheme, and then we compute the cosine similarity on the two weighted sentences' vectors. We compute the TF-IDF weights using the Comparable Wikipedia Corpus [196].

III. *DynaMax:* It is an unsupervised and non-parametric similarity measure based on fuzzy theory that dynamically extracts good features from the word embeddings depending on the sentence pair [246].

Since the training dataset is very small, it was not possible to find the best similarity technique statistically. Thus, we decided to manually investigate the ranked sentences and we found that using *DynaMax* we get the most semantically similar evidence sentences at the top ranks.

[3]We discarded sentences that have less than 35 characters. This kind of sentences appeared when a dot and a comma occur closely.

[4]https://fasttext.cc/docs/en/crawl-vectors.html

Textual Entailment: For this step, we propose to train a system on par with state-of-the-art results in NLI task, that is the Enhanced Sequential Inference Model (ESIM) [40]. We follow the implementation details of [236]. We train the ESIM on a large NLI corpus for English, namely MultiNLI [236]. Since the claims' language is Arabic, we first project the Arabic word embeddings to the vectors space of the English word embeddings[5] we used during the training of the ESIM model. To this end, we learn a linear projection matrix by solving the Procrustes problem [208, 93] using 5K automatically obtained English-Arabic word translations as supervision[6]. To evaluate the performance of our model, we use a multilingual XNLI corpus [49] created by translating development and test sets of the MultiNLI corpus. Our cross-lingually transferred ESIM system achieved 58% accuracy on the Arabic test set of the XNLI corpus.

In this step of our approach, we receive a claim with its 20 ranked sentences from the *Evidence Ranking* step. We feed the claim with each ranked sentence to the ESIM model and we estimate their prediction probabilities with respect to Entailment, Neutral, Contradiction labels. Since each claim is represented by 20 predictions, we weigh the predictions in one of two methods:

I. **Similarity Weighting**: We weigh the predictions by the evidence ranking similarity values. Given the prediction probability P of one of the classes C, we weigh it as: $P_c = \sum_{i=1}^{20} P_{ci} * SentenceSimilarity_i$.

II. **Majority Class**: Given the NLI predictions for each claim P, we extract the majority class by: $count_{classes}(\ argmax\ P)$.

Finally, after weighting the predictions for each claim, we infer the final 2-classes prediction (True, False) from the 3-classes (NLI labels) using the following rule:

$$f(P_{entailment}, P_{contradiction}) = \begin{cases} True, & \text{if } P_{entailment} \geq P_{contradiction} \\ False, & \text{otherwise} \end{cases}$$

For the Majority Class weighing method, the $P_{entailment}$ and $P_{contradiction}$ of a claim are represented by the count frequency of the class instead of its probability.

[5]We used English fastText embeddings: https://github.com/facebookresearch/fastText

[6]The 5k words obtained by translating the most frequent words appeared in an English Wikipedia corpora using Google Translator.

3.5 Experiments and Results

Task2 subtask-B: In this subtask, we use the first two steps of our approach to submit a run. In the first step, we retrieve the sentences from the web pages using character n-grams. Here, we retrieve all the sentences with a cosine similarity value greater than 0. Then, we pass them to the next step, where we rank them based on the word embeddings. At this step, we discard the ranks and we only average the sentence similarity values for each web page (WP_{avg}). Then with a rule-based method, we map the web pages averaged values into the 4 classes:

$$f(WP_{avg}) = \begin{cases} very\_useful, & \text{if } WP_{avg} \geq 0.45 \\ useful, & \text{if } WP_{avg} > 0.35 \ \& \ WP_{avg} < 0.45 \\ not\_useful, & \text{if } WP_{avg} \leq 0.35 \\ not\_relevant, & \text{if } WP_{avg} = -1 \end{cases}$$

In the cases that we do not get any sentence from the retrieval process, we set WP_{avg} to -1. The thresholds are set experimentally. Table 3.1 presents the results of the subtask-B, for both 2-classes and 4-classes prediction. Our submission for the 2-classes prediction obtains the best performance, but still lower than the provided baseline by the organizers. For the 4-classes prediction, we obtain a lower overall rank, lower than the baseline as well.

Table 3.1: The subtask-B results in terms of Accuracy, Precision, Recall, and F1 metrics.

Evaluation criteria	Acc.	Prec.	Recall	F1
2-classes prediction				
Baseline	0.57	0.30	0.72	0.42
2-classes submission	0.49	0.26	0.73	0.38
4-classes prediction				
Baseline	0.30	0.32	0.32	0.28
4-classes submission	0.24	0.3	0.29	0.23

Task2 subtask-D: For subtask-D, we use our three steps approach. For each of the two cycles (see Section on Task Description) we submit two runs, one using the *Similarity Weighting* and the other using the *Majority Class*[7].

Table 3.2 presents the results on the test set for the subtask-D. Considering the second cycle submissions' results, since they are less biased, we observe that the similarity value weighting performs better than the majority class method clearly. We obtain the best performing runs in both cycles, higher than the baselines with 0.25 F1 value on average.

[7]We submitted our runs for cycle-1 at late time, thus the organizers considered them as submissions for cycle-2.

Table 3.2: The subtask-D results in terms of Accuracy, Precision, Recall, and F1 metrics.

run #	method	Acc.	Prec.	Recall	F1
Cycle-1: Unlabeled web pages					
1	Similarity Value	0.56	0.56	0.56	0.55
2	Majority Class	0.58	0.65	0.57	0.51
-	Baseline	0.51	0.25	0.50	0.34
Cycle-2: Useful web pages					
1	Similarity Value	0.63	0.63	0.63	0.62
2	Majority Class	0.58	0.60	0.57	0.54
-	Baseline	0.51	0.25	0.50	0.34

3.6 Analysis

In our experiments we consider the first 20 sentences to be fed to the ESIM model. In Figure 3.2, we investigate the effect of varying the number of sentences to consider for each claim on the test set. We use the second cycle (given the labeled web pages) in this experiment.

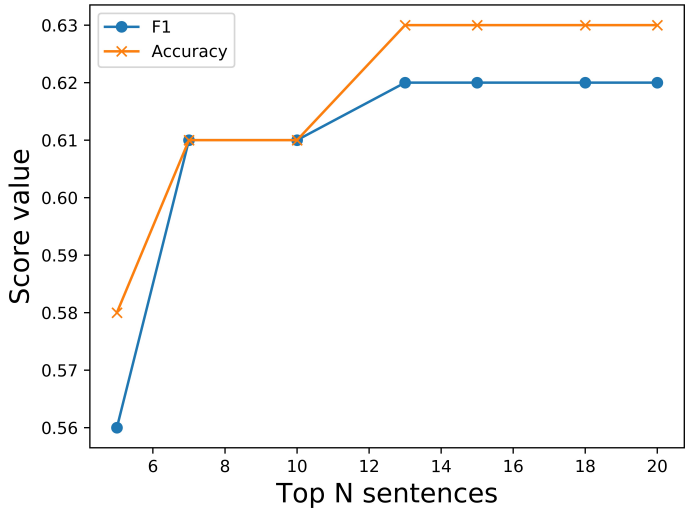
Understanding the causes of errors of our approach is important for future improvements. We manually examined the predictions to understand the causes of errors. We categorize them into the following cases:

I. **Unfamous news**: Some of the truthful claims were not covered by many news sites. We found that our approach retrieved few correct evidence (two or three evidence) while the rest of the evidence describe things related to the main entity but not regarding the same claim issue. Since in our approach we use the first 20 evidence to infer the factuality, the first 3 similar evidence, as an example, voted positively for the factuality of the sentence, where the rest 17 voted negatively. This kind of errors can be resolved by using a dynamic number of evidence sentences for each claim instead of a fixed one.

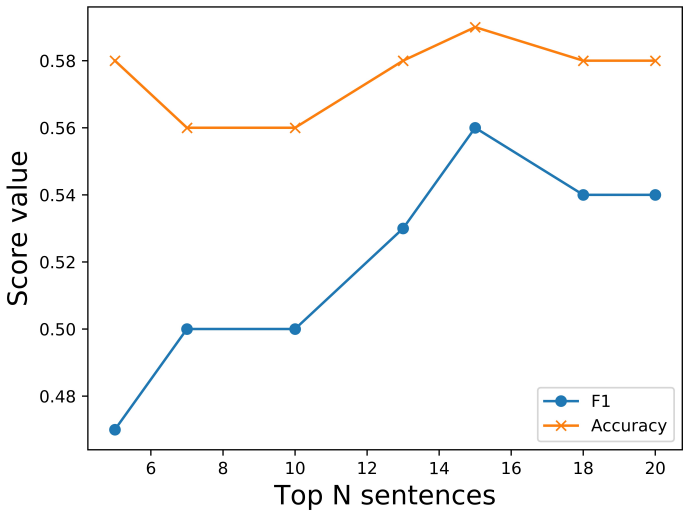
II. **The spread of false rumors**: The spread of rumors over the web can mislead people. Since our approach is based on retrieving the claim's evidence from the web, the existence of these false rumors can consequently mislead our system. As an example, given the following false claim:

توفي رفعت الأسد عم بشار الأسد في أحد مستشفيات باريس

Translation: *Rifaat al-Assad, the uncle of Bashar al-Assad, died in a hospital in Paris*



(a)



(b)

Figure 3.2: The performance of our approach on the test set using (**a**) the Similarity Value weighting and (**b**) Majority Class with varying the number of evidence sentences.

Our approach retrieved the following evidence which supports the claims:

أنباء عن وفاة جزار حماة وسجن تدمر؛ رفعت الأسد في أحد مستشفيات باريس

Translation: *News about the death of the butcher of Hama and Palmyra prisons, Rifaat al-Assad in a Paris hospital*

This evidence was retrieved as a Twitter post. Considering only news agencies as source of news where random users are not allowed to post news, can prevent these errors.

III. **Inaccurate sentence segmentation**: The Arabic language has a complicated sentence structure, where using dots to split a document into sentences is inaccurate step. Following previous work in Arabic, we used *dot* (.) and *comma* (,) to split the evidence documents into sentences. We found that in some cases, the important evidence sentence in a document has a *comma* between the object and the predicate. As an example:

أعدمت مصر ١٥ متشددا أدينوا بشن هجمات نتج عنها
مقتل عدد من رجال الجيش والشرطة في شبه جزيرة سيناء

Translation: *Egypt executed 15 militants convicted of attacks that resulted in the deaths of a number of military and police men in the Sinai Peninsula*

The evidence in a document presented as follows:

نفذت مصلحة السجون رابع حكم بالإعدام فى ١٥ متهماً،
على خلفية اتهامهم بقتل ضباط وجنود القوات المسلحة فى شمال سيناء

Translation: *The Prison Service carried out a fourth death sentence in 15 accused, (**COMMA**) for killing officers and soldiers of the armed forces in Northern Sinai*

The *comma* between the sentence's parts made the evidence unsupportive to the claim by splitting it.

IV. **Weak ESIM predictions**: We found some claims whose evidence was retrieved correctly, but the ESIM model was unable to verify them. We argue that this kind of error is due to the aligned cross-lingual embedding.

3.7 Conclusion and Future Work

In this paper, we presented our participation in CheckThat! lab - Task 2 at CLEF-2019. We presented an approach that consists of 3 main steps from Arabic claims verification. Our proposed approach managed to achieve a good performance. Also, from the error analysis, the results showed that our cross-lingual model is solid since the majority of errors were due to the other previous reasons. As a future work, we plan to focus and improve the error cases we identified for more effective retrieval, ranking, and prediction.

Part II

False Information and Emotions

In this second part we investigate how emotions and sentiment can help to verify rumors and detect false information in general.

In Chapter 4 we present a system for the tasks of stance detection in rumors context and rumors verification. The proposed system consists of a diverse lexical feature set, including emotions and sentiment. In addition, another two novel techniques that exploit the structure of the social media threads are applied. The results show that the proposed features are effective. Moreover, for a more detailed understanding, we present features and error analysis.

In Chapter 5 we study false information from an emotional perspective. We focus on four main types of false information: propaganda, hoax, clickbait, and satire. Our study targets false information in social media and online news articles. We propose a deep learning system that takes advantage of emotions to detect false information types. Finally, we conduct a comprehensive analysis of the false information to understand the role of emotions in each type of it.

Chapter 4

UPV-28-UNITO at SemEval-2019 Task 7: Exploiting Post's Nesting and Syntax Information for Rumor Stance Classification

Abstract. In the present paper we describe the UPV-28-UNITO system's submission to the RumorEval 2019 shared task. The approach we applied for addressing both the subtasks of the contest exploits both classical machine learning algorithms and word embeddings, and it is based on diverse groups of features: stylistic, lexical, emotional, sentiment, meta-structural and Twitter-based. A novel set of features that take advantage of the syntactic information in texts is moreover introduced in the paper.

Published in:

- **Ghanem, B.**, Cignarella, A. T., Bosco, C., Rosso, P., and Rangel, F. (2019). 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 (pp. 1125-1131).

4.1 Introduction

The problem of rumor detection lately is attracting considerable attention, also considering the very fast diffusion of information that features social media platforms. In particular rumors are facilitated by large users' communities, where also expert journalists are unable to keep up with the huge volume of online generated information and to decide whether a news is a hoax [176, 235, 251].

Rumour stance classification is the task that intends to classify the type of contribution to the rumours expressed by different posts of a same thread [179] according to a set of given categories: supporting, denying, querying or simply commenting on the rumour. For instance, referring to Twitter, once a tweet that introduces a rumour is detected (the "source tweet"), all the tweets having a reply relationship with it, (i.e., being part of the same thread), are collected to be classified.

Our participation to this task is mainly focused on the investigation of linguistic features of social media language that can be used as cues for detecting rumors[1].

4.2 Related work

The RumorEval 2019 shared task involves two tasks: Task A (rumour stance classification) and Task B (verification).

Stance Detection (SD) consists in automatically determining whether the author of a text is in favour, against, or neutral towards a given target, i.e., statement, event, person or organization, and it is generally indicated as TARGET-SPECIFIC STANCE CLASSIFICATION [146].

Another type of stance classification, more general-purpose, is the OPEN STANCE CLASSIFICATION task, usually indicated with the acronym SDQC, by referring to the four categories exploited for indicating the attitude of a message with respect to the rumour: Support (S), Deny (D), Query (Q) and Comment (C) [1]. Target-specific stance classification is especially suitable for analysis about a specific product or a political actor, being the target given as already extracted, e.g., from conversational cues. On this regard several shared tasks have been organized in recent years: see for instance SemEval-2016 Task 6 [147] considering six commonly known targets in the United States, and StanceCat at IberEval-2017 on stance and gender detection in tweets on the matter of the Independence of Catalonia [216]. On the other hand, the open stance classification, (i.e., the task addressed in this paper), is more suitable for classifying emerging news or novel contexts, such as working with online media or streaming news analysis.

[1]Source code is available on GitHub: https://github.com/bilalghanem/UPV-28-UNITO

Provided that attitudes around a claim can act as proxies for its veracity, and not only of its controversiality, it is reasonable to consider the application of SDQC techniques for accomplishing rumour analysis tasks. A first shared task, concerning SDQC applied to rumor detection, has been organized at SemEval-2017, i.e., RumorEval 2017 [58]. Furthermore, several research work have analyzed the open issue of the impact of rumors in social media [189, 253, 251], for instance exploiting linguistic features [85]. Such kind of approaches may be also found in work which deal with the problems of Fake News Detection [44, 101].

Furthermore, a rumor is defined as a "circulating story of questionable veracity, which is apparently credible but hard to verify, and produces sufficient scepticism and/or anxiety so as to motivate finding out the actual truth" [253].

Concerning veracity identification, increasingly advanced systems and annotation schemas have been developed to support the analysis of rumour veracity and misinformation in text [179, 127, 243].

4.3 Description of the Task

The RumorEval task is articulated in the following sub-tasks: **Task A** (open stance classification – SDQC) is a multi-class classification for determining whether a message is a "support", a "deny", a "query" or a "comment" wrt the original post; **Task B** (verification) is a binary classification for predicting the veracity of a given rumour into "true" or "false" and according to a confidence value in the range of 0-1.

4.3.1 Training and Test Data

The RumourEval 2019 corpus contains a total of 8,529 English posts, namely 6,702 from Twitter and 1,827 from Reddit.

The portion of data from Twitter has been built by combining the RumorEval 2017 training and development datasets [58], and includes **5,568** tweets: 325 source tweets (grouped into eight overall topics such as Charlie Hebdo attack, Ottawa shooting, Germanwings crash...), and 5,243 discussion tweets collected in their threads.
The dataset from Reddit, which has been instead newly released this year, is composed by **1,134** posts: 40 source posts and 1,094 collected in their threads.

All data have been split in training and test set with a proportion of approximately $80\% - 20\%$ (see Table 4.1).

	Training	Test
Twitter	5,568	1,066
Reddit	1,134	761
Total	6,702	1,827

Table 4.1: Training and test data distribution.

4.4 UPV-28-UNITO Submission

The approach and the feature selection we applied is the same for both tasks and is based on a set of manual features described in Section 4.4.1. We built moreover another set of features (i.e., second-level features) extracted by using the manual features together with features based on word embeddings (see Section 4.4.2 for a detailed description). For modeling the features distribution with respect to each thread, we used for task B the same features as in task A. Then, in both tasks, we fed the features to a classical machine learning classifier.

4.4.1 Manual Features

For enhancing the selection of features, we investigated the impact of diverse groups of them: emotional, sentiment, lexical, stylistic, meta-structural and Twitter-based. Furthermore, we introduced a novel set of syntax-based features.

Emotional Features - We exploited several emotional resources in order to build features for our system. Three lexica: (a) **EmoSenticNet**, a lexicon that assigns six WordNet Affect emotion labels to SenticNet concepts [173]; (b) the **NRC Emotion Lexicon**, a list of English words and their associations with eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive) [148]; and (c) **SentiSense**, an easily scalable concept-based affective lexicon for Sentiment Analysis [55]. We also exploited two tools: (d) **Empath**, a tool that can generate and validate new lexical categories on demand from a small set of seed terms [71]; and (e) **LIWC**, a text analysis dictionary that counts words in psychologically meaningful categories [163].

Sentiment Features - Our sentiment features were modeled using sentiment resources such as: (a) **SentiStrength**, a sentiment strength detection program which uses a lexical approach that exploits a list of sentiment-related terms [219]; (b) **AFINN**, a list of English words rated for valence with an integer between minus five (negative) and plus five (positive) [156]; (c) **SentiWordNet**, a lexical resource in which each WordNet synset is associated with three numerical scores, describing how objective, positive, and negative the terms contained in the synset are [68]; (d) **EffectWordNet**, a

lexicon about how opinions are expressed towards events, which have positive or negative effects on entities (+/-effect events) [43]; (e) **SenticNet**, a publicly available resource for opinion mining built exploiting Semantic Web techniques [31]; and (f) the **Hu&Liu** opinion lexicon[2].

Lexical Features - Various lexical features already explored in similar Sentiment Analysis tasks were used: (a) the presence of **Bad Sexual Words**, a list extracted from the work of Frenda et. al [73]; (b) the presence of **Cue Words** related to the following categories: *belief, denial, doubt, fake, knowledge, negation, question, report* [9]; the categories *an, asm, asf, qas, cds* of the multilingual hate lexicon with words to hurt **HurtLex** [18]; (d) the presence of **Linguistic Words** related to the categories of *assertives, bias, fatives, implicatives, hedges, linguistic words, report verbs*; (e) the presence of specific categories present in **LIWC**: *sexual, certain, cause, swear, negate, ipron, they, she, he, you, we, I.* [163].

Stylistic Features - We employed canonical stylistic features, already thoroughly explored in Sentiment Analysis tasks and already proven useful in multiple domains: (a) the number of **question marks**; (b) the count of **exclamation marks**; (c) **length** of a sentence; (d) the **uppercase ratio**; (e) the count of consecutive **characters** and **letters**[3] (f) and the presence of **URLs**.

In addition to the above-listed, common features exploited in Sentiment Analysis tasks, in this work we introduce two novel sets of features: (1) **Problem-specific features** (considering the fact that the dataset is composed by Twitter data and Reddit data) and (2) **syntactic features**.

Meta-structural features - Since training and test data are from Twitter and Reddit both, we explored meta-structural features suitable for data coming from both platforms: (a) the **count of favourites/likes**, in which we have two different value distribution (Twitter vs. Reddit), so we normalized them in a range 0-100; (b) the **creation time** of a post, encoded in seconds; (c) the **count of replies**; and (d) the **level**, i.e., the degree of "nestedness" of the post in the thread.

Twitter-only Features - Because of the duplicitous nature of the RumorEval 2019 dataset (Twitter and Reddit), some of the several features, already thoroughly used in Sentiment Analysis tasks and based on Twitter metadata, could not be used in this task[4]. As follows: (a) the presence of **hashtags**; (b) the presence of **mentions**; (c) the count of **retweets**. And also some user-based features: (d) whether the user is **verified** or not; (f)

[2] http://www.cs.uic.edu/liub/FBS

[3] We considered 2 or more consecutive characters, and 3 or more consecutive letters.

[4] For the instances from Reddit, that did not have a representation of one of the following features, the empty values have been filled with a weighted average of the values obtained by other similar instances.

the count of **followers**; (g) the count of **listed** (i.e., the number of public lists of which this user is a member of); (h) the count of **statuses**; (i) the count of **friends** (i.e., the number of users that one account is following); (l) the count of **favourites**.

Syntactic Features - In our system some feature has been also modeled by referring to syntactic information involved in texts [197]. After having parsed[5] the dataset in the *Universal Dependency*[6] format, thus obtaining a set of syntactic "dependency relations" (*deprel*), we were able to exploit: (a) the **ratio of negation** dependencies compared to all the other relations; (b) the Bag of Relations (BoR_all) considering all the *deprels* attached to **all the tokens**; (c) the Bag of Relations (BoR_list) considering all the *deprels* attached to the tokens belonging to a selected **list of words** (from the lists already made explicit in the paragraph "Lexical Features" in Section 4.4.1); and finally (d) Bag of Relations (BoR_verbs) considering all the *deprels* attached to all the **verbs**, thus fully exploiting morpho-synctactic knowledge.

4.4.2 Second-level Features

For the second-level features, we employed (a) the cosine similarity of one instance wrt its parents and (b) information about the tree structure of a thread, exploiting its "nesting" and depth from the source tweet.

Similarity with Parents - In this feature, we used the cosine similarity to measure the similarity between each post with its parents. The parents of a reply are the (A) direct upper-level post and (B) the source post in the thread (see Figure 4.1).

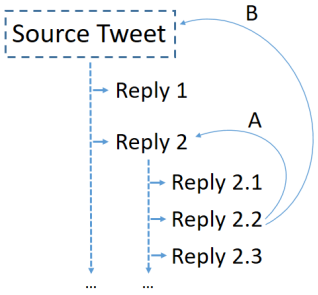


Figure 4.1: An example for reply (2.2) parents.

We extracted the cosine similarity in A and B by using the manual features' final vector and words embeddings average vectors of the posts; the

[5]The parsing system we applied is UDPipe, available at: https://pypi.org/project/ufal.udpipe/

[6]The *de facto* standard for the representation of syntactical knowledge in the NLP community: https://universaldependencies.org/

words embeddings average vector for a post is extracted by averaging the embeddings of the post's words[7].

SDQC Depth-based Clusters - We built level-based stance clusters from the posts. For each stance class (SDQC), we extracted all the belonging posts that correspond to one of the four classes and we computed the average value of the feature vectors (as one unique cluster). Since we have four main stances, this process ended with four main clusters. For the feature extraction, we measured the cosine similarity for each post wrt these four clusters. As done in the previous feature described above, we built these clusters by using both the manual features' vectors and word embeddings' vectors of the posts, so each stance cluster is represented in two ways. In these four main clusters, we did not consider the nesting of the posts in the thread.

Also, we obtained the same clusters but instead of averaging all the posts that correspond to a stance, we considered the nesting of the posts in the thread. We split the nesting of the threads into five groups: posts with depth one, two, three, four, five or larger. For each of these levels, we extracted four SDQC clusters (depth-based). For instance, if a post occurs in depth two, we measured the cosine similarity between this post and 1) the four main SDQC clusters[8], 2) the four depth-based SDQC clusters two.

Concerning task B, we modeled the distribution of the features used for task A. For each thread we did the following:

I. We counted how many posts in the thread correspond to each of the stances.

II. We extracted the averaged features' vectors for each stance's posts in the thread.

III. We extracted the standard deviation for each stance's posts in the thread.

4.5 Experiments

We tested different machine learning classifiers in each task performing 10-fold cross-validation. The results showed that the Logistic Regression (LR) produces the highest scores. For tuning the classifier, we used the Grid Search method. The parameters of the LR are: $C = 61.5$, $penalty = L2$, and since the dataset is not balanced, we used different weights for the classes

[7]We used the pre-trained Google News word embeddings in our system: https://code.google.com/archive/p/word2vec/

[8]Four features using the manual features, and another four using the words embeddings.

as COMMENT = 0.10, DENY = 0.35, SUPPORT = 0.20 and QUERY = 0.35. We conducted an ablation test on the features employed in task A in order to investigate their importance in the classification process. Table 4.2 presents the ablation test results as well as the system performance using 10-fold cross-validation.

SET	FEATURE	M-F1
A	All features	54.9
B	A - Emotional features	54.5
C	A - Sentiment features	54.7
D	A - Lexical features	53.6
E	A - Syntactic features	54.7
F	A - Stylistic features	50.1
G	A - Meta-structural features	54.5
H	A - Twitter-only features	54.9
I	A - Cosine similarity with parents	55.3
I.1	I using only manual features	54.9
I.2	I using only words embeddings	54.9
J	A - SDQC depth-based clusters	47.7
J.1	J using only manual features	53.3
J.2	J using only words embeddings	51.1
K	A-(C+E+I)	55.6
L	A-(B+C+E+G)	55.7
M	A-(B+C+E+G+I.2)	55.9

Table 4.2: Ablation test.

Provided that the organizers allowed two submissions for the final evaluation, on both tasks we used all the features (set A) in the first submission and set M for the second submission. In Table 4.3 we present the final scores achieved on both tasks.

	MACRO-F1	RMSE
Task A	48.95	–
Task B	19.96	82.64

Table 4.3: Final results.

4.6 Error Analysis

A manual error analysis allows us to see which categories and posts turned out to be the most difficult to be dealt with our system. We found out that SUPPORT was misclassified 114 times, DENY 92 times, QUERY 44 times, and COMMENT 57 times. Therefore, SUPPORT seems to be the hardest category to be correctly classified.

		PREDICTED			
		S	D	Q	C
GOLD	S	–	0	13	101
	D	1	–	6	85
	Q	5	1	–	38
	C	5	17	35	–

Table 4.4: Confusion matrix of errors.

Table 4.4 reports the detailed confusion matrix of predicted vs. gold labels and shows that the most of errors are related to the category SUPPORT (in the gold dataset) and COMMENT (in our runs), while any error involves the more contrasting classes (e.g. SUPPORT and DENY). By better investigating the gold test set, it should be moreover observed that several semantically empty messages of the test set have been marked using some class, while our system marks them as COMMENT, i.e., selecting the more frequent class when a clear indication of the content is lacking.

4.7 Conclusion

In this paper we presented an overview of the UPV-28-UNITO participation for *SemEval 2019 Task 7 - Determining Rumour Veracity and Support for Rumours*.

We submitted two different runs in the detection of rumor stance classification (Task A) and veracity classification (Task B) in English messages retrieved from Twitter and Reddit both. Our approach was based on emotional, sentiment, lexical, stylistic, meta-structural and Twitter-based features. Furthermore, we introduced two novel sets of features, i.e., *syntactical* and *depth-based* features, which proved to be successful for the task of rumor stance classification, where our system ranked as 5th (out of 26) and, according to the RMSE score, we ranked 6th in Task B for veracity classification. Since the two latter groups of features produced an interesting contribution to the score for Task A, but they were fairly neutral in Task B, we will follow this trail and try to inquire more on these aspects in our future work.

Chapter 5

An Emotional Analysis of False Information in Social Media and News Articles

Abstract. Fake news is risky since it has been created to manipulate the readers' opinions and beliefs. In this work, we compared the language of false news to the real one of real news from an emotional perspective, considering a set of false information types (propaganda, hoax, clickbait, and satire) from social media and online news articles sources. Our experiments showed that false information has different emotional patterns in each of its types, and emotions play a key role in deceiving the reader. Based on that, we proposed an LSTM neural network model that is emotionally-infused to detect false news.

Published in:

- **Ghanem, B.**, Rosso, P., and Rangel, F. (2020). An Emotional Analysis of False Information in Social Media and News Articles. ACM Transactions on Internet Technology (TOIT), 20(2), pp. 1-18. (Impact Factor: 2.382, Q2)

5.1 Introduction

With the complicated political and economic situations in many countries, some agendas are publishing suspicious news to affect public opinions regarding specific issues [157]. The spreading of this phenomenon is increasing recently with the large usage of social media and online news sources. Many anonymous accounts in social media platforms start to appear, as well as new online news agencies without presenting a clear identity of the owner. Twitter has recently detected a campaign[1] organized by agencies from two different countries to affect the results of the last U.S. presidential elections of 2016. The initial disclosures by Twitter have included 3,841 accounts. A similar attempt was done by Facebook, as they detected coordinated efforts to influence U.S. politics ahead of the 2018 midterm elections[2].

False information is categorized into 8 types[3] according to [242]. Some of these types are intentional to deceive where others are not. In this work, we are interested in analyzing 4 main types, i.e., **hoax**, **propaganda**, **clickbait**, and **satire**. These types can be classified into two main categories - misinformation and disinformation - where misinformation considers false information that is published without the intent to harm (e.g., satire and clickbait). Disinformation can be seen as a specific kind of false information with the aim to mislead and harm (e.g., hoax and propaganda). **Propagandas** are fabricated stories spread to harm the interest of a particular party. **Hoaxes** are similar to propagandas but the main aim of the writer is not to manipulate the readers' opinions, but to convince them of the validity of a paranoia-fueled story [186]. **Satire** is a type of misinformation, where the writer's main purpose is not to mislead the reader, but rather to deliver the story in an ironic way (to entertain or to be sarcastic). **Clickbait** is another type of misinformation which refers to the deliberate use of misleading headlines, thumbnails, or stories' snippets to redirect attention (for traffic attention).

The topic of fake news is gaining attention due to its risky consequences. A vast set of campaigns has been organized to tackle fake news. The owner of Wikipedia encyclopedia created the news site WikiTribune[4] to encourage the evidence-based journalism.

Another way of addressing this issue is by fact-checking websites. These websites like *politifact.com*, *snopes.com* and *factchecking.org* aim to debunk false news by manually assess the credibility of claims that have been circu-

[1] https://blog.twitter.com/official/en_us/topics/company/2018/enabling-further-research-of-information-operations-on-twitter.html

[2] https://www.businessinsider.es/facebook-coordinated-effort-influence-2018-us-midterm-elections-2018-7

[3] Types of False information: Fabricated, Propaganda, Conspiracy Theories, Hoaxes, Biased or one-sided, Rumors, Clickbait, and Satire News.

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

lated massively in online platforms. These campaigns were not limited to the English language where other languages such as Arabic have been targeted by some fact-checking organizations like *fatabyyano.net*[5].

Hypothesis Trusted news is recounting its content in a naturalistic way without attempting to affect the opinion of the reader. On the other hand, false news is taking advantage of the presented issue sensitivity to affect the readers' emotions which sequentially may affect their opinions as well. A lot of work was done previously to investigate the language of false information. The authors in [229] have studied rumours in Twitter. They have investigated a corpus of true and false tweets rumours from different aspects. From an emotional point of view, they found that false rumours inspired fear, disgust, and surprise in their replies while the true ones inspired joy and anticipation. Some kinds of false information are similar to other language phenomena. For example, satire by its definition showed similarity with irony. The work in [70] showed that affective features work well for the detection of irony. In addition, they confirmed that positive words are more relevant for identifying sarcasm and negative words for irony [231]. The results of these work motivate us to investigate the impact of emotions on false news types. These are the research questions we aim to answer:

RQ1 *Can emotional features help detect false information?*

RQ2 *Do the emotions have similar importance distributions in both Twitter and news articles sources?*

RQ3 *Which of the emotions have a statistically significant difference between false information and truthful ones?*

RQ4 *What are the top-N emotions that discriminate false information types in the two textual sources?*

In this work, we investigate suspicious news in two different sources: Twitter and online news articles. Concerning the news articles source, we focus on the beginning part of them, since they are fairly long, and the emotional analysis could be biased by their length. We believe that the beginning part of false news articles can present a unique emotional pattern for each false information type since the writer in this part is normally trying to trigger some emotions in the reader.

Throughout the emotional analysis, we go beyond the superficial analysis of words. We hope that our findings in this work will contribute to fake news detection.

Our key contributions of this chapter are:

[5]fatabyyano is an Arabic term which means "to make sure".

- **Model**: We propose an approach that combines emotional information from documents in a deep neural network. We compare the obtained results with a set of baselines. The results show that our approach is promising.

- **Analysis**: We show a comprehensive analysis on two false information datasets collected from social media and online news articles, based on a large set of emotions. We compare the differences from an affective perspective in both sources, and obtain valuable insights on how emotions can contribute to detect false news.

The rest of the chapter is structured as follows; After a brief review of related work in Section 5.2, Section 5.3 introduces our emotionally-infused model. Then, we present the evaluation framework in Section 5.4. Section 5.5 reports the experiments and the results, followed by an analysis on the false information types from emotional perspective in Section 5.6. Finally, the conclusions of this work are summarized in Section 5.7.

5.2 Related Work

The work that has been done previously on the analysis of false information is rather small regarding the approaches that were proposed. In this section, we present some recent work on the language analysis and detection of false information.

Recent attempts tried to analyze the language of false news to give a better understanding. A work done in [227] has studied the false information in Twitter from a linguistic perspective. The authors found that real tweets contain significantly fewer bias markers, hedges, subjective terms, and less harmful words. They also found that propaganda news targets morals more than satires and hoaxes but less than clickbaits. Furthermore, satirical news contains more loyalty and fewer betrayal morals compared to propaganda. In addition, they built a model that combined a set of features (graph-based, cues words, and syntax) and achieved a good performance comparing to other baselines (71% vs. 59% macro-F1). Another similar work [186] has been done to characterize the language of false information (propaganda, hoax, and satire) in online news articles. The authors have studied the language from different perspectives: the existence of weak and strong subjectivity, hedges, and the degree of dramatization using a lexicon from Wiktionary. As well, they employed in their study the LIWC dictionary to exploit the existence of personal pronouns, swear, sexual, etc. words. The results showed that false news types tend to use first and second personal pronouns more than truthful news. Moreover, the results showed that false news generally uses words to exaggerate (subjectives, superlatives, and modal adverbs), and specifically, the satire type uses more adverbs. Hoax stories tend to use

fewer superlatives and comparatives, and propagandas use relatively more assertive verbs. Moving away from these previous false information types, the work in [229] has focused on analyzing rumours in Twitter (from factuality perspective: True or False). They analyzed about 126,000 rumours and found that falsehood widespread significantly further, faster, deeper, and more broadly than truth in many domains. In addition, they found that false rumours are more novel than truthful ones, which made people more likely to share them. From an emotional perspective, they found that false rumours triggered "fear", "disgust", and "surprise" in replies while truthful ones triggered "anticipation", "sadness", "joy", and "trust". Another work [128] has studied the problem of detecting hoaxes by analyzing features related to the content in Wikipedia. The work showed that some features like hoaxes articles' length as well as the ratio of wiki markups (images, references, links to other articles and to external URLs, etc.) are important to discriminate hoaxes from legitimate articles. Many approaches have been proposed on fake news detection. In general, they are divided into social media and news claims-based approaches. The authors in [179, 245, 194, 136, 126] have proposed supervised methods using recurrent neural networks or by extracting manual features like a set of regular expressions, content-based, network-based etc. As an example, the work by [35] assessed the credibility of tweets by analyzing trending topics. They used message-based, user-based, and propagation-based features, and they found that some features related to the user information like user's age, number of followers, statuse counts etc. have helped the most to discriminate truthful from deceitful tweets. Other news claims-based approaches [82, 170, 117, 132, 172] have been mainly focusing on inferring the credibility of the claims by retrieving evidences from Google or Bing search engines. These approaches have employed a different set of features starting from manual features (e.g. cosine similarity between the claims and the results, Alexa Rank of the evidence source, etc.) to a fully automatic approach using deep learning networks. A recent trend started to appear and is trying to approach the detection of fake news from a stance perspective. The aim is to predict how other articles orient to a specific fact [85, 103, 22].

5.3 Emotionally-infused Model

In this section we describe the Emotionally-Infused Network we propose (EIN).

5.3.1 Emotional Lexicons

Several emotional models well-grounded in psychology science have been proposed, such as the ones by Magda Arnold [4], Paul Ekman [65], Robert Plutchik [168], and Gerrod Parrot [160]. On the basis of each of them,

many emotional resources (lexicons) were built in the literature. In this work, we consider several emotional resources to increase the coverage of the emotional words in texts as well to have a wider range of emotions in the analysis. Concretely, we use EmoSenticNet, EmoLex, SentiSense, LIWC and Empath:

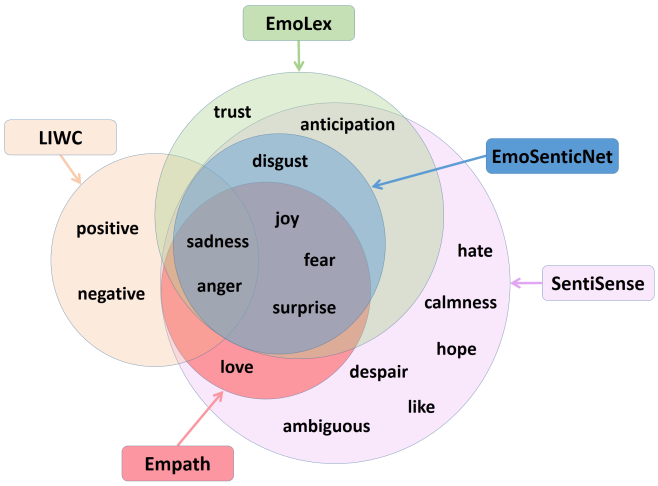


Figure 5.1: The emotional lexicons with their own emotions.

- EmoSenticNet [173] is a lexical resource that assigns WordNet-Affect[6] emotion labels to SenticNet[7] concepts. It has a total of 13,189 entries annotated using the six Ekman's basic emotions.

- EmoLex [148] is a word-emotion association lexicon that is labeled using the eight Plutchik's emotions. This lexicon contains 14,181 words.

- SentiSense [55] is a concept-based affective lexicon that attaches emotional meanings to concepts from the WordNet[8] lexical database. SentiSense has 5,496 words labeled with emotions from a set of 14 emotional categories, which is an edited version of the merge between Arnold, Plutchik, and Parrott models.

- LIWC [217] is a linguistic dictionary that contains 4,500 words categorized to analyze psycholinguistic patterns in text. Linguistic Inquiry and Word Count (LIWC) has 4 emotional categories: "sadness", "anger", "positive emotion", and "negative emotion".

[6] http://wndomains.fbk.eu/wnaffect.html
[7] https://sentic.net/
[8] https://wordnet.princeton.edu

- Empath [71] is a tool that uses deep learning and word embeddings to build a semantically meaningful lexicon for concepts. Empath uses Parrott's model for the emotional representation, but we use only the primary emotions (6 emotions) in the Pattrott's hierarchy ("love", "joy", "surprise", "anger", "sadness", "fear").

In our study we consider the 17 emotions that we shown in Figure 5.1[9].

5.3.2 Model

We choose an Long short-term memory (LSTM) [108] that takes the sequence of words as input and predicts the false information type. The input of our network is based on word embedding (content-based) and emotional features (see Figure 5.2).

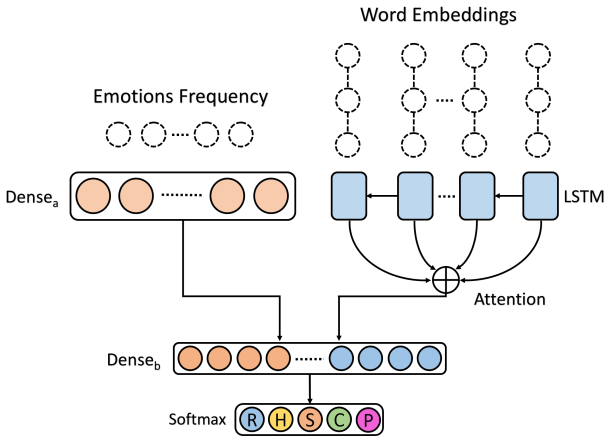


Figure 5.2: Emotionally-infused neural network architecture for false information detection. RHSCP in the Softmax layer stands for Real, Hoax, Satire, Clickbait, and Propaganda respectively.

5.3.3 Input Representation

Our network consists of two branches. In the content-based one, we use an embedding layer followed by a LSTM layer. Then, we add an attention layer [180] to make this branch focus on (highlighting) particular words over others [10]. The attention mechanism assigns a weight to each word vector result from the LSTM layer with a focus on the classification class. The input representation for this branch is represented as follows: the input sentence

[9]We investigated the performance of different combinations of lexicons; in the article we show the results obtained with the best performing combination.

[10]We tested our model without the attention layer, but we got a lower result.

S of length n is represented as $[S_1, S_2..S_n]$ where $S_n \in \mathbb{R}^d$; $\mathbb{R}^d$ is a d-dimensional word embedding vector of the i-th word in the input sentence. The output vectors of the words are passed to the LSTM layer, where the LSTM learns the hidden state h_t by capturing the previous timesteps (past features). The produced hidden state h_t at each time step is passed to the attention layer which computes a "context" vector c_t as the weighted mean of the state sequence h by:

$$c_t = \sum_{t=1}^{T} \alpha_t h_t, \tag{5.1}$$

Where T is the total number of timesteps in the input sequence and α_t is a weight computed at each time step t for each state h_t. This output vector is then concatenated with the output from the dense_a (see Figure 5.2) layer and passed to the dense_b layer, which precedes a final Softmax function to predict the output classes. Since the content-based branch is concatenated with the other emotional-based branch.
On the other hand, the input representation for the emotional-based branch is defined as follows: each lexicon is represented as L_{nm} where n is the number of emotional lexicons ($n \in [1, 5]$), and m is the number of emotions categories used depending on the emotion model (e.g. Plutchik, Arnold, etc.). In our implementation, the emotional vector of a Lexicon L_{nm} is built using word frequency and normalized by the input sentence's length. Each input sentence is represented using Equation 5.2.

$$v = L_{1m} \oplus L_{2m} \oplus L_{3m} \oplus L_{4m} \oplus L_{5m}, \tag{5.2}$$

Where $v \in \mathbb{R}^q$, $\oplus$ denotes the concatenate operation, and q is defined in Equation 5.3.

$$q = \sum_{i=1}^{n} ||L_i E_M||, \tag{5.3}$$

Next, the built emotion vector is fed to a dense layer to obtain emotion-specific representations of each input document (Equation 5.4).

$$a = f(W_a v + b_a), \tag{5.4}$$

where W_a and b_a are the corresponding weight matrix and bias terms, and f is an activation function, such as $ReLU$, $tanh$, etc.

5.4 Evaluation Framework

5.4.1 Datasets

Annotated data is a crucial source of information to analyze false information. Current status of previous work lacks available datasets of false in-

formation, where most work focus on annotating datasets from a factuality perspective. However, to analyze the existence of emotions across different sources of news, we rely on two publicly available datasets and a list contains suspicious Twitter accounts.

News Articles Our dataset source of news articles is described in [186]. This dataset was built from two different sources, for the trusted news (real news) they sampled news articles from the English Gigaword corpus. For the false news, they collected articles from seven different unreliable news sites. These news articles include satires, hoaxes, and propagandas but not clickbaits. Since we are interested also in analyzing clickbaits, we slice a sample from an available clickbait dataset called Stop_Clickbait [39] that was originally collected from two sources: Wikinews articles' headlines and other online sites that are known to publish clickbaits. The satire, hoax, and propaganda news articles are considerably long (some of them reach the length of 5,000 words). This length could affect the quality of the analysis as we mentioned before. We focus on analyzing the initial part of the article. Our intuition is that it is where emotion-bearing words will be more frequent. Therefore, we shorten long news articles into a maximum length of N words (N=300). We choose the value of N based on the length of the shortest articles. Moreover, we process the dataset by removing very short articles, redundant articles or articles that do not have a textual content[11].

Twitter For this dataset, we rely on a list of several Twitter accounts for each type of false information from [227]. This list was created based on public resources that annotated suspicious Twitter accounts. The authors in [227] have built a dataset by collecting tweets from these accounts and they made it available. For the real news, we merge this list with another 32 Twitter accounts from [118]. In this work we could not use the previous dataset[12] and we decide to collect tweets again. For each of these accounts, we collected the last M tweets posted (M=1000). By investigating these accounts manually, we found that many tweets just contain links without textual news. Therefore, to ensure of the quality of the crawled data, we chose a high value for M (also to have enough data). After the collecting process, we processed these tweets by removing duplicated, very short tweets, and tweets without textual content. Table 5.1 shows a summary for both datasets.

[11] e.g. "BUYING RATES US 31.170 .."

[12] Due to Twitter terms of usage, the authors provided in their dataset the ids of the tweets and when we tried to collect these tweets many of them were deleted.

Table 5.1: News articles and Twitter datasets' statistics.

Category	News Articles	Twitter
Satire	5,750 (18%)	12,502 (8%)
Hoax	5,750 (18%)	6,247 (4%)
Propaganda	5,750 (18%)	66,225 (43.5%)
Clickbait	5,750 (18%)	36,103 (23.5%)
Real News	8,550 (28%)	30,949 (21%)
Total	**31,550**	**152,026**

5.4.2 Baselines

Emotions have been used in many natural language processing tasks and they showed their efficiency [182]. We aim at investigating their efficiency to detect false information. In addition to EIN, we created a model (Emotion-based Model) that uses emotional features only by converting the input documents into vectors of emotions frequency (see Equation 5.2) and compare it to two baselines. Our aim is to investigate if the emotional features independently can detect false news. The two baselines of this model are Majority Class baseline (MC) and the Random selection baseline (RAN).

For the EIN model, we compare it to different baselines: **a**) The first one is bag-of-words with a support vector machine classifier (BOW-SVM). We test different classifiers, and we choose SVM since it gives the highest result in the 10-fold Cross Validation (CV); **b**) We use another baseline that is based on word embeddings where for each input document we extract an average word embedding vector by taking the mean of the embeddings for the document's words. Similarly, we test different classifiers and the Logistic Regression classifier shows the best performance (WE-LR); **c**) The last baseline is the same as our neural architecture but without the emotional features branch: an LSTM layer followed by attention and dense layers.

5.5 Experiments and Results

5.5.1 Emotion-based Model

In our experiments, we use 20% of each of the datasets for testing and we apply 10-fold cross-validation on the remain part for selecting the best classifier as well for tuning it. We tested many classifiers and we finally choose Random Forest for both datasets since it obtained the best results[13]. Table 5.2 presents the classification results on both datasets.

[13]The other classifiers that we tested are: Support vector machine (testing both kernels), naive bayes, logistic regression, k-nearest neighbor and multilayer perceptron.

Table 5.2: The results of the emotion-based model with the emotional features comparing to the baselines.

	Accuracy	Macro-Precision	Macro-Recall	Macro-F1
News Articles				
Majority Class	0.34	0.07	0.20	0.10
Random Selection	0.21	0.21	0.21	0.20
Emotion-based Model	0.50	0.48	0.51	0.48
Twitter				
Majority Class	0.44	0.09	0.20	0.12
Random Selection	0.20	0.20	0.20	0.18
Emotion-based Model	0.52	0.55	0.38	0.41

The results in both datasets show that emotional features clearly detect false news, compared to the baselines (**RQ1**). The emotional features perform better in the news articles dataset compared with these of tweets. We are interested in investigating also how good are the emotional features in detecting each class comparing to the RAN baseline. We choose the RAN baseline since it shows better results with regard to macro-F1 score. For doing so, we investigated the True Positive (TP) classification ratio for each class in each dataset.

The clickbait class shows the highest TPs comparing to the other classes. From this we can infer that clickbaits exploit emotions much more than the other classes to deceive the reader. It is worth to mention that for the hoax class the proposed approach is better than the random baselines by a small ratio (4% difference). This could be justified by the fact that hoaxes, by definition, try to convince the reader of the credibility of a false story. Hence, the writer tries to deliver the story in a normal way without allowing the reader to fall under suspicion. The number of instances related to the false information classes in the news articles dataset is the same. Therefore, there is not a majority class that the classifier can be biased to. This is not the case in the Twitter dataset. For the Twitter dataset, the dataset is not balanced. Therefore, where the results are biased by the majority class (propaganda). But in general, all the classes' TP ratios are larger than the corresponding ones obtained with RAN baseline. From these results, we can conclude that suspicious news exploits emotions with the aim to mislead the reader. Following, we present the results obtained by the proposed emotionally-infused model.

5.5.2 Emotionally-Infused Model

In the neural model, to reduce the computational costs, instead of the cross-validation process we take another 20% from the training part as a validation

Table 5.3: Models' parameters used in the three datasets (News articles, Twitter, Stop_Clickbaits). LSTM: the 3rd baseline, EIN: Emotionally-Infused Network.

Parameter	News Articles		Twitter		Stop_Clickbait	
	LSTM	EIN	LSTM	EIN	LSTM	EIN
LSTM units	140	90	180	180	120	120
Dense_a units	-	320	-	100	-	60
Dense_b units	320	60	120	60	260	120
Batch	64	64	64	64	32	32
Activation	relu	relu	relu	relu	tanh	relu
Optimizer	adadelta	adam	adadelta	rmsprop	rmsprop	Adam
Drop_c	0.5	0.5	0.5	0.2	0.2	0.2
Drop_d	0.2	0.1	0.2	0.2	0.2	0.2

set[14] (other than the 20% that is prepared for testing). For the pretrained word embeddings, we use Google News Word2Vec 300-Embeddings[15] in the neural network as well as in the W2V-LR baseline. For the classical machine learning classifiers for the baselines, we use the Scikit-Learn python library, and for the deep learning network, we use Keras library with Tensorflow as backend. To tune our deep learning network (hyper-parameters), we use the Hyperopt[16] library. And to reduce the effect of overfitting, we use early stopping technique.

In Table 5.3 we summarize the parameters with respect to each dataset. We have to mention that we use Dropout after the dense layer in the emotional features branch (Drop_c) as well as after the attention layer in the other one (Drop_d) before the concatenation process. Since it is a multiclass classification process, we use categorical cross-entropy loss function. A summary of the models' parameters is presented in Table 5.3.

Table 5.4 summarizes the performance of the proposed model in comparison to those obtained by the baselines. We report Macro- precision, recall, and F1, including also the metric of accuracy; for comparing the models' results we consider the macro of metrics since it shows an averaged result over all the classes. The baselines that we propose clearly show high results, where the LSTM baseline has the best performance in news articles dataset. In Twitter there is a different scenario, the BOW-SVM baseline shows a higher performance with respect to LSTM. We are interested in

[14] We are forced to use different validation scenarios because for selecting the best parameters in the classical machine learning Scikit-Learn library we used Grid Search technique where CV is the only option for tuning. On the other hand, it is too expensive computationally to use CV to tune a deep neural network using a large parameter space.

[15] https://code.google.com/archive/p/word2vec/

[16] https://github.com/hyperopt/hyperopt

investigating the reason behind that. Therefore, we checked the coverage ratio of the used embeddings in the Twitter dataset. We have to mention that we excluded stop words during representing the input documents using the pre-trained Google News word embeddings[17]. In the news articles dataset, we found that the coverage ratio of the embeddings is around 94% while in Twitter it is around 70%. Therefore, we tuned the word embeddings during the training process to improve the document's representation since we have a larger dataset from Twitter. This process contributed with 1.9% on the final macro-F1 results in Twitter (the result without tuning is 0.54). Even though, the results obtained with the LSTM baseline is still lower than the one obtained with BOW-SVM. This experiment gives us some intuition that the weaker performance on Twitter may be due to the embeddings. Therefore, we tried different embeddings but none of them improved the result[18]. The second baseline (W2V-LR) proved the same issue regarding the embeddings. The W2V-LR macro-F1 result in the news articles dataset is competitive, where it is much lower in Twitter. The usage of LSTM is two folds: in addition to being a good baseline, it shows also how much the emotional features contribute in the emotionally-infused network.

EIN results outperform the baselines with a large margin (around 3% in Twitter and 6% in news articles), especially in the news articles dataset. The margin between EIN and the best baseline is lower in the Twitter dataset. The results also show that combining emotional features clearly boosts the performance. We can figure out the improvement by comparing the results of EIN to LSTM. EIN shows superior results in news articles dataset with regard to the LSTM (0.79). A similar case appears in the Twitter dataset but with a lower margin (0.60). The results of EIN in Twitter dataset show that emotional features help the weak coverage of word embeddings to improve the performance as well as to overcome the BOW-SVM baseline.

Furthermore, to investigate the improvement that the emotions produced in the detection of the classes, in Table 5.5 we present the F1 score results for each class. For the news articles dataset, the results show that employing emotions in the EIN model improves the detection in all the cases, especially in real news, propaganda, and satire classes. On the other hand, for the Twitter dataset, emotions contribute especially in the case of clickbait and satire.

We observed before that clickbait TP's ratio of the news articles dataset is the highest one, and this result points out that the clickbait class is less difficult to detect specifically from an emotional perspective. Therefore, in order to assess how our model separates false information types, we employ dimensionality reduction using t-distributed Stochastic Neighbor Embedding

[17]The existence of stop words is importance to conserve the context in the LSTM network, but we got better results without them.

[18]e.g. Glove (using multiple embedding dimensions) and FastText.

Table 5.4: Results of the proposed model (EIN) vs. the baselines.

	Accuracy	Macro-Precision	Macro-Recall	Macro-F1
News Articles				
BOW+SVM	0.74	0.72	0.71	0.71
W2V+LR	0.72	0.70	0.70	0.70
LSTM	0.75	0.77	0.74	0.74
EIN	0.80	0.79	0.80	0.79
Twitter				
BOW+SVM	0.63	0.60	0.56	0.57
W2V+LR	0.53	0.49	0.35	0.36
LSTM	0.64	0.65	0.54	0.56
EIN	0.65	0.61	0.59	0.60

Table 5.5: F1 score results of the proposed model (EIN) vs. the baselines with respect to each class.

	clickbait	hoax	propaganda	realnews	satire
News Articles					
BOW+SVM	0.90	0.54	0.59	0.84	0.66
W2V+LR	0.85	0.57	0.66	0.83	0.57
LSTM	0.88	0.67	0.65	0.80	0.68
EIN	0.91	0.69	0.75	0.85	0.76
Twitter					
BOW+SVM	0.49	0.40	0.70	0.67	0.62
W2V+LR	0.32	0.07	0.66	0.45	0.32
LSTM	0.48	0.35	0.72	0.65	0.63
EIN	0.53	0.37	0.74	0.67	0.67

(T-SNE) technique [139] to project the document's representation from a high dimensional space to a 2D plane. Thus, we project the embeddings in EIN by extracting them from the outputs of Dense_{b} layer (see Figure 5.3). We extract the embeddings twice, once from a random epoch (epoch 10) at the beginning of the training phase and the other at the last epoch.

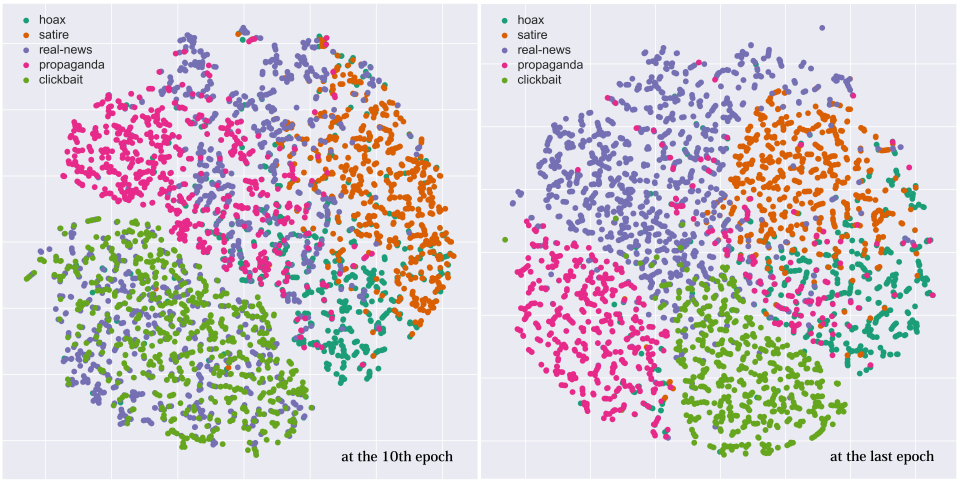


Figure 5.3: Projection of documents representation from the news articles dataset.

Our aim from the early epoch projection is to validate what we have noticed: the clickbait class is less difficult to detect compared to the other classes. As we can notice in the 10-epoch plot, the clickbait class needs few epochs to be separated from the other types, and this supports what we found previously in the manual investigation of the classes' TP ratios. Despite this clear separation, there is still an overlapping with some real-news records. This result points out that emotions in clickbaits play a key role in deceiving the reader. Also, the figure shows that the disinformation classes still need more training epochs for better separation. Real-news records are totally overlapping with the false information classes as well as the false information classes with each other. On the other hand, for the last epoch, clearly, the classes are separated from each other and the more important, from the real news. But generally, there still a small overlapping between satires and hoaxes as well few records from the propaganda class.

5.5.3 EIN as Clickbaits Detector

From the previous results in Section 5.5.1 as well as from what we notice in Figure 5.3, EIN obtains a clear separability of the clickbait class. These observations motivate us to investigate EIN as clickbait detector. Concretely, we test EIN on the source of our clickbait instances [39] in the news articles dataset. As we mentioned previously, this dataset originally was built

Table 5.6: The performance of EIN on the clickbaits dataset using 10-fold CV.

	Accuracy	Precision	Recall	F1
Stop_Clickbait	0.93	0.95	0.90	0.93
LSTM	95	0.95	0.96	0.95
EIN	0.96	0.96	0.97	0.96

using two different text sources. For clickbaits, the authors have manually identified a set of online sites that publish many clickbait articles. Whereas for the negative class, they collected headlines from a corpus of Wikinews articles collected in other research work. They took 7,500 samples from each class for the final version of the dataset. The authors also proposed a clickbaits detector model[19] that employed a combination of features: *sentence structure* (sentence length, average length of words, the ratio of the number of stop words to the number of thematic words and the longest separation between the syntactically dependent words), *word patterns* (presence of cardinal number at the beginning of the sentence, presence of unusual punctuation patterns), *clickbait language* (presence of hyperbolic words, common clickbait phrases, internet slangs and determiners), and *N-grams features* (word, Part-Of-Speech, and syntactic n-grams). Using this set of features group, the authors tested different classifiers where SVM showed the state-of-the-art results. They considered Accuracy, Precision, Recall and F1 to compare their approach to a baseline (an online web browser extension for clickbaits detection called Downworthy[20]).

In this experiment, we consider the third baseline (LSTM) to observe the improvement of the emotional features in the EIN model. Different from the previous experiments, this is a binary classification task. Therefore, we use binary cross-entropy as loss function and we change the Softmax layer to a Sigmoid function. The new parameters for both LSTM and EIN models are mentioned in Table 5.3.

In Table 5.6 we present the results of the Stop_Clickbait approach, LSTM baseline, and the EIN model. The results show that our baseline outperforms the proposed clickbait detector with a good margin. Furthermore, the results of the EIN are superior to the LSTM and the Stop_Clickbait detector. Considering emotions in the EIN deep learning approach improved the detection of false information. This is due to the fact that in clickbaits emotions are employed to deceive the reader.

[19] We use Stop_Clickbait to refer to this approach in the rest of the experiments.
[20] http://downworthy.snipe.net/

5.6 Discussion

The results show that the detection of suspicious news in Twitter is harder than detecting them in news articles. Overall, the results of EIN showed that emotional features improve the performance of our model, especially in the case of the news articles dataset. We manually inspected the Twitter dataset and observed that the language of the tweets has differences compared to the news articles one. We found that news in Twitter has many abbreviations (amp, wrt, JFK...etc.), bad words abbreviations (WTF, LMFO...etc.), informal language presentation, and typos. This reduces the coverage ratio of word embeddings. We also noticed that suspicious news in Twitter are more related to sexual issues. To validate our observations, we extracted the mean value of sexual words using a list of sexual terms [73]. The mean value is the average number of times a sexual/bad word appears in a tweet normalized by the length of the tweet. The mean value in Twitter is 0.003[21] while in news articles is 0.0024. Similarly, suspicious news in Twitter presented more insulting words[22] than in news articles where the mean value in Twitter is 0.0027 and 0.0017 in news articles.

Following, we focus on analyzing false information from an emotional perspective. We are aiming to answer the rest of the questions, **RQ2**, **RQ3**, and **RQ4**.

RQ2 *Do the emotions have similar importance distributions in both Twitter and news articles sources?*

Intuitively, the emotions contribution in the classification process is not the same, where some words could manifest the existence of specific kind of emotions rather than others. To investigate this point, we use Information Gain (IG) in order to identify the importance of emotions in discriminating between real and all the other types of false news (multiclass task) in both Twitter and news articles datasets (see Figure 5.4). Before going through the ranking of features importance, we notice that the emotions ranking shapes are very similar in both Twitter and news articles. This states that despite the fact that the language is different, both sources have similar overall emotions distribution. In other words, false news employs a similar emotional pattern in both text sources. Since the news language in Twitter is not presented clearly as in news articles, this observation can help to build a cross-source system that is trained on suspicious news from news articles to detect the corresponding ones in Twitter. Figure 5.4 shows also that the emotion "joy" is the most important emotion in both datasets. It also mentions that "despair" and "hate" are almost not used in the classification process. The ranking of the features in both sources is different, where in the news articles dataset the top important emotions are "joy", "anticipation",

[21] The mean value is normalized by the sentence length since the news articles documents are longer than Tweets.

[22] Insult-wiki: http://www.insult.wiki/wiki/Insult_List

"fear", and "disgust" respectively. On the other hand, the top ones in Twitter are "joy", "sadness", "fear", and "disgust".

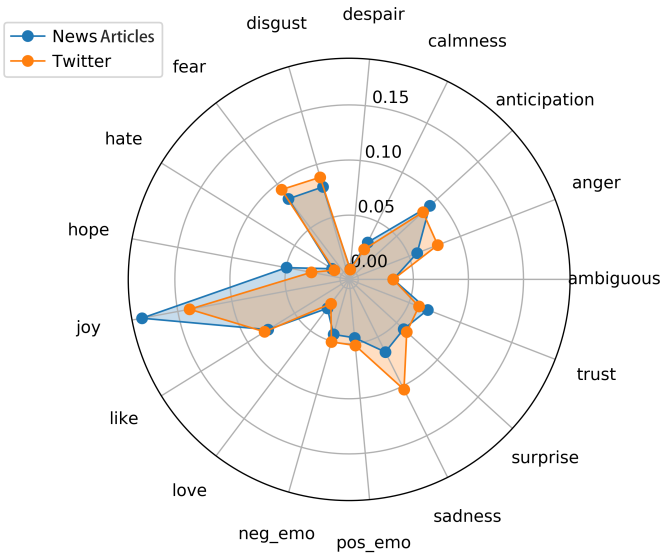


Figure 5.4: Best ranked features according to Information Gain.

RQ3 *Which of the emotions have a statistically significant difference between false information and truthful ones?*

We measure the statistical significant differences using the t-test on emotions across real news and false news (binary task) in the both datasets in Figure 5.5. These findings provide a deeper understanding of the EIN performance. The results show that "joy", "neg_emo", "ambiguous", "anticipation", "calmness", "disgust", "trust" and "surprise" have significant statistical differences between real and suspicious news in both datasets. Some other emotions such as "despair" and "anger" have no statistical difference in both datasets. It turns out that the results we obtain are generally consistent with the IG results in research question **RQ2**. We notice in the IG analysis that some emotions have a higher importance in one of the news sources: "sadness", "anger", and "fear" have a higher importance in Twitter than in news articles, and the opposite for "hope". We observe the same findings using the t-test.

RQ4 *What are the top-N emotions that discriminate false information types in both textual sources?*

False information types are different in the way they present the news to the reader. This raises a question: what are the top employed emotions

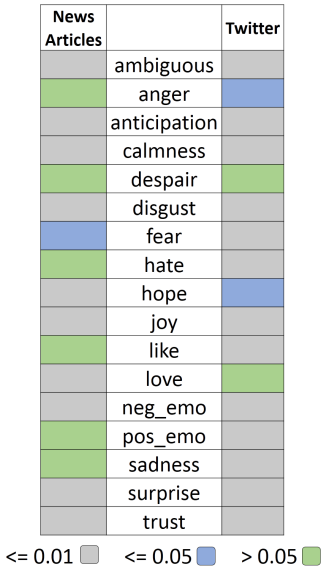


Figure 5.5: Statistical significant differences between false and real news on Twitter and news articles datasets using t-test.

in each type of false information? In Table 5.7, we present the first three[23] emotions that contribute mostly to the classification process to each type. This can indicate to us what are the emotion types that are used mostly in each type of false information.

Table 5.7 shows that clickbaits express "surprise" and "negative emotion" at the most. This validates the definition of clickbaits as "attention redirection" by exploiting the reader and convincing him/her that there is an unexpected thing with negative emotion. The result of seeing "fear" in the top features in Twitter is interesting; one of the recent studies is presenting the hypothesis that says: *curiosity is the best remedy for fear* [135] based on psychological interpretations. Taking into account the definition of clickbaits as "attention redirection", looking at our results, we can proof this hypothesis. Furthermore, despite the language differences in both datasets, we obtain almost the same results, which emphasize our results. For hoaxes, it is not simple to interpret a specific pattern of emotions in the results. We might justify it by the fact that hoaxes are written to convince the reader of the validity of a story. Therefore, the writer is trying to present the story in a normal way (truthful) similar to a real story. Therefore, the top emotions are not unique to the hoax type. But what we find from the top hoaxes emotions in both datasets is that they are generally different except the emotion "like". Despite the natural narrative way of presenting the story,

[23]We used SVM classifier coefficients (linear kernel) to extract the most important emotions to each classification class.

Table 5.7: The top 3 most important emotions in each false information type.

Rank	clickbait	hoax	propaganda	satire
News Articles				
1	surprise	hope	joy	disgust
2	neg_emo	anger	fear	neg_emo
3	like	like	calmness	pos_emo
Twitter				
1	surprise	like	fear	pos_emo
2	neg_emo	disgust	hope	disgust
3	fear	anticipation	calmness	sadness

the analysis shows that the writer still uses "like" to grab reader's attention smoothly. Propaganda type has clearer emotional interpretation considering its definition. We find that propaganda expresses "joy", "fear" and at the same time "calmness" in the news articles. Both "joy" and "fear" are contrary from an emotional polar perspective, where "joy" shows the extreme of the positive emotions and "fear" the extreme negative, and at the same time, "calmness" is present. The emotional shifting between the two extremes is a clear attempt of opinion manipulation from an emotional perspective. We obtain a similar emotion set from Twitter, but instead of "joy" we get "hope". Lastly, satire is defined as a type of parody presented in a typical format of mainstream journalism, but in a similar way to irony and sarcasm phenomena [192]. The results of the analysis show that "disgust" and "positive emotion" are present in both datasets, but we get "negative emotion" in the news articles and "sadness" in Twitter (both are placed in the negative side of emotions). We are interested in investigating the cause of the emotion "disgust" which appeared in the results from both datasets. We conduct a manual analysis on the text of the satire type in both datasets in order to shed some light on the possible causes. We notice that the satire language in the news often employs the emotion "disgust" to give a sense of humor. Figure 5.6 shows some examples from the news articles dataset highlighting the words that triggered the emotion "disgust".

5.7 Conclusions and Future Work

In this article we have presented an emotionally-infused deep learning network that uses emotional features to identify false information in Twitter and news articles sources. We performed several experiments to investigate the effectiveness of the emotional features in identifying false information.

News Articles:

freshman nate washburn was mutilated in front of students players....midnight madness sacrifice a freshman....so they can devour it as one eruditio et religio following the ritualistic....

***

....phil zipper was smacked into this week by a forceful blow delivered by his wife during...after materializing in a burst of swirling colored....of last week's smack zipper who....have hatred and prejudice finally been eradicated....

Twitter:

marine corps adds file to trash bin to command climate survey procedures.

***

nice this guy has podcasts and no toilet paper.

***

florida zoo employee killed while attempting to rape alligator.

Figure 5.6: Examples from news articles and Twitter datasets trigger the emotion "disgust".

We validated the performance of the model by comparing it to a LSTM network and other baselines. The results on the two datasets showed that clickbaits have a simpler manipulation language where emotions help detect them. This demonstrates that emotions play a key role in deceiving the reader. Based on this result, we investigated our model performance on a clickbaits dataset and we compared it to the state-of-the-art performance. Our model showed superior results near to 96% F1 value.

Overall results confirmed that emotional features have boosted EIN model performance achieving better results on 3 different datasets (**RQ1**). These results emphasized the importance of emotional features in the detection of false information. In Twitter, false news content is deliberately sexual oriented and it uses many insulting words. Our analysis showed that emotions can help detect false information also in Twitter.

In the analysis section, we answered a set of questions regarding the emotions distribution in false news. We found that emotions have similar importance distribution in Twitter and news articles regardless of the differences in the used languages (**RQ2**). The analysis showed that most of the used emotions have statistical significant difference between real and false news (**RQ3**). Emotions plays a different role in each type of false information in line with its definition (**RQ4**). We found that clickbaits try to attract the attention of the reader by mainly employing the "surprise" emotion. Propagandas are manipulating the feelings of the readers by using extreme positive and negative emotions, with triggering a sense of "calmness" to confuse the readers and enforcing a feeling of confidence. Satire news instead use the "disgust" emotion to give a sense of humor. To sum up, we can say that the initial part of false news contains more emotions than the rest of document. Our approach exploit this fact for their detection.

To the best of our knowledge, this is the first work that analyzes the impact of emotions in the detection of false information considering both social media and news articles. As a future work, the results of our approach as a clickbaits detector motivate us to develop for a clickbaits detector as a web browser extension. Also, we will study how the emotions flow inside the articles of each kind of false information, which is worthy to be investigated as the results of this work confirmed.

Part III

False Information Spreaders

In this third part we address the problem of false information from the perspective of the users that could be behind its propagation. We study how different features extracted from social media accounts can help for the detection of false information spreaders.

In Chapter 6 we propose an approach for detecting Twitter accounts that spread false information in social media. In our model, we utilize the sequential order of tweets in the profiles' streams for the detection process. We study the importance of several semantic and lexicon-based features that we extract from the tweets' texts. The results show the importance of our representation by comparing its performance to several baseline models.

Chapter 6

FacTweet: Profiling Fake News Twitter Accounts

Abstract. We present an approach to detect fake news in Twitter at the account level using a neural recurrent model and a variety of different semantic and stylistic features. Our method extracts a set of features from the timelines of news Twitter accounts by reading their posts as chunks, rather than dealing with each tweet independently. We show the experimental benefits of modeling latent stylistic signatures of mixed fake and real news with a sequential model over a wide range of strong baselines.

Published in:

- **Ghanem, B.** and Ponzetto, S. P. and Rosso, P. (2020). FacTweet: Profiling Fake News Twitter Accounts. (eds) Statistical Language and Speech Processing (SLSP). Lecture Notes in Computer Science. Springer, Cham.

6.1 Introduction

Social media platforms have made the spreading of fake news easier, faster as well as able to reach a wider audience. Social media offer another feature which is the anonymity for the authors, and this opens the door to many suspicious individuals or organizations to utilize these platforms. Recently, there has been an increased number of spreading fake news and rumors over the web and social media [229]. Fake news in social media vary considering the intention to mislead. Some of these news are spread with the intention to be ironic or to deliver the news in an ironic way (satirical news). Others, such as propaganda, hoaxes, and clickbaits, are spread to mislead the audience or to manipulate their opinions. In the case of Twitter, suspicious news annotations should be done on a tweet rather than an account level, since some accounts mix fake with real news. However, these annotations are extremely costly and time consuming – i.e., due to high volume of available tweets. Consequently, a first step in this direction, e.g., as a pre-filtering step, is the task of detecting fake news at the account level. The main obstacle for detecting suspicious Twitter accounts is due to the behavior of mixing some real news with the misleading ones. Consequently, we investigate a way to detect suspicious accounts by considering their tweets in groups (chunks). Our hypothesis is that suspicious accounts have a unique pattern in posting tweet sequences. Since their intention is to mislead, the way they transition from one set of tweets to the next has a hidden signature, biased by their intentions. Therefore, reading these tweets in chunks has the potential to improve the detection of the fake news accounts.

In this work, we investigate the problem of discriminating between factual and non-factual accounts in Twitter. To this end, we collect a dataset of tweets using a list of *propaganda*, *hoax* and *clickbait* accounts and compare different versions of sequential chunk-based approaches using a variety of feature sets against several baselines. Several approaches have been proposed for news verification, whether in social media (rumors detection) [229, 227], or in news claims [13]. The main line of research of previous work is to verify the textual tweets but not their sources. Another existing direction in the literature is the detection of online trolls or bots [199]. This is different from our setting, since online trolls are less formal and try to imitate individuals by spreading a mixed content, e.g., social media funneling [29], news, personal opinions [46], etc. On the other hand, the content of fake news Twitter accounts is formal, objective, and focused on spreading news content only. To the best of our knowledge, this is the first work aiming to detect factuality at account level, specifically from a textual perspective. The contributions of this work are the following ones:

- We propose an approach to detect non-factual Twitter accounts by treating post streams as a sequence of tweets' chunks. We test several

semantic and dictionary-based features together with a neural sequential approach, and apply an ablation test to investigate their contribution.

- We benchmark our approach against other approaches that discard the chronological order of the tweets or read the tweets individually. The results show that our approach produces superior results at detecting non-factual accounts.

The rest of the paper is structured as follows. In the following section, we present an overview on the related work. In Section 6.3, we present the methodology of our approach. Section 6.4 describes the collected dataset, the experiments, and the results. Finally, we draw some conclusions and discuss possible future works.

6.2 Related Work

Fake news detection has gained a lot of attention and has been approached from several perspectives in both social media and online news sites. Our work is closely related to the following areas.

6.2.1 Fake News Sources

Previous works focuses on approaching and analyzing online news texts or claims [86, 88]. Instead, the work in [13] looks at characterizing entire news media. The authors propose a set of features for the detection of low-factual news media. They use features based on Wikipedia pages and Twitter accounts, like *Does it have Wikipedia page?*, *Is the Twitter account verified?, etc.*. Also, they use manual features to identify the low-factual media using their malicious URLs, a set of features to capture the reporting language of the news articles, and the *Alexa Rank* metric to model the web traffic over the news media. The system shows a macro-F1 value of ∼0.6 over 3 classes, low, mixed, and high factuality. Another work [14] approaches the problem of detecting the trustworthiness of news media by combining the factuality with bias in a multi-task ordinal regression framework that models the two problems jointly. The authors use the same feature set that was proposed in [13] and show that their system can generate a good result using the Mean Absolute Error metric with a value of ∼0.53. In the direction of understanding the characteristics of not credible news sources, the work [2] studied the correlation of a set of features with credible and transparent news media. And in [3], the same authors propose a regression task for source credibility assessment using a set of features like *Google page rank, Alexa rank, Spam score, etc.*, and achieve a value of ∼17.7 using RMSE (Root Mean Squared Error).

6.2.2 Fishy Twitter Accounts

Suspicious accounts in social media play a key role in spreading fake news and deceiving other online users. A set of work has been done to detect bots or trolls accounts. Many work [29, 8, 113, 77] propose a set of features to detect online trolls, starting from textual features such as *the existence of hashtags and URLs in the trolls tweets, bag-of-words, part-of-speech features* or with including more sophisticated features such as *bot likelihood, topic-based information, and activity-related account metadata.* The majority of these work focus on online Russian trolls that were spreading fake news during the US 2016 elections, and produced superior results comparing to baselines.

The work in [54] propose a bots detection system called *BotorNot*[1] to detect bots in Twitter. The system uses content, sentiment, friend, network, temporal, and user features. The authors use a dataset of Twitter accounts – collected previously in another work – that spread tweets about online products (advertisements), duplicate others' tweets, etc.. The system obtained an Area Under ROC Curve (AUC) value of 0.95. In a similar attempt, the authors of [62] propose *SentiBot* to detect online bots in the context of the 2014 Indian election. The system uses a large combination of features that contain sentiment, topic, network, and syntax features. The proposed model obtains Receiver Operating Characteristic Curve (ROC) value of ∼0.73 on a dataset collected within a year from Twitter.

6.3 Methodology

Given a news Twitter account, we read its tweets from the account's timeline. Then we sort the tweets by the posting date in ascending way and we split them into N chunks. Each chunk consists of a sorted sequence of tweets labeled by the label of its corresponding account. We extract a set of features from each chunk and we feed them into a recurrent neural network to model the sequential flow of the chunks' tweets. We use an attention layer with dropout to attend over the most important tweets in each chunk. Finally, the representation is fed into a softmax layer to produce a probability distribution over the account types and thus predict the factuality of the accounts. Since we have many chunks for each account, the label for an account is obtained by taking the majority class of the account's chunks.

Input Representation. Let t be a Twitter account that contains m tweets. These tweets are sorted by date and split into a sequence of chunks $ck = \langle ck_1, \ldots, ck_n \rangle$, where each ck_i contains s tweets. Each tweet in ck_i is represented by a vector $v \in \mathbb{R}^d$, where v is the concatenation of a set of features'

[1] Later on, the authors created an online API for the system called Botometer in: https://botometer.iuni.iu.edu.

vectors, that is $v = \langle f_1, \ldots, f_n \rangle$. Each feature vector f_i is built by counting the presence of tweet's words in a set of lexical lists.

Features. We argue that different kinds of features like the sentiment of the text, morality, and other text-based features are critical to detect the nonfactual Twitter accounts by utilizing their occurrence during reporting the news in an account's timeline. We employ a rich set of features borrowed from previous works in fake news, bias, and rumors detection [229, 227, 13].

- **Emotion**: We build an emotions vector using word occurrences of 8 emotion types from the NRC lexicon [148], which contains ∼14K words labeled using the eight Plutchik's emotions. The emotions feature can detect if an account is frequently triggering negative emotions like fear, anger, etc.

- **Sentiment**: We extract the sentiment of the tweets by employing EffectWordNet [43], SenticNet[2] , NRC [148][3], and subj_lexicon [238], where each has the two sentiment classes, *positive* and *negative*. The sentiment feature can highlight the polarity in a more abstract level than emotions.

- **Morality**: Features based on morality foundation theory [97] where words are labeled in one of the following 10 categories (*care, harm, fairness, cheating, loyalty, betrayal, authority, subversion, sanctity,* and *degradation*). Using the morality features, we can highlight if some Twitter fake news accounts are posting more frequently news about harmful, subversion, or degradation events. It has been proved that fake news accounts usually post messages about very negative events to catch the readers' eyes [86].

- **Style**: We use canonical stylistic features, such as the count of question marks, exclamation marks, consecutive characters and letters[4], links, hashtags, users' mentions. In addition, we extract the uppercase ratio and the tweet length. We aim to detect if a specific account uses a fixed language style.

- **Words embeddings**: We extract words embeddings of the tweets' words using $Glove840B - 300d$[5] pretrained model[6]. The tweet final representation is obtained by averaging its words embeddings. The word embeddings is important to extract the topic information from the messages. Fake news accounts usually post news about specific

[2] https://sentic.net/

[3] NRC has also two sentiment categories, positive and negative.

[4] We considered 2 or more consecutive characters, and 3 or more consecutive letters.

[5] https://nlp.stanford.edu/projects/glove/

[6] Experimentally, we found that the *GloVe* model achieves better results than *Google News word2vec* or *fastText* models.

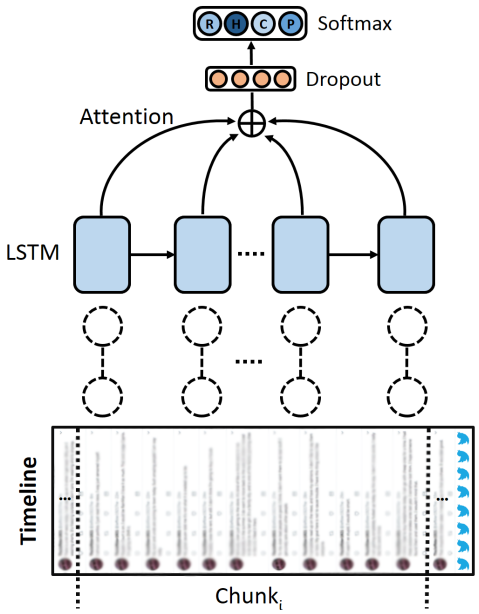


Figure 6.1: The FacTweet's architecture.

topics. Also, this feature is complementary to the previous ones where, for an example, detecting a negative sentiment without knowing the topic of the messages would not be useful.

Model. To account for chunk sequences we make use of a *de facto* standard approach and opt for a recurrent neural model using long short-term memory (LSTM). In our model, the sequence consists of a sequence of tweets belonging to one chunk (Figure 6.1). The LSTM learns the hidden state h_t by capturing the sequential changes in the timesteps. The produced hidden state h_t at each time step is passed to the attention layer which computes a 'context' vector c_t as the weighted mean of the state sequence h by: $c_t = \sum_{j=1}^{T} \alpha_{tj} h_j$, Where T is the total number of timesteps in the input sequence and α_{tj} is a weight computed at each time step j for each state h_j.

6.4 Experiments and Results

Data. We build a dataset of Twitter accounts based on two lists annotated by professional journalists. For the non-factual accounts, we rely on a list of approximately 180 Twitter accounts from [227][7]. This list was created based on public resources[8] where suspicious Twitter accounts were annotated with

[7]Many of the accounts were deactivated during the collecting process, consequently only 144 accounts were used.

[8]http://www.propornot.com/p/the-list.html

Table 6.1: Statistics on the data with respect to each account type: propaganda (**P**), clickbait (**C**), hoax (**H**), and real news (**R**).

	Accounts Types			
	P	C	H	R
# of accounts	96	36	7	32
Max # of tweets/account	3,250	3,246	3,250	3,250
Min # of tweets/account	33	877	453	212
Avg # of tweets/account	2,978	3,112	2,723	3,124
Total # of tweets	291,885	112,050	19,065	99,967

the main fake news types (clickbait, propaganda, satire, and hoax). We discard the satire labeled accounts since their intention is not to mislead or deceive. On the other hand, for the factual accounts, we use a list with another 32 Twitter accounts from [118] that are considered trustworthy by independent third parties[9]. We discard accounts that publish news in languages other than English (e.g., Russian or Arabic). Moreover, to ensure the quality of the data, we remove the duplicate, media-based, and link-only tweets. For each account, we collect the maximum amount of tweets allowed by Twitter API. Table 6.1 presents statistics on our dataset.

Baselines. We compare our approach (FacTweet) to the following baselines:

- **LR + Bag-of-words**: We aggregate the tweets of a feed and we use a bag-of-words representation with a logistic regression (LR) classifier.
- **Tweet2vec**: We use the model proposed in [61] which is a Bidirectional Gated recurrent neural network to predict the tweets based on their hashtags. Their model converts the tweets into character one-hot encoding and feed them to the model. We used our collected dataset which consists of ∼0.5M tweets to train this model. We keep the default parameters that were provided with the implementation. To represent the tweets, we use the decoded embedding produced by the model. With this baseline we aim at assessing if the tweets' hashtags may help detecting the non-factual accounts.
- **LR + All Features (tweet-level)**: We extract all our features from each tweet and feed them into a LR classifier. Here, we do not aggregate over tweets and thus view each tweet independently.
- **LR + All Features (chunk-level)**: We concatenate the features' vectors of the tweets in a chunk and feed them into a LR classifier.
- **FacTweet (tweet-level)**: Similar to the FacTweet approach, but at

[9] https://tinyurl.com/yctvve9h

tweet-level; the sequential flow of the tweets is not utilized. We aim at investigating the importance of the sequential flow of tweets.

- **Botometer**: We use Botometer [54], a state-of-the-art Twitter bots detection system. Botometer uses Network, User, Friends, Temporal, Content, and Sentiment features for bots detection. We aim at checking whether we can detect the Twitter fake news accounts using a bots detection system, where such accounts might have employed automated softwares to release fake news. Also, with this baseline, we assess the performance of the state-of-the-art bots detection system in our task. We fed the Botometer generated predictions to a Random Forest (RF) classifier. We chose RF after testing several classifiers, e.g., Logistic Regression, Support Vector Machine, Naive Bayes, and Feed Forward Neural Network.

- **Top-k replies, likes, or re-tweets**: Some approaches in rumors detection use the number of replies, likes, and re-tweets to detect rumors [78]. Thus, we extract top k replied, liked or re-tweeted tweets from each account to assess the accounts factuality. We tested different k values between 10 tweets to the max number of tweets from each account. Figure 6.2 shows the macro-F1 values for different k values. It seems that $k = 500$ for the top *replied* tweets achieves the highest result. Therefore, we consider this as a baseline.

Experimental Setup. We report the results using accuracy and macro F1. We experiment with 25% of the accounts for validation and parameters selection, and we apply 5 cross-validation on the rest of the data (75%). The validation split is extracted on the class level using stratified sampling: for this, we take a random 25% of the accounts from each class since the dataset is unbalanced. Discarding the classes' size in the splitting process may affect the minority classes (e.g., hoax). We use hyperopt library[10] to select the hyper-parameters on the following values: LSTM layer size (16, 32, 64), dropout $(0.0 - 0.9)$, activation function (*relu*, *selu*, *tanh*), optimizer (*sgd*, *adam*, *rmsprop*) with varying the value of the learning rate (1e-1,..,1e-5), and batch size (4, 8, 16). To reduce the effect of overfitting in FacTweet, we use the early stopping technique. For the baselines' classifier, we tested many classifiers and the LR showed the best overall performance.
Results. Table 6.2 presents the results. We present the results using a chunk size of 20, which was found to be the best size using the validation set. Figure 6.3 shows the results of different chunks sizes.

FacTweet performs better than the proposed baselines and obtains the highest macro-F1 value of 0.565. Our results indicate the importance of taking into account the sequence of the tweets in the accounts' timelines. The

[10] https://github.com/hyperopt

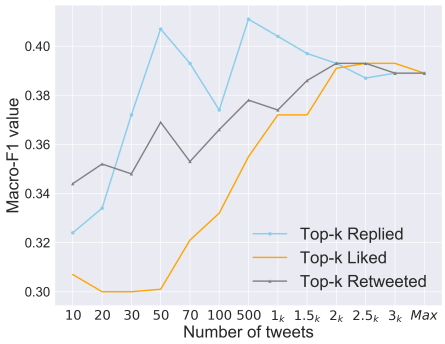


Figure 6.2: Results on the top-K replied, linked or re-tweeted tweets.

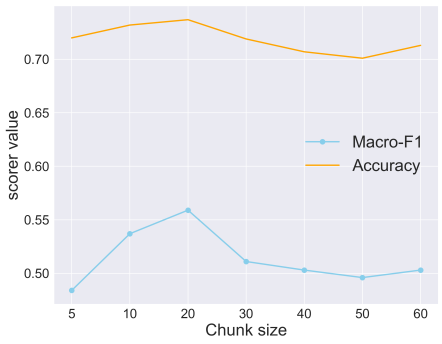


Figure 6.3: The FacTweet performance on difference chunk sizes.

sequence of these tweets is better captured by our proposed model sequence-agnostic or non-neural classifiers. Moreover, the results demonstrate that the features at tweet level do not perform well to detect the Twitter accounts factuality, since they obtain a result near to the majority class (0.18). Another finding from our experiments shows that the performance of the Tweet2vec is weak. This demonstrates that tweets' hashtags are not informative to detect non-factual accounts. Furthermore, the results show that the performance of the Botometer system is weak comparing to the other models, and this emphasizes that fake news accounts use more advanced techniques to spread fake news comparing to the more basic bots techniques. Also, we argue that the low performance of Botometer is due to the different nature of our task. Bots and trolls spread mixed information that contains advertisements and opinions, where the proposed bots detection systems, like Botometer, utilize features that give importance to such information in tweets. Also, bots accounts usually are not well connected with other users accounts (considering the network features e.g. number of followers), and such features are important to detect these accounts but not fake news accounts that gained the trust of many followers. In Table 6.3, we present ablation tests so as to quantify the contribution of subset of features. The results indicate that most performance gains come from words embeddings, style, and morality features. Other features (emotion and sentiment) show lower importance: nevertheless, they still improve the overall system performance (on average 0.35% macro-F_1 improvement). These performance figures suggest that non-factual accounts use semantic and stylistic hidden signatures mostly while tweeting news, so as to be able to mislead the readers and behave as reputable (i.e., factual) sources.

Since the dataset is highly imbalanced, we apply upsampling by replicating the minority classes. In Table 6.4 we present the results. For the model (LR + All) that is applied on the chunk-level, we do not get any

Table 6.2: Results on accounts classification.

Methods	A	P	R	F1
Baselines				
Majority Class	0.563	0.141	0.251	0.18
Random class	0.252	0.21	0.21	0.209
Bag-of-Words	0.601	0.252	0.327	0.284
Tweet2vec	0.558	0.157	0.213	0.181
Botometer	0.512	0.356	0.371	0.363
Tweet-level approaches				
LR + All	0.671	0.378	0.411	0.393
LR + All (top-500 replied)	0.443	0.368	0.467	0.411
LR + FacTweet	0.651	0.34	0.37	0.351
Chunk-level approaches				
LR + All	0.737	0.603	0.552	0.559
FacTweet	**0.74**	**0.549**	**0.582**	**0.565**

Table 6.3: Ablation tests.

Methods	Accuracy	Precision	Recall	F1
LR + All	0.737	0.603	0.552	0.559
− Emotion	0.731	0.581	0.535	0.557
− Sentiment	0.731	0.535	0.575	0.554
− Morality	0.725	0.554	0.542	0.548
− Style	0.737	0.521	0.508	0.514
− Words embeddings	0.678	0.43	0.444	0.437

improvement. For the FacTweet, we notice a small improvement in terms of F1 score. We leave a more fine-grained, diachronic analysis of semantic and stylistic features – how semantic and stylistic signature evolve across time and change across the accounts' timelines – for future work.

6.5 Conclusions and Future Work

In this paper, we proposed a model that utilizes chunked timelines of tweets and a recurrent neural model in order to infer the factuality of a Twitter news account. Our experimental results indicate the importance of analyzing tweet stream into chunks, as well as the benefits of heterogeneous knowledge source (i.e., lexica as well as text) in order to capture factuality. In future work, we would like to extend this line of research with further in-depth analysis to understand the flow change of the used features in the accounts' streams. Moreover, we would like to take our approach one step further incorporating explicit temporal information, e.g., using timestamps. Crucially, we are also

Table 6.4: Up-sampling (Up-s).

Methods	Accuracy	$F1_{macro}$
LR + All	0.737	0.559
LR + All + Up-s	0.737	0.559
FacTweet	0.74	0.565
FacTweet + Up-s	0.74	0.571

interested in developing a multilingual version of our approach, for instance by leveraging the now ubiquitous cross-lingual embeddings [79]. Finally, we will investigate the potential of applying transfer learning from social media posts. As transfer learning models are starving for data, we will work on extending the used dataset with further social media accounts to enable more accurate fine-tuning process.

Part IV

Summary

Chapter 7

Discussion of the Results

7.1 Introduction

As we mentioned previously, false information has been categorized into misinformation and disinformation based on the intent to harm. According to [233], information disorder has three main types as in Figure 7.1. In addition to mis/disinformation, malinformation type is considered. The authors defined it as: "is when genuine information is shared to cause harm, often by moving information designed to stay private into the public sphere". Malinformation has three main types which are leaks, hate speech, and harassment.

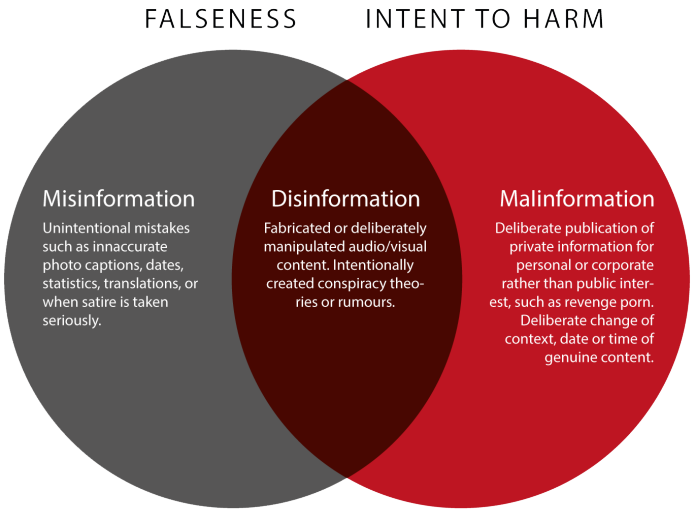


Figure 7.1: Information disorder categories [57].

Malinformation has gained a lot of attention recently by the research community since it leads to a considerable amount of harm to victims. For

instance, several shared tasks have been organized to detect hate speech, for instance, against immigrants and women [17]. Another shared task organized [28] to detect more general hateful messages that covers a set of categories, e.g., religion, race, sex, etc. In addition to the work on mis/disinformation, in the framework of our PhD we also addressed the problem of detecting malinformation and concretely misogyny in Twitter messages [74, 72, 73].

In this chapter, we present some further experiments in order to have a more complete picture of the problem. This should allow us to better answer the three research questions. First, in Section 7.2, we present a research work for detecting check worthy news claims, and for fact-checking those claims. Moreover, we propose an approach for detecting fake news considering the stance information.

In Section 7.3, we present further experiments we have conducted to investigate the role of emotions in false information. Unlike our proposed work in Chapter 5 where we took into account emotions but without considering their sequential order in texts to detect false information, in this work, we propose a deep neural network that considers the order of affective information that appears in fake news articles.

Finally, in Section 7.4, first, we compare fake news checkers and spreaders on Twitter from a textual perspective using personality and linguistic -based features extracted from tweets text. Then, we study false information spreaders during the 2016 US elections from several perspectives (e.g topical, stylistic, affective, etc.). Also, we present a proposed textual approach to detect them. Last, we present the overview of the shared task that we organized on profiling fake news spreaders on Twitter. We present our baseline models, the collected datasets for both English and Spanish languages, and the preliminary results that were obtained.

7.2 False Information and Fact-Checking

The need for automatic fact-checking systems is increasing over the years, and massively recently. Our fact-checking approach that we presented in Chapter 3 is a good example of that kind of systems we need to verify claims. Besides that the proposed approach is fully automatic, it effectively validates factual claims from a cross-lingual perspective. Cross-lingual approaches can help with the low-resource languages where the amount of the available data to train a fact-checking system is small. The experiments that we have conducted in Section 3.5 demonstrated this need. On top of this cross-lingual system, previously we proposed a monolingual fact-checking system that uses similarity metrics to verify claims [82]. Given the large number of online claims, many of them are not check-worthy; sentences that do not contain an event to be validated. Thus, as a first step towards accurate fact-checking systems, we proposed in [83] a system to detect check-worthy

claims in the context of political debates.

Figurative language plays an essential role in false information, precisely in the satire one. Understanding the language of the news may help to identify its truthfulness. For instance, detecting an ironic sense in a news article indicates that the news is satirical. Irony detection was also investigated by us previously in a work [84] where we addressed the problem with a model that exploits low dimensional features extracted from the text based on the occurrence probability of words depending on each class (irony or not). It is worthy of mentioning at this point that in our way to limit false information, we organized a shared task on detecting ironic messages in social media for Arabic language [81].

Following, we present some additional experiments we carried out for further investigating the issue of false information.

7.2.1 Claims Verification

In Chapter 3 we presented a cross-lingual approach for validating factual claims. In this work, we present additional experiments that we have done to validate claims from a monolingual perspective. We present our claim verification approach that proposed for the CheckThat! 2018 shared task [82].

7.2.1.1 Method

Factual claims have been discussed and mentioned in online news agencies. In our approach, we used the distribution of the claims in the search engines results. Furthermore, we supposed that truthful claims have been mentioned more by trusted web news agencies than the untruthful ones. Therefore, our approach depends on modeling the returned results from search engines using similarity measures and with extracting the reliability of the results' sources (dependent features). Also, we captured the distribution of these previous features from each search engine (independent features).

At the beginning, we started to reformulate each claim into a query. We fed this query to the Google and the Bing search engines to obtain a set of results. In our approach, we use only the returned snippets and we do not investigate more the original web pages. Given the search engine results, we used in our approach the first N results for the feature extraction. Next, we built the representation of the features:

1. ***Independent features***: For each returned result, we extracted the following three features:

    - *Cosine over embedding*: we used pre-trained Google News word2vec embedding to measure the cosine similarity between each snippet

and the query. We used the main sentence components, discarding the stopwords. In the same way, we built this feature for the Arabic language, but we used fastText pre-trained embedding [27] since Google news word2vec is not available.

- *AlexaRank*: For each result, we used Amazon Alexa Rank to retrieve the rank of its site. The sites that have lower values are the sites that have higher reliability.
- *Text similarity*: we used another text similarity measure, but using the full sentence components (similarity over tokens), and without text embedding. For the English part, we used the Spacy python library[1], while for the Arabic language, since this library is not available, we implemented the text similarity approach that used in [76] for plagiarism detection.

As we mentioned, we considered the first N results from the search engines, thus, we ended with a features vector of size 3×N.

2. ***Dependent features***: We extracted a set of features based on the previous independent features. These features model the distribution of the previous feature set, that has been extracted using Average (Avg) and Standard Deviation (Std).

   - *Avg and Std of AlexaRank feature*: We computed both Avg and Std features for the Alexa values that were extracted for the first N results.
   - *Avg and Std of the Cosine over embedding feature*: Similarly, we computed also the Avg and the Std features for the cosine similarities values that were extracted.

   At the end, our representation has (3×N) + 4 features.

All of these previous dependent and independent features were extracted twice, once from Google and another one from Bing search engines. In the following section, we will investigate their importance.

7.2.1.2 Data

This task concerns with investigating claims veracity in presidential debates. Therefore, a set of presidential debates from the US presidential debates is presented. Factual claims have been tagged as True, False, and Half-True. These debates are provided in two languages: English and Arabic, where the Arabic text is translated from the original English debates. The dataset that was provided is imbalanced, where the total number of factual claims is 81; claims as True: 19, Half-True: 22, False: 41.

[1] https://spacy.io/, visited in May 2018

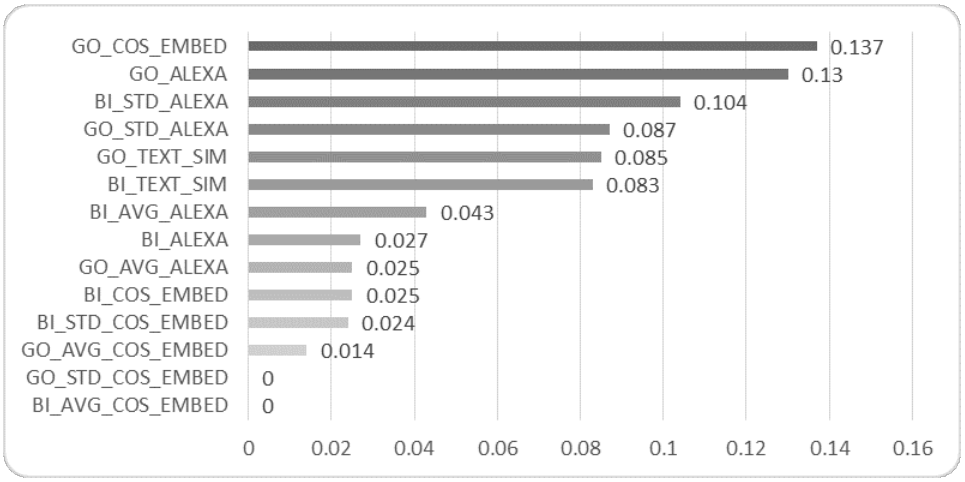


Figure 7.2: The Information Gain values of the feature set. The features that started with BI are the ones built using Bing, similarly, GO for Google.

7.2.1.3 Experiments and Results

As we mentioned in the previous section, we built our features based on the first N results from the search engines. Experimentally, we found that choosing the first 5 results (N=5) has produced the highest results. Based on that, our feature vector length is 38 features.

In Figure 7.2, we show the information gain of these features for each search engine. From these results, we can infer that the features that were obtained by the Google search engine are more important than the Bing features. Based on that, we can notice that the Google results can improve the performance more than the Bing results. At the beginning of our experiments, we tried also to combine Yahoo results, but unfortunately, in all of our experiments, Yahoo results had a lower performance. Since our approach is search engines -based, for the Arabic task, we found that these claims did not exist because they were written originally in English and translated into Arabic for this task. Therefore, we translated back these claims into English to retrieve results. After that, the results and the query were translated back to Arabic.

During our experiments, many classifiers were tested. We found that the Random Forest classifier achieved the highest results. By using the K-Fold stratified technique, we achieved 0.34 of macro F1 score. The chosen value of K is 5, where we have a small number of data instances and an imbalanced dataset. Thus, higher values of K may lead to absence of some classes in one of the training/testing cycles. We tried also to build a different type of queries, using main sentence components, or phrase queries, but we found that when we changed the query, the results were affected negatively, especially when we used the phrase query, we noticed that the search en-

Table 7.1: Official results released using the MAE measure.

Team	English	Arabic
Copenhagen	0.705	–
FACTR	0.913	0.657
UPV-INAOE	0.949	0.820
bigIR	0.964	–
Check It Out	0.964	–

gines snippets became meaningless (phrases appeared in the snippets but as small text clips connected using "..." characters and combined into the main snippet, where the semantic meaning of the main snippet became biased). For this reason, we passed the queries without any modification, letting the search engines to retrieve the most appropriate results for each one. For the shared task official testing phase the Mean Absolute Error (MAE) was used as a performance measure. In Table 1, the results of the task are shown.

In the English part, our approach obtained the 3rd best results, while for the Arabic part, only two teams have submitted their runs. We can observe that the results are low, showing the difficulty of the task.

7.2.2 Check-worthy Claims

In an attempt to improve the validation process of fact-checking systems, in this section we propose a system for detecting check-worthy claims in the context of political debates [83]. This work was proposed for the checkthat! 2018 shared task. The task goal is to detect claims that are worthy for checking and to rank them, from the most worthy one for checking to the lowest one.

7.2.2.1 Method

Previously, the authors in [98] have proposed a text distortion technique to enhance thematic text clustering by maintaining the words that have a low frequency in a document. Later on, a similar research in [210] has used the same text distortion technique for authorship attribution task, where the author has maintained the words that have the highest frequency in the documents, in an attempt to detect the author from her writing style.

We believe that this type of tasks is more thematic than stylistic, where the writing style is not important as the thematic words. In our approach, we used the same text distortion technique to detect worthy claims, where we concealed words that have high frequency in documents and maintaining (highlighting) other cue words that are used more in factual claims. Therefore, we followed [98] and we maintained the thematic words (that have the lowest frequency) using a threshold (C). The higher value of C is, the more

Table 7.2: Samples from the linguistic lexicons.

Linguistic lexicons	Examples
Assertives	appear, declare, guarantee, hypothesize
Factives	learn, realize, know, discover
Hedges	almost, guess, indicate, mostly
Implicatives	cause, manage, hesitate, neglect
Report	admit, answer, clarify, comment
Bias	adhere, act, agree, allow, addition
Subjectivity	afraid, champ, apologist, amusement

Table 7.3: An example of the text distortion process using different values of C.

Original claim	It was actually $1.7 billion in cash, obviously, I guess for the hostages.
C = 0	** *** ******** $ # . # ******* ** **** , ********* , * ***** *** *** ******** .
C = 0 & LC+NE	** *** actually $ # . # ******* ** **** , obviously , * guess *** *** ******** .
C = 2000 & LC+NE	** *** actually $ 1 . 7 ******* ** cash , obviously , * guess *** *** hostages .
C = 3500 & LC+NE	It *** actually $ 1 . 7 billion in cash , obviously , I guess for *** hostages .

thematic words are maintained. Also, we maintained a set of linguistic cue words (LC) that was used previously in [150] to infer the credibility of news (see Table 1). Additionally, we maintained also the named entities (NE) from being distorted, such as: Iraq, Trump, America. Through manually checking the claims, we found the check-worthy claims tended to list different types of named entities.

In Table 2, we show an example of the distortion process.

After applying the distortion process, the new version of the text was used by the char n-gram model using Tf-Idf weighting scheme. The new distorted text, becomes less biased by the high frequency words, such as stopwords. Finally, after preparing the distorted text, there is still one issue which is the value of C variable. The value of C is crucial, being a threshold between the amount of thematic and the stylistic words. In the next section we show how we select the most appropriate value of C. For the Arabic language, we employed the same approach, where the only issue we had was the Arabic version of the linguistic lexicons. The manual translation of them is a time-consuming process, where they are quite large. Therefore, we used Google Translation API to translate these lexicons.

Table 7.4: The results obtained during the tuning phase using word and char n-gram models. We chose @N in our experiments as the last record in the testing part. TD is an abbreviation for Text Distortion.

Approach	Classifier	K	C	n-rams	AVG Prec. @N
TD + char n-gram	KNN	1	1700	4	0.234
TD + char n-gram (Without NE)	KNN	1	1900	4	0.209
TD + char n-gram (raw text)	KNN	1	2100	3	0.141
TD + word n-gram	KNN	2	2500	1	0.157
Baseline, word n-gram	SVM	_	_	1	0.163

7.2.2.2 Data

A set of presidential debates from the US presidential debates is presented for the task, where each claim in the debate text has been tagged manually as worth to be check (1) or not (0). The text of the debates is used for the task as it is, to give the opportunity to the potential approaches to exploit contextual features in the debates. These debates are provided in two languages, English and Arabic, where the Arabic text is obtained translating from the English debates. The dataset that was provided is totally imbalanced, where the total number of claims is 4064: 90 claims are worth to be check and 3970 are not.

7.2.2.3 Experiments and Results

We carried out many experiments to test different machine learning classifiers. We found that the K-Nearest-Neighbor (KNN) has achieved the highest Average Precision value; the Average Precision was used as a performance measure. We have had two parameters to select the best model: the value of K-neighbors (K) of the KNN classifier and the C value of the distortion ratio. The selection process of these two values is hard to be set manually, therefore, we used the Grid Search technique to select the most appropriate values of these two parameters. The best value of K is 1, and for C value is 1700. The low value of K is due to the highly imbalanced situation of the dataset; larger values tend to bias the classifier to the majority class. A similar process was applied to select the best parameters but using word n-gram rather than character. For the evaluation, the Average Precision @N was used. The results of both runs are showed in Table 3. From these runs, we can see that char n-gram model outperformed clearly the one using word n-gram. After the claims have been detected, it is important to rank them based on their worthiness for checking. For the ranking process, we used the KNN classifier. We ranked the claims based on the KNN confidence in the classification process. At the beginning, we extracted the distances to the nearest neighbor (since we used K-neighbor equal to 1) for all the predictions in the test file. Then we applied a normalization for the distances to range

Table 7.5: Official results for the Task 1, released using MAP measure.

Team	English	Arabic
Prise de Fer	0.1332	–
Copenhagen	0.1152	–
UPV-INAOE	0.1130	0.0585
bigIR	0.1120	0.0899
Fragarach	0.0812	–
blue	0.0801	–
RNCC	0.0632	–

0-1. For each predicted instance, we checked the class type of the nearest neighbor: if it was positive, we subtracted the distance value from 1 and we used it for the ranking. We subtracted the distance from 1 to take the inverse of it: the small distance value (near to zero) means a high classification confidence. The highest value (near to 1) is the one that obtained a higher rank (more worthy for checking). We applied the same process when the nearest neighbor is from the negative class, the rank value by -1, in order to discriminate the positive and the negative instances.

As we mentioned before, the used measure for this task is the Average Precision. In the official testing phase, multiple testing files were presented. For the final results the Mean Average Precision (MAP) was used. The official results of the task 1 are shown in Table 4. In the English part of the task, our approach has achieved the third position among seven teams, where the results are close to each other. In the Arabic part, only two teams have submitted their results. Similar to English, the results are close and there is not a big difference between them. We believe that the lower results of our approach in the Arabic part is because of the automatic translation of the lexicons. A manual translation would have been more reliable.

7.2.3 Stance Detection in Fake News

In Chapter 4, we proposed an approach for the task of rumors validation using a stance-based model. The detection of fake news articles has been approached from several perspectives, but not from a stance one. In an attempt different from the literature work, the Fake News Challenge[2] (FNC) proposed to detect fake news articles from a stance perspective. Given a pair of text fragments (title and article) obtained from news, the task goal is to estimate the relative perspective (stance) of these two fragments with respect to a specific topic. In other words, the stance prediction of an article towards the title of this article. For each input pair, there are 4 stance labels: Agree, Disagree, Discuss, and Unrelated. "Agree" if the article supports the title;

[2] http://www.fakenewschallenge.org/

Feature	Example Words
Belief	assume, believe, think, consider
Denial	refuse, reject, rebuff, oppose
Doubt	wonder, unsure, guess, doubt
Report	evidence, assert, told, claim
Knowledge	confirm, definitely, support
Negation	no, not, never, don't, can't
Fake	liar, false, rumor, hoax, debunk

Table 7.6: The cue words categories and examples.

"disagree" if refuses it; "discuss" whether the article discusses the title but without showing an in favor or against stance; and "unrelated" when the article describes a different topic than the one of the title. In the following section, we present an approach we proposed on the FNC dataset [85].

7.2.3.1 Method

The literature work on the FNC dataset showed that the best results are not obtained with a pure deep learning architecture, and simple BOW representations showed a good performance. In our approach, we combine n-grams, word embeddings and cue words to detect the stance of the title with respect to its article. In our approach we combine simple feature representation to model the title-article tuples:

- **Cue words**: We employ a set of cue words categories that was used previously in [10] to identify the stance of Twitter users towards rumor tweets. As Table 1 shows, the cue words categories are *Belief, Denial, Doubt, Report, Knowledge, Negation and Fake*. The Fake cue list is a combination of some words from FNC baseline polarized words list and words from the original list. The provided set of cue words is quite small, therefore, we use Google News word2vec to expand it. For each word, we retrieve the most 5 similar words. As an example, for the word "misinform", we retrieved "mislead", "misinforming", "disinform", "misinformation", and "demonize" as the most similar words.

- **Google News word2vec embedding**: For each title-article tuple, we measure the cosine similarity of the embedding of each sentence. Also, we use the full 300 length embedding vector for both the title and the article. The sentence embeddings is obtained by averaging its words embeddings. Previously in [76], the authors showed that using the main sentence components (verbs, nouns, and adjectives) improved the detection accuracy of a plagiarism detection approach[3] rather than

[3]For extracting the main sentence components, we used NLTK POS tagger:

using the full sentence components. Therefore, we build these embeddings vectors using the main sentence components. Furthermore, we maintain the set of cue words that showed in the previous point.

- **FNC features**: we use the same baseline feature set (see Section 7.2.3.2).

7.2.3.2 Data and Baseline

The presented dataset by FNC was built using 300 different topics. The training part consists of 49,972 tuples in a form of title, article, and label, while the test part consists of 25,413 tuples. The ratio of each label (class) in the dataset is: 73.13% Unrelated, 17.82% Discuss, 7.36% Agree, and 1.68% Disagree. Clearly the dataset is heavily biased towards the unrelated label. Titles length ranges between 8 and 40 words, whereas for the article ranges between 600 and 7000 words [22]. These numbers show a real challenge to predict the stance between these two fragments that are totally different in lengths. As we noticed, the task's dataset is imbalanced in a high ratio. Therefore, the organizers introduced a weighted accuracy score for the evaluation. Their proposed score gave 25% of the final score for predicting the unrelated class, while 75% for the other classes. Later, the authors in [101] proposed an in-depth analysis to discuss FNC experimental setup. They showed that this accuracy metric is not appropriate and fails to take into account the imbalanced class distribution, where models performing well on the majority class and poorly on the minority classes are favored. Therefore, they proposed macro F1 metric to be used in this task. Accordingly, in this paper we show the experimental results using the Macro F1 measure.

The organizers presented a tough baseline using Gradient Boost decision tree classifier. In contrast to other shared tasks, their baseline employed more sophisticated features. As features, they employed n-gram co-occurrence between the titles and articles using both character and word grams (using a combination of multiple lengths) along with other hand-crafted features such as: word overlapping between the title and the article and the existence of highly polarized words from a lexicon (ex. fake, hoax). Their baseline achieved an FNC score value of 75% and 45.4% value of Macro F1.

7.2.3.3 Experiments and Results

In our experiments, we tested Support Vector Machines (SVM) (using each Linear and RBF kernels), Gradient Boost, Random Forest and Naive Bayes classifiers but the Neural Network (NN) showed better results[4]. Our NN architecture consists of two hidden layers with rectified linear unit (ReLU)

https://www.nltk.org/book/ch05.html.

[4]The Scikit-learn python package was used in our implementation

Systems	Macro-F1
Majority vote	0.210
FNC baseline	0.454
Talos [11]	0.582
UCLMR [190]	0.583
Athene [102]	0.604
stackLSTM[5] [101]	0.609
Our approach	0.596
Cue words	0.250
Word2vec embeddings	0.488

Table 7.7: The Macro F1 score results of the participants in the FNC challenge.

activation function as non-linearity for the hidden layers, and Softmax activation function for the output layer. Also, we employed the Adam weight optimizer. The used batch size is 200. Table 7.7 shows the results of our approach and those of the FNC participants. We investigated the score of each of our features independently. The word2vec embeddings feature set has achieved 0.488 Macro F1 value, while the cue words achieved 0.25. The extension of the cue words has improved the final result by 2.5%.

The tuples of the "Unrelated" class had been created artificially by assigning articles from different documents. This abnormal distribution can affect the result of the cue words feature when we test it independently; since we extract the cue words feature from the articles part (without the titles) and some articles could be found with different class labels, this can bias the classification process. The state-of-the-art result (stackLSTM) was obtained by an approach that combined LSTM with other features. Our approach achieved 0.596 value of macro F1 score which is very close to the best result.

The combination of the cue words categories with the other features has improved the overall result. Each of them had impact in the classification process. In Figure 7.3, we show the importance of each category using the Information Gain. We extract it using Gradient Boost classifier as it achieves the highest result comparing to the other decision tree-based classifiers. The figure clarifies that *Report* is the category that has the highest importance in the classification process, where *Negation* and *Belief* categories have lower importance, whereas both of the *Denial* and *Knowledge* categories have the lowest importance. Surprisingly, both of the *Fake* and *Doubt* categories have a lower importance than the other three. Our intuition was that the

[5]The stackLSTM is not one of the FNC participated approaches, but it achieved state-of-the-art result.

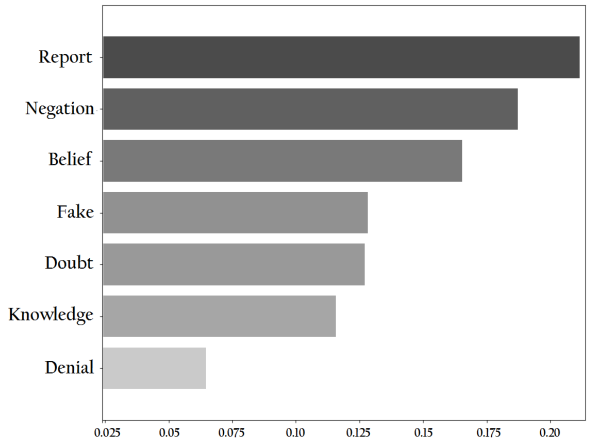


Figure 7.3: The importance of each cue words category using Information Gain.

Fake category will have the highest importance in discriminating the classes, where this category contains words that: may not appear in the "Agree" class records, appear profusely in the "Disagree" class (where the title is fake and the article proving that), and a medium appearance amount in the "Discuss" class. Similarly, for the *Doubt* category, it seems that it may appear frequently in both "Discuss" and "Disagree" classes where its words normally mentioned when an article discusses a specific idea or when refuse it.

7.3 False Information and Emotions

In today's information landscape, fake news are used to manipulate the public opinions [249] by reshaping readers' opinions regarding some issues. In order to achieve this goal, authors of fake news' narratives need to capture the interest of the reader. Thus, they put efforts to make their news articles look objective and realistic. This is usually done together with adding misleading terms that can have a negative or positive impact on the readers' emotions. Previous work [186, 34, 201] have discarded the sequential order of events in fake news articles. In this section we propose a model that takes into account the affective changes in texts to detect fake news. In this work, we hypothesize that fake news have a different distribution of affective information across their texts comparing to real news, e.g., more fear emotion in the first part, more overall harmful terms, etc. Therefore, modeling the flow of such information will help discriminate fake from real news. Our model consists of two main sub-modules: topic-based and affective information detection. We combine these two sub-modules since a news article's topic correlates with its affect information. A fake news article about Islam

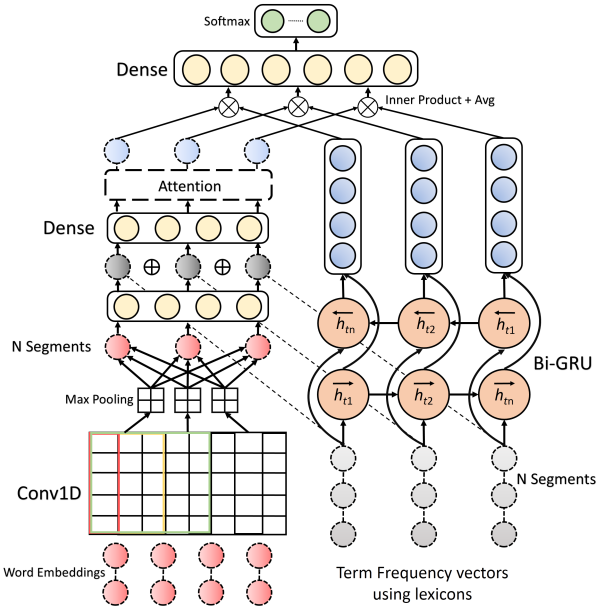


Figure 7.4: The architecture of the FakeFlow model.

or Black people likely triggers fear and negative sentiment. On the other hand, another fake news that is in favor of a politician triggers more positive emotions and some exaggerations.

7.3.1 FakeFlow Model

Given a document as an input, the FakeFlow model divides it into N segments. Then, it uses both words embeddings and other affective features such as *emotions, hyperbolic words, etc.* in a way to catch the flow of these information in the document. The model learns the flow of affective information throughout the document's text to detect whether it is fake or real. Figure 7.4 shows the architecture of the FakeFlow model. The neural architecture has two main modules. The first module uses a Convolutional Neural Network (CNN) to extract topic-based information from articles. The second module models the flow of the affective information within articles' texts via a Bidirectional Gated Recurrent Units (Bi-GRUs).

7.3.1.1 Topic-based Information

Given a segment $n \in N$ of words, the model first embeds words to vectors through an embeddings matrix. Then it uses a CNN that applies convolving processes and max pooling to get an abstractive representation of the input segment. This representation highlights important words, in which the topic information of the segment is summarized. Then it applies a fully connected

layer on the output of each segment to get a smaller representation (v_{topic}) that makes a later concatenation process with affective information more effective. This is due to the high dimensionality of the word embeddings vectors (a vector dimension is 300, see Section 7.3.3) and the low dimension of the affective vectors (dimensionality is 23, see Section 7.3.1.2).

As we previously discussed, it is important to consider the relevance of the affective information with respect to the topics. Thus, FakeFlow concatenates the topic summarized vector v_{topic} with v_{affect} vector which is the affective information that extracted from each segment (see Section 7.3.1.2). To merge the different representations and capture their joint interaction, the model processes the produced concatenated vector v_{concat} with another fully connected layer. In order to create an attention-focused representation of the segments to highlight important ones, the model applies an attention mechanism [247] on v_{concat} to consider the context of each timestep (segment) and outputs an attention matrix l_t that has scores for each token at each timestep.

7.3.1.2 Affective Flow of Information

To model the affective information flow in the news articles, we choose the following lexical features, under the assumption that they have a different distribution across the articles' segments. We use a term frequency representation weighted by the articles' length to extract the following features from each segment n:

- *Emotions*: We use emotions as features to detect their change among articles' segments. For that we use the NRC emotions lexicon [148] that contains ∼14K words labeled using the eight Plutchik's emotions (*8 Features*).

- *Sentiment*: We extract the sentiment from the text, *positive* and *negative*, similarly from the NRC lexicon [148] (*2 Features*).

- *Morality*: We consider the cues based on the morality foundation theory [97] where words are labeled in one of the following set of categories: *care, harm, fairness, unfairness, loyalty, betrayal, authority, subversion, sanctity,* and *degradation* (*10 Features*).

- *Imageability*: We use a list of words rated by the degrees of abstractness and imageability[6]. These words have been extracted from the MRC psycholinguistic database [237] and then using a supervised learning algorithm, the words annotated by the degrees of abstractness and imageability. The list contains 4,295 and 1,156 words rated by the degrees of abstractness and imageability, respectively (*2 Features*).

[6] https://github.com/ytsvetko/metaphor/tree/master/resources/imageability

- *Hyperbolic*: We use the list of hyperbolic words of [39], that are words with high positive or negative sentiments (e.g., terrifying, breathtakingly, soul-stirring, etc.). The authors extracted these eye-catching words from clickbaits news headlines (*1 Feature*).

To model the flow of the above features, we present each segment of an article by a vector v_{affect}, where v_{affect} is a vector of length 23. Then we feed the document's vectors to a Bi-GRUs network to summarize the contextual flow of the features from both directions.

Now that we have the segments' flow representation (v_{flow}) of an article and their relevance to the topics (l_t), FakeFlow applies a dot product operation and then averages the output matrix across the segments to get a final representation v_{final}. Then, the model processes this final vector by feeding it to a fully connected layer. Finally, to generate the overall factuality label of an article, a Softmax layer is applied on the output of the fully connected layer.

7.3.2 Collected Dataset

Despite the recent efforts for debunking online fake news, there is a dearth of publicly available datasets. Most of the available datasets are small in size, and become irrelevant to evaluate the performance of recent deep learning-based approaches that are starving for data. Thus, in this work we build our dataset from a large set of sources. We build this dataset in two parts, training and test. For the training part, we use *OpenSources.co* (OS), *MediaBiasFactCheck.com* (MBFC), and PolitiFact[7] news Websites' lists. OS list contains 560 domains, MBFC list has 548 domains, and the PolitiFact list has 227 domains[8]. These lists have been annotated by professional journalists. The lists contain domains of online news Websites annotated based on the content type (as in the OS news list: *satire, reliable*, etc.; and in the PolitiFact news list: *imposter, parody, fake news*, etc.) or from a factuality perspective (as in the MBFC news list: low, medium, and high factuality). In the OS list, we select domains that are in one of the following categories: *fake, bias, reliable, hate, satire,* or *conspiracy*. We consider domains under the *reliable* category as real news sources, and the rest as fake. The PolitiFact list is different than the OS list since it has only labels for domains that are either fake or with mixed content. We discard the mixed ones[9] and we map the rest to the fake news label. Finally, we select from the MBFC list those domains that are annotated either as high or low factual news and

[7] https://www.politifact.com/article/2017/apr/20/politifacts-guide-fake-news-websites-and-what-they/

[8] The lists' sizes are smaller in our experiments since that many domains were inactive when we were scraping the content of the lists' domains. The original sizes of the lists are: OS list is 1001, MBFC list is 1066, and PolitiFact list 328 domains.

[9] The discarded label is "Some fake stories".

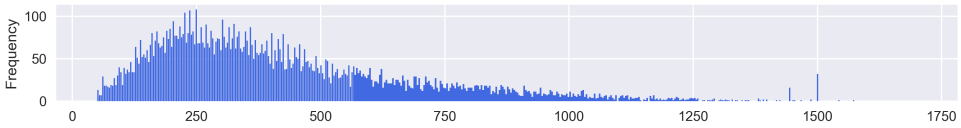


Figure 7.5: The distribution of the documents' length for the collected dataset.

we map them to real and fake labels respectively. Out of these three final lists, we select domains for our dataset that are only annotated in all lists in a consistent way; for example, we discard those domains that are annotated as real in the OS list but their label in the MBFC list is fake (low factuality). The final list contains 85 news Websites. Our approach is to project the domain-level ground truth onto the content of those domains, and thus we sample randomly a maximum of 100 articles per domain[10]. For the test part, we use *leadstories.com* fact-checking Website in which expert journalists annotated online news articles on article-level as fake or real news. We do not follow the way we build the training part since the projection of the domain-level ground truth inevitably introduces noise. The journalists in the *leadstories.com* assigned a set of labels for the fake news articles like e.g. *false, no evidence, satire, misleading*, etc.; we map them all to the fake label. We discard articles that are multimedia-based. After collecting the news articles[11], we postprocess the articles by discarding very short ones (less than 30 words). The final version consists of 4,994 real and 4,714 fake news articles. In the test part, we have 689 fake articles and we slice 1,000 real news articles from the collected part. In Figure 7.5, we show the distribution of the documents' length (number of words in a document) in the collected articles.

7.3.3 Experiments and Results

Experiments Setup As a preprocessing step, we clean the text by removing special chars and lower casing words. We split the articles' text to N segments, we set the maximum length of segments to 800 words, and do zero padding for shorter ones. For FakeFlow hyper-parameters, we tune various parameters (*dropout, the sizes of the dense layers, activation functions, CNN filters' sizes and their numbers, size of the GRU layer, and the optimization function*) using early stopping technique on the validation set. In addition to these hyper-parameters, we also use the validation set to pick the best number of segments (N). Regarding the collected dataset, we use 20% of the training part for validation, and we follow the authors setup for the rest

[10]Some of the Websites have few news articles, less than 100 news articles.
[11]We use *Newspaper3k* python library for scraping the articles content.

of the datasets. We represent terms using pre-trained *Google-News-300* embeddings[12]. We implemente our model using Keras with Tensorflow as a backend. Regarding the experiments' metrics, we follow the related works' setup. We use accuracy and weighted precision, recall, and macro F1 score.
Baselines To evaluate the performance of the proposed model, we use a combination of fake news detection models and deep neural network architectures:

- **CNN, LSTM**: We use CNN and LSTM models to validate their performance when we treat each document as one fragment, without considering any hierarchical information. We experiment with different hyper-parameters and we use the best ones using the validation set.
- **HAN**: The authors of [240] proposed a Hierarchical Attention Networks (HAN) model for long document classification. The proposed model consists of two levels of attention mechanisms, that are word and sentence attentions. The model splits a document into sentences (splits on dots), and starts learning the sentences' representation from words. The model showed strong results comparing to different models.
- **BERT_CLS, BERT_LSTM**: BERT is a text embeddings-based model that showed leading performance on multiple natural language processing (NLP) benchmarks [60]. We use the pre-trained *bert-base-uncased* version which has 12-layers and yields output embeddings with a dimension of size 768. We build two baselines based on it: 1) In **BERT_CLS** we feed the hidden representation of the special [CLS] token, that BERT uses to summarize the full input sentence, to a Softmax layer; 2) In **BERT_LSTM** we feed the hidden states representation of each sentence's words to a Long-short Term Memory (LSTM) neural network that has a Softmax layer as an output layer. Experimentally, we found that not finetuning the last [1,2] BERT layers gives a higher performance. It is worthy to mention that BERT input length is limited to 512 words [60], thus, we only feed the first 512 words of each news article.
- **Fake News Detection Models**: We compare our model to several fake news detection models. We use [165] , [110], [186], and our model that proposed in Chapter 5 (EIN).

Table 7.8 presents the results of our proposed model and the baselines on the collected dataset. Our best result was achieved by using 10 as the number of segments (N). In Figure 7.6 we show the model's performance on different segments length[13]. In general, the results show that models that are based

[12] https://code.google.com/archive/p/word2vec/

[13] In the case of N=1 in the Figure 7.6, we set the maximum segment length to 1700 words instead of 800 to not lose parts of long articles.

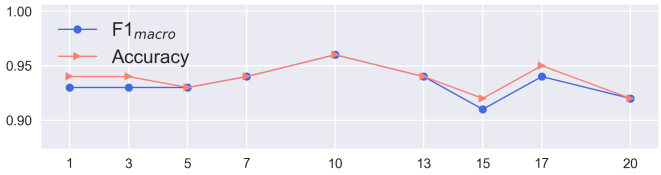


Figure 7.6: The accuracy and F1 results of FakeFlow model using different number of segments.

Model	Acc.	Prec.	Rec.	$F1_{macro}$
Majority Class	0.59	0.35	0.59	0.37
Horne & Adali, 2017 [110]	0.80	0.75	0.78	0.80
BERT_CLS	0.82	0.84	0.82	0.82
Pérez-Rosas et al., 2018 [165]	0.86	0.86	0.86	0.86
BERT_LSTM	0.89	0.89	0.89	0.89
LSTM	0.91	0.86	0.91	0.90
CNN	0.91	0.89	0.89	0.91
Rashkin et al., 2017 [186]	0.92	0.92	0.92	0.92
EIN	0.93	0.94	0.93	0.93*
HAN	0.94	0.94	0.94	0.93*
FakeFlow	0.96	0.93	0.97	**0.96**

Table 7.8: Results on the collected dataset. A star (*) indicates a statistically significant improvement of *FakeFlow* result over the referred model using McNemar test.

on either word-ngrams or word embeddings are performing better than other models that use handcraft features, e.g. [110]. Both BERT models perform lower than our proposed model and than the majority of the other models. This probably is due to the fact that the input length in BERT is limited to 512 words, as we mentioned previously, and a large portion of the news articles in the collected dataset has a length greater than 512 words (see Figure 7.5). This emphasizes that despite the strong performance of BERT on multiple NLP benchmarks, it is unable to handle long text documents.

Network Interpretation The proposed FakeFlow model shows that taking into account the flow of the affective information in fake news is an important perspective to identify them. Understanding the behaviour of the model makes it transparent to the end-users. Figure 7.7 illustrates the attention weights of a fake news article across the 10 segments (left bar). The figure shows that FakeFlow attends more on the beginning part of the article. We match the affective information with the attention weights for a better understanding. Regarding the news text in the figure, the *emotions* features[14]

[14]Words with multiples colors mean that they have been annotated with multiple emo-

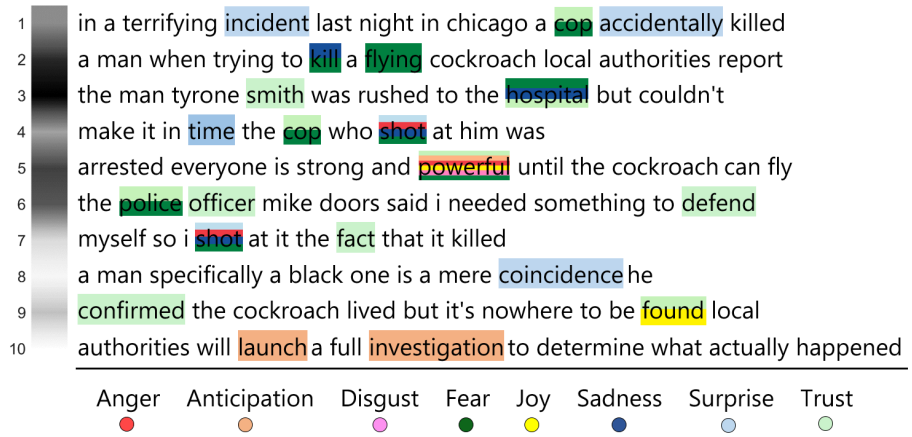


Figure 7.7: Emotional interpretation of a *fake* news article by showing the attention weights (the bar on the left) and highlighting the emotions in the text.

show a clear example of how fake news articles try to manipulate the reader. It looks that the existence of *fear, sadness*, and *surprise emotions* at the beginning of the article is the cause of the attention on this part. On the other hand, at the end of the article, we can notice that such negative emotions do not exist, while *emotions* like *joy* and *anticipation* appear. This shows how fake news try to attract the readers' attention in the first part of the text. Regarding *morality* features, we only match the word "kill" with the *harm* category. Also, for the *hyperbolic* feature, we match the words "terrifying" and "powerful". In the same manner, both *morality* and *hyperbolic* features match words that occur at the beginning of the article. Lastly, for both *sentiment* and *imageability* features, we are not able to find a clear interpretation in this example where many words across the segments match with each feature.

Real vs. Fake In Table 7.9 we present an analysis on both real and fake news articles. The analysis gives an intuition to the reader on the distribution of the used features across the articles' segments. It shows that an emotion like *fear* has a higher difference in fake news between $\mu\ first_{seg.}$ and $\mu\ last_{seg.}$, than in real ones. Also, a feature like *hyperbolic*, has a higher $\mu\ all_{seg.}$ in fake news than real news, with a lower $\sigma\ all_{seg.}$; this indicates that fake news have a higher amount of hyperbolic words with similar high values.

tion types in the NRC lexicon.

	Features	Real News				Fake News			
		$\mu\ first_{seg.}$	$\mu\ last_{seg.}$	$\mu\ all_{seg.}$	$\sigma\ all_{seg.}$	$\mu\ first_{seg.}$	$\mu\ last_{seg.}$	$\mu\ all_{seg.}$	$\sigma\ all_{seg.}$
Emotions	Anger	0.175	0.167	0.170	0.003	0.183	0.170	0.171	0.008
	Anticipation	0.301	0.315	0.264	0.025	0.293	0.305	0.260	0.022
	Disgust	0.095	0.101	0.095	0.004	0.096	0.091	0.091	0.007
	Fear	0.254	0.250	0.238	0.010	0.265	0.226	0.238	0.011
	Joy	0.217	0.226	0.183	0.021	0.207	0.203	0.175	0.020
	Sadness	0.161	0.158	0.160	0.006	0.155	0.155	0.158	0.007
	Surprise	0.140	0.144	0.123	0.012	0.142	0.123	0.120	0.008
	Trust	0.446	0.466	0.400	0.031	0.461	0.421	0.401	0.029
Sentiment	Positive	0.599	0.623	0.558	0.030	0.608	0.591	0.554	0.032
	Negative	0.369	0.337	0.347	0.011	0.367	0.336	0.350	0.013
Morality	Harm	0.007	0.011	0.007	0.002	0.008	0.013	0.007	0.002
	Care	0.026	0.023	0.019	0.004	0.021	0.022	0.019	0.003
	Fairness	0.003	0.013	0.007	0.002	0.005	0.020	0.009	0.004
	Unfairness	0.000	0.000	0.001	0.000	0.001	0.000	0.001	0.001
	Loyalty	0.016	0.017	0.019	0.002	0.014	0.016	0.019	0.003
	Betrayal	0.004	0.003	0.005	0.001	0.002	0.003	0.004	0.001
	Authority	0.025	0.032	0.026	0.003	0.024	0.028	0.026	0.002
	Subversion	0.005	0.004	0.004	0.001	0.006	0.007	0.005	0.002
	Sanctity	0.005	0.005	0.004	0.001	0.005	0.006	0.005	0.002
	Degradation	0.003	0.004	0.003	0.001	0.006	0.004	0.003	0.001
Imageability	Imageability	0.845	1.203	1.144	0.122	0.877	1.184	1.145	0.124
	Abstraction	0.424	0.331	0.352	0.028	0.382	0.304	0.342	0.037
	Hyperbolic	0.042	0.05	0.045	0.005	0.046	0.044	0.047	0.003

Table 7.9: A quantitative analysis of the features across articles' segments. We present the average value in the first segment ($\mu\ first_{seg.}$), the average value in the last segment ($\mu\ last_{seg.}$), the average value in the all 10 segments ($\mu\ all_{seg.}$), and the standard deviation ($\sigma\ all_{seg.}$) of a feature across the 10 segments, both in real and fake news. The values are represented as percentage values.

7.4 False Information Spreaders

Suspicious online users have fostered the spread of false information online, especially in social media networks. To raise the awareness of online users regarding false information, fact checkers started to spread anti false information messages to prevent further dissemination. In this section, we present some experiments that we have done in an attempt to discriminate between fake news spreaders and fact checkers, considering linguistic features and also the information about the personality of the authors. Moreover, we briefly present an overview of the shared task that we organized[15] to profile fake news spreaders on Twitter.

7.4.1 Fake News Checkers vs. Spreaders

In this sub-section we propose the CheckerOrSpreader model that can classify a Twitter user as a potential fact checker or a potential fake news spreader. Our model is based on a Convolutional Neural Network (CNN) and combines word embeddings with features that represent users' personality traits and linguistic patterns used in their tweets [87].

7.4.1.1 Model

The proposed model is based on a CNN. The architecture of the CheckerOrSpreader system is depicted in Figure 7.8. CheckerOrSpreader consists of two different components, the word embeddings and the user's psycho-linguistic component. The embeddings component is based on the tweets that users have posted on their timeline. The psycho-linguistic component represents the psychometric and linguistic style patterns and the personality traits that were derived from the textual content of the posts.

To extract the linguistic patterns and the personality traits we use the following approaches:

- **Linguistic patterns**: For the linguistic patterns, we employ LIWC [162] that is a software for mapping text to 73 psychologically-meaningful linguistic categories[16]. In particular, we extract pronouns (I, we, you, she/he, they), personal concerns (work, leisure, home, money, religion, death), time focus (past, present, future), cognitive processes (causation, discrepancy, tentative, certainty), informal language (swear, assent, nonfluencies, fillers), and affective processes (anxiety).

- **Personality scores**: The Five-Factor Model (FFM) [116], also called the Big Five, constitutes the most popular methodology used in auto-

[15] https://pan.webis.de/clef20/pan20-web/author-profiling.html

[16] For a comprehensive list of LIWC categories see: http://hdl.handle.net/2152/31333

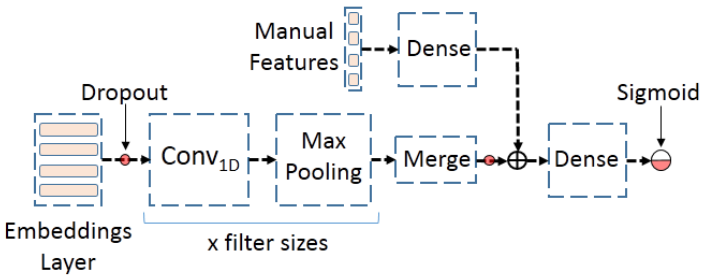


Figure 7.8: Architecture of the CheckerOrSpreader model.

matic personality research [152]. In essence, it defines five basic *factors* or *dimensions* of personality. These factors are:

- *openness to experience* (unconventional, insightful, imaginative)
- *conscientiousness* (organised, self-disciplined, ordered)
- *agreeableness* (cooperative, friendly, empathetic)
- *extraversion* (cheerful, sociable, assertive)
- *neuroticism* (anxious, sad, insecure)

Each of the five factors presents a *positive* and a complementary *negative* dimension. For instance, the complementary aspect to neuroticism is defined as *emotional stability*. Each individual can have a combination of these dimensions at a time. To obtain the personality scores, we followed the approach developed by Neuman and Cohen [153]. They proposed the construction of a set of vectors using a small group of adjectives, which according to theoretical and/or empirical knowledge, encode the essence of personality traits and personality disorders. Using a context-free word embedding they measured the semantic similarity between these vectors and the text written by different individuals. The similarity scores derived, allowed to quantify the degree in which a particular personality trait or disorder was evident in the text.

7.4.1.2 Collected Dataset

To build our collection, we first collect articles that have been debunked as fake from the Lead Stories website[17]. Crawling articles from fact check websites is the most popular way to collect articles since they are already labeled by experts. This approach has been already used by other researchers in order to create collections [201]. In total, we collected 915 titles of articles that have been labeled as fake by experts. Then, we removed stopwords

[17] https://leadstories.com/

Table 7.10: Titles of the articles with the highest and lowest number of tweets.

Titles of articles with the highest number of tweets	**Titles of articles with the lowest number of tweets**
1. Doctors Who Discovered Cancer Enzymes In Vaccines NOT All Found Murdered	1. Make-A-Wish Did NOT Send Terminally Ill Spider-Man To Healthy Kid
2. Sugar Is NOT 8 Times More Addictive Than Cocaine	2. Man Did NOT Sue Radio Station For Playing Despacito 800 Times A Day
3. George H.W. Bush Did NOT Die at 93	3. Man-Eating Shark NOT Spotted In Ohio River
4. NO Alien Invasion This Is NOT Real	4. FBI DID NOT Classify President Obama As A Domestic Terrorist

from the headlines and we used the processed headlines to search for relevant tweets.

To extract the tweets we use Twitter API. In total we collected 18,670 tweets that refer to the articles from Lead Stories. For some of the articles we managed to collect a high number of tweets, whereas other articles were not discussed a lot in Twitter. Table 7.10 shows examples of the articles for which we collected the highest and the lowest number of tweets. From this table, we observe that the most popular article was about a medical topic and for which we collected 1,448 tweets.

The tweets that we collected can be classified in two categories. The first category contains tweets that debunk the original article by claiming its falseness (fact check tweet), and usually citing one of the fact-checking websites (snopes, politifact or leadstories). The second category contains tweets that re-post the article (spreading tweet) implying its truthfulness. To categorise the tweets into fact check and spreading tweets, we follow a semi-automated process. First, we manually identify specific patterns that are followed in the fact check tweets. According to those rules, if a tweet contains any of the terms {hoax, fake, false, fact check, snopes, politifact leadstories, lead stories} is a fact check tweet, otherwise it is a spreading tweet.

Figure 7.9 shows some examples of articles debunked as fake together with fact check and spreading tweets. We notice that in the fact check tweets we have there are terms such as fake, false and fact check, whereas in the spreading tweets we have re-posts of the specific article. Then, we manually checked a sample of the data to check if there are any wrong annotations. We manually checked 500 tweets and we did not find any cases of misclassification.

After the annotation of the tweets, we annotate the authors of the tweets as checkers or spreaders based on the number of fact check and spreading

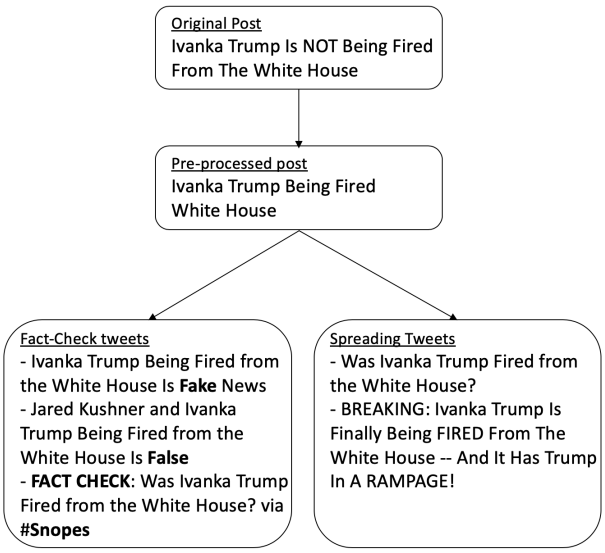


Figure 7.9: Examples of fact check and spreading tweets.

tweets they posted. In particular, if a user has both fact check and spreading tweets, then we consider that this user belongs to the category for which s/he has the larger number of tweets. Finally, we collect the timeline tweets that the authors have posted to create our collection. In total, our collection contains tweets posted by 2,357 users, of which 454 are checkers and 1,903 spreaders.

7.4.1.3 Experiments and Results

For our experiments, we use 25% of our corpus of users for validation, 15% for test and the rest for the training. We initialize our embedding layer with the 300-dimensional pre-trained GloVe embeddings [164]. We allow the used embeddings to be tuned during the training process to fit more our training data. It's worth to mention that at the beginning of our experiments, we tested another version of our system by replacing the CNN with an Long Short-Term Memory (LSTM) network. The overall results showed that the CNN performs better for the particular task. To find the best parameters of the different approaches on the validation set, we use the hyperopt library[18]. For the evaluation, we use macro-F1 score.

We use the following baselines to compare our results:

- *SVM+BoW* is based on a Support Vector Machine (SVM) classifier trained on bag of words using Term Frequency - Inverse Document Frequency (Tf-Idf) weighting scheme.

[18] https://github.com/hyperopt/hyperopt

Table 7.11: Performance of the different systems on the fact checkers detection task.

Model	F1-score
SVM+BoW	0.48
USE	0.53
LR+emotion	0.45
LR+sentiment	0.44
LR+LIWC	0.50
LR+personality	0.44
LSTM	0.44
CNN	0.54
CNN+LIWC	0.48
CNN+personality	0.57
CheckerOrSpreader	0.59

- *Logistic Regression* trained on the different linguistic and personality scores features. In particular, we tried sentiment, emotion, LIWC and personality traits. For emotions we use NRC emotions lexicon [148] and we extracted anger, anticipation, disgust, fear, joy, sadness, surprise, and trust. We use the same lexicon to estimate the positive and negative sentiment in users' tweets.

- *Universal Sentence Encoder (USE)* [37]: For the USE baseline, we represent the final concatenated documents (tweets) using USE embeddings[19].

- *LSTM*: is based on a LSTM network with Glove pre-trained word embeddings for word representation.

- *CNN*: is a CNN with Glove pre-trained word embeddings for word representation.

Table 7.11 shows the results of our experiments. We observe that CNN performs better than LSTM when they are trained only using word embeddings. In particular, CNN outperforms LSTM by 20.41%. Also, we observe that Logistic Regression achieves a low performance when it is trained with the different psycho-linguistic features. The best performance regarding Logistic Regression is achieved with the linguistic features extracted with LIWC.

From Table 7.11 we also observe that combining CNN with the personality traits leads to a higher performance compared to combining CNN with the LIWC features. In particular, CNN+personality outperforms CNN+LIWC

[19] https://tfhub.dev/google/universal-sentence-encoder-large/3

by 17.14%. This is an interesting observation that shows the importance of considering personality traits of users for their classification in checkers and spreaders. Also, the results show that CheckerOrSpreader (CNN+personality+LIWC) achieves the best performance. In particular, CheckerOrSpreader manages to improve the performance by 8.85% compared to the CNN baseline and by 3.45% compared to the CNN+personality version.

7.4.2 Online Trolls

In this sub-section we present a case study of fake news spreaders, namely trolls, that were unmasked after the 2016 US elections. We propose a text-based approach that uses topic-based and style based features to detect those trolls. We also present two classification models that employ the proposed features to identify trolls out of regular users [77].

7.4.2.1 Models

In order to identify IRA trolls, we use a rich set of textual features. With this set of features we aim to model the tweets of the accounts from several perspectives.

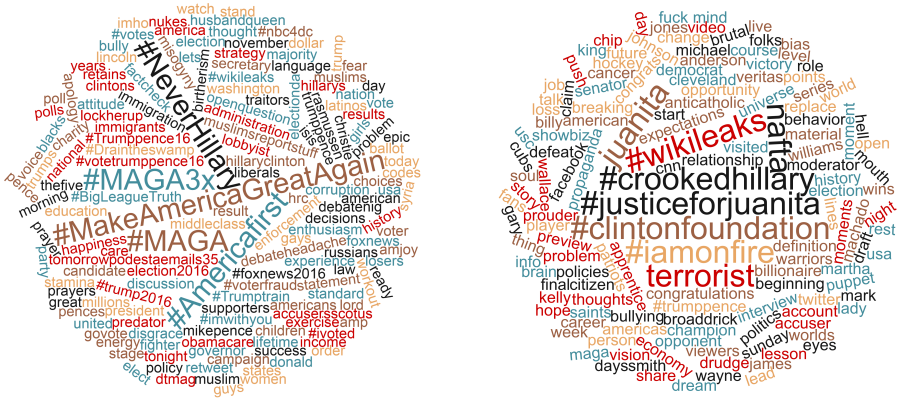
Topic Information: Previous work [154] have investigated IRA campaign efforts on Facebook, and they found that IRA pages have posted more than ∼80K posts focused on divisive issues in US. Later on, the work in [29] has analyzed Facebook advertised posts by IRA and they specified the main topics that these advertisements discussed. Given the results of the previous works, we applied a topic modeling technique, namely Latent Dirichlet Allocation (LDA) [25], on our dataset to extract its main topics. We aim to detect IRA trolls by identifying their suspicious ideological changes across a set of topics.

Given our dataset (see Section 7.4.2.2), we applied LDA on the tweets after a prepossessing step where we maintained only nouns and proper nouns using the SpaCy part-of-speech (POS) tagger, which is an off-the-shelf POS tagger[20]. In addition, we removed special characters (except HASH "#" sign for the hashtags) and lowercase the final tweet. To ensure the quality of the topics, we removed the hashtags we used in the collecting process where they may bias the modeling algorithm. We tested multiple numbers of topics and finally we use seven. We manually observed the content of these topics to label them. The extracted topics (T) are: *Police shootings, Islam and War, Trump, Black People, Civil Rights, Hillary,* and *Crimes*. In some topics, like *Trump* and *Hillary*, we found contradicted opinions, in favor and against the main topics, but generally we can notice that the *Trump* topic has a support stance to Trump, on the other hand, the *Hillary* topic has an against stance towards Hillary (see Figure 7.10 for the frequency-based wordcloud). Also,

[20] https://spacy.io/models

the topics *Police Shooting* and *Crimes* are similar, but we found that some words such as: *police, officers, cops, shooting, gun, shot, etc.* are the most discriminative between these two topics. In addition, we found that the *Crimes* topic focuses more on raping crimes against children and women. Our resulted topics are generally consistent with the ones obtained from the Facebook advertised posts in [29], and this emphasizes that IRA efforts organized in a similar manner in both social media platforms.

Based on our topic information, we model the users textual features w.r.t. each of these topics. In other words, we model a set of textual features which could change in the users' tweets across the topics. We aim at modeling the trolls manipulating effort in which they interact in a different way with each topic; e.g. a troll account may trigger positive emotions in a set of topics in favor of and negative if against. Similarly, showing supporting stance intensively in some topics and denial stance in others. Thus, we used LDA to annotate the tweets of the users in one of the T topics to capture the changes of the following proposed features among the topics.



(a) (b)

Figure 7.10: (a) *Trump* and (b) *Hillary* topics word clouds.

We chose the following lexical features under the assumption that they may characterise the trolls' language changes across the topics. We use term frequency representation to extract the following features from the tweets:

- **Emotions**: Since the results of the previous works [29, 154] showed that IRA efforts engineered to seed discord among individuals in US, we use emotions features to detect their emotional attempts to manipulate the public opinions (e.g. fear spreading behaviour). For that, we use the NRC emotions lexicon [148] that contains ∼14K words labeled using the eight Plutchik's emotions (*8 Features*).

- **Sentiment**: We extract the sentiment of the tweets from NRC [148], *positive* and *negative* (*2 Features*).

- **Bad & Sexual Cues**: During the manual analysis of a sample from IRA tweets, we found that some users use bad words to mimic the language of a US citizen. Thus, we model the presence of such words using a list of bad and sexual words from [73] (*2 Features*).

- **Stance Cues**: Stance detection has been studied in different contexts to detect the stance of a tweet reply with respect to a main tweet/thread [146]. Using this feature, we aim to detect the stance of the users regarding the different topics we extracted. To model the stance we use a set of stance lexicons employed in previous works [9, 78] . Concretely, we focus on the following categories: *belief, denial, doubt, fake, knowledge, negation, question,* and *report* (*8 Features*).

- **Bias Cues**: We rely on a set of lexicons to capture the bias in text. We model the presence of the words in one of the following cues categories: *assertives verbs* [109] , *bias* [187] , *factive verbs* [125] , *implicative verbs* [123] , *hedges* citehyland2018metadiscourse , *report verbs* . A previous work has used these bias cues to identify bias in suspicious news posts in Twitter [226] (*6 Features*).

- **LIWC**: We use a set of linguistic categories from the LIWC linguistic dictionary [217]. The used categories are: *pronoun, anx, cogmech, insight, cause, discrep, tentat, certain, inhib, incl*[21] (*10 Features*).

- **Morality**: Cues based on the morality foundation theory [97] where words labeled in one of a set of categories: *care, harm, fairness, unfairness, loyalty, betrayal, authority, subversion, sanctity,* and *degradation* (*10 Features*).

Profiling IRA Accounts: As Twitter declared, although the IRA campaign originated in Russia, it has been found that IRA trolls concealed their identity by tweeting in English. Furthermore, for any possibility of unmasking their identity, the majority of IRA trolls changed their location to other countries, as well as, the language of the Twitter interface they use. Thus, we propose the following features to identify these users using only their tweets text:

- **Native Language Identification (NLI)**: This feature was inspired by earlier works on identifying native language of essays writers [141]. We aim to detect IRA trolls by identifying their way of writing English tweets. As shown in [226], English tweets generated by non-English speakers have a different syntactic pattern. Thus, we use SOTA NLI features to detect this unique pattern [45, 142, 96]. The feature set consists of bag of: stopwords (*179 Features*), POS tags (*46 Features*), and

[21] Total pronouns, Anxiety, Cognitive processes, Insight, Causation, Discrepancy, Tentative, Certainty, Inhibition, and Inclusive respectively.

syntactic dependency relations (DEPREL) (*45 Features*). We extract the POS and the DEPREL information using spaCy. To normalize the tweets, we clean them from the special characters and maintain dots, commas, and first-letter capitalization of words. We use regular expressions to convert a sequence of dots to a single dot, and similarly for sequence of characters (in total *270 Features*).

- **Stylistic**: We extract a set of stylistic features following previous work in the authorship attribution domain [248, 21, 211], such as: the count of special characters, consecutive characters and letters[22], URLs, hashtags, users' mentions. In addition, we extract the uppercase ratio and the tweet length (*8 Features*).

Given the above two sets of features, we use them in two different approaches in order to build trolls detectors. The proposed approaches utilize a classical machine learning classifier and a CNN:

All Features + LG: In this approach, we model the extracted textual features as follows: Given V_n as the concatenation of the previous 46 topic information features of a tweet n, we represent each user by considering the *average* and *standard deviation* of her tweets' $V_{1,2,..N}$ in each topic t independently. We concatenate the final vectors; final vectors are seven since the number of topics (T) is equals seven in our case. Mathematically, the final feature vector of a user x is defined as follows:

$$user_x = \bigodot_{t=1}^{T} \left[\frac{\sum_{n=1}^{N_t} V_{nt}}{N_t} \odot \sqrt{\frac{\sum_{n=1}^{N_t} (V_{nt} - \overline{V_t})}{N_t}} \right] \tag{7.1}$$

where given the tth topic, N_t is the total number of tweets of the user (annotated with the tth topic), V_{nt} is the nth tweet feature vector, $\overline{V_t}$ is the mean of the tweets' feature vectors; $\odot$ represents the vectors concatenation process.

Regarding the profiling features, we represent each user by considering the *average* and the *standard deviation* of her tweets' feature vectors, similar to the representation of the previous features but without considering the topic information. In short, we apply the *average* and the *standard deviation* on all the tweets of a user at once:

$$user_x = \frac{\sum_{n=1}^{N} V_n}{N} \odot \sqrt{\frac{\sum_{n=1}^{N} (V_n - \overline{V})}{N}} \tag{7.2}$$

where N is her total number of tweets, V_n is the nth tweet feature vector, $\overline{V}$ is the mean of the tweets feature vectors of a user x. After preparing the

[22] We considered 2 or more consecutive characters, and 3 or more consecutive letters.

two feature set vectors, we concatenate them, and we feed them to a Logistic Regression (LG) classifier.

CNN: We use a CNN to model the proposed features. We use a CNN that has two branches: one models the topic information (A) and the other models the profiling features (B). Figure 7.11 shows the proposed network.

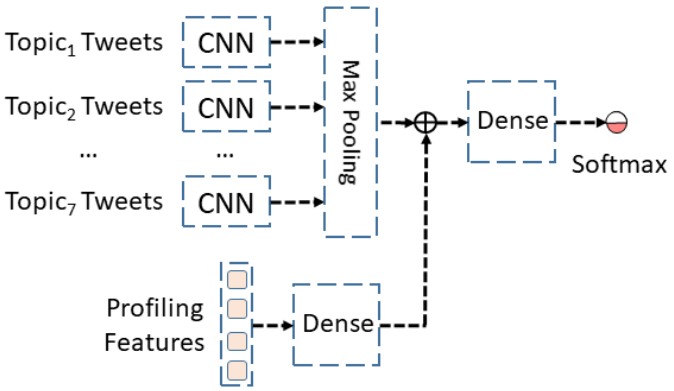


Figure 7.11: The CNN structure.

In branch A, first we divide a user's tweets into seven tweets' groups based on their topics and then we feed each group to a different CNN. The tweets of a specific group are considered as one long document. Each CNN applies a convolution and max-pooling layers. The input document D of length n is represented as $[D_1, D_2..D_n]$ where $D_n \in \mathbb{R}^d$; $\mathbb{R}^d$ is a d-dimensional one-hot vector of the i-th word in the input document. The words' d-dimensional vectors have a length of 46, that is, the total number of topic information features. After processing the input group of tweets, we apply another max-pooling layer to extract the important global features from the seven topics' CNNs. The structure of this branch is inspired by the Hierarchical Attention model [240] that has been proposed for document classification.

On the other hand, for branch B we concatenate all tweets of a user into one document, and we use the Equation 7.2 to extract a vector of the profiling features (length of 278) and we feed it to a dense layer $f(W_a v + b_a)$, where W_a and b_a are the corresponding weight matrix and bias terms, and f is an activation function, such as *ReLU*, *tanh*, etc.

After processing the input tweets in both branches, we concatenate the output vectors ($\oplus$) and we feed them to another dense layer to learn their joint interaction. Finally, to get the classes probability of a document, we add a Softmax layer.

7.4.2.2 Dataset

To model the detection of the IRA trolls, we considered a large dataset

Table 7.12: Statistics of the dataset.

	IRA Trolls	**Regular Accounts**
Total # of Accounts	2,023	94,643
Total # of Tweets	$\sim$ 1.8 M	$\sim$ 1.9 M
Avg. # of Tweets	357	19
Avg. # of Followers	1,834	9,867
Avg. # of Followees	1,025	2,277

of both regular users (legitimate accounts) and IRA troll accounts. In the following we describe the dataset. In Table 7.12 we summarizes its statistics.

Russian Trolls (IRA): We used the IRA dataset[23] that was released by Twitter after identifying the Russian trolls. The original dataset contains 3,841 accounts, but we use a lower number of accounts and tweets after filtering them: We focus on accounts that use English as the main language. In fact, our goal is to detect Russian accounts that mimic a regular US user. Then, we remove from these accounts non-English tweets, and maintain only tweets that were tweeted originally by them. Our final IRA accounts list contains 2,023 accounts.

Regular Accounts: To contrast IRA behaviour, we sampled a large set of accounts to represent the ordinary behaviour of accounts from US. We collected a random sample of users that they post at least 5 tweets between 1^{st} of August and 31 of December, 2016 (focusing on the US 2016 debates: first, second, third and vice president debates and the election day) by querying Twitter API hashtags related to the elections and its parties (e.g #trump, #clinton, #election, #debate, #vote, etc.). In addition, we selected the accounts that are located within US and use English as language of the Twitter interface. We focus on users during the presidential debates and elections dates because we suppose that the peak of trolls efforts concentrated during this period. The final dataset is totally imbalanced (2% for IRA trolls and 98% for the regular users). This class imbalance situation represent a real scenario. From Table 7.12, we can notice that the number of total tweets of the IRA trolls is similar to the one obtained from the regular users. This is due to the fact that IRA trolls were posting a lot of tweets before and during the elections in an attempt to try to make their messages reach the largest possible audience.

7.4.2.3 Experiments and Results

In order to evaluate our approach, we use the following baselines:

- **BOW + LR**: We use bag-of-words (BOW) representation (weighted

[23] https://about.twitter.com/en_us/values/elections-integrity.html

using TF-IDF scheme) with a LR classifier where we aggregate all the tweets of a user into one long document. We aim to assess how a simple word-based model can perform.

- **LSTM**: Word embeddings-based models showed significant improvements in many tasks previously. We use Long short-term memory (LSTM) network with *Glove (840b.300d) words embeddings*. Similar to BOW baseline, we we aggregate all the tweets of a user into one long document.

- **Number of Tweets + NB**: Based on the dataset statistics (see Table 7.12), we can notice that the IRA accounts have a large amount of tweets. Thus, as a baseline, we use the number of tweets for each account and we feed them to a NB classifier. We use this baseline to investigate if it is possible to detect the trolls accounts using only the number of tweets.

- **Tweet2vec + LR**: A previous work [26] showed that IRA trolls were playing a hashtag game which is a popular word game played on Twitter, where users add a hashtag to their tweets and then answer an implied question [105]. IRA trolls used this game in a similar way but focusing more on offending or attacking the targeted section of the audience; an example from IRA tweets:

  > *#OffendEveryoneIn4Words undocumented immigrants are ILLEGALS*

  Thus, we use as a baseline *Tweet2vec* [61] which is a character-based Bidirectional Gated Recurrent neural network that reads tweets and predicts their hashtags. We aim to assess if the tweets hashtags can help identifying the IRA tweets. The model reads the tweets in a form of character one-hot encodings and uses them for training with their hashtags as labels. To train the model, we use our collected dataset which consists of ∼3.7M tweets[24]. To represent the tweets in this baseline, we use the decoded embedding produced by the model and we feed them to a LR classifier.

- **Network Features + LR**: IRA dataset provided by Twitter contains few information about the accounts details, and they are limited to: profile description, account creation date, number of followers and followees, location, and account language. Therefore, as a baseline we use the number of followers and followees to assess their identification performance. We feed these features to a LR classifier.

[24] We used the default parameters that were provided with the system code.

- **Botometer + RF**: *Botometer* is the SOTA bots detection system, which uses content, sentiment, friend, network, temporal, and user features. We extract these features and feed them to a Random Forest (RF) classifier with 100 as the number of estimators following the authors setup.

- **Still Out There + ABDT**: Also as a baseline, we use the available proposed model in the related work [113], which uses profile, language distribution, and stop-words usage features with an Adaptive Boosted Decision Trees (ABDT) classifier.

Table 7.13 presents the classification results of the baselines and our approaches. We report the results of our classical classifier -based approach with top 3 performing classifiers (RF, NN, and LR). The best results in terms of F1 score obtained with the LR classifier. The results show that both proposed models perform best comparing to the used baselines. Also, the results show that the *All Features + LG* model performs better than the *CNN* with a noticeable difference in terms of F1 measure. Generally, we can notice that we are able to detect the IRA trolls effectively using textual features (RQ1).

The *topic features* have a good performance comparing to most baselines. The result obtained with the *Profiling* features is interesting; we are able to detect the IRA trolls from the users' writing style with an F1 value of 0.88 using the *All Features + LG* model. To assess whether the topic information improves the performance of each of the lexical features, we run the *All Features + LG* model with each feature independently, with and without utilizing the topic information (without considering the topics in Eq. 7.1). Following, we present the results obtained with each feature: Emotions (+0.74|-0.02)[25]; Sentiment (+0.28|-0.0); Bad & Sexual (+0.58|-0.0); Stance Cues (+0.72|-0.12); Bias Cues (+0.73|-0.03); LIWC (+0.71|-0.04), and Morality (+0.72|-0.36). We conclude from these results that the model weakly detects the changes in stances, variations in emotions, etc., for a user when we discard the topic information. Clearly, we can notice that the model became aware of the flipping behaviour across the topics. These results emphasize the importance of the topic information (RQ2), especially with the emotions. This motivates us to analyze further the emotions in the IRA tweets (see the following section). Finally, the baselines' results show us that the *Network features* are not able to detect the IRA trolls. A previous work [241] showed that the IRA trolls tend to follow many users, and nudging other users to follow them (e.g., by writing "follow me" in their profile description) to hide their identity (account information) with the regular users. Finally, similar to the *Network features*, the *Tweet2vec* baseline performs poorly. This indicates that, although the IRA trolls used the hash-

[25](+) stands for the F1 result with the topic information and (-) without them.

Table 7.13: Classification results.

Method	**Precision**	**Recall**	**F1**
Network Features + LR	0.0	0.0	0.0
Random Selection	0.02	0.5	0.04
Tweet2vec + LR	0.18	0.64	0.28
Number of Tweets + NB	0.47	0.53	0.5
BOW + LR	0.86	0.51	0.64
LSTM	0.86	0.69	0.76
Still Out There + ABDT [113]	0.97	0.75	0.84
Botometer + RF	**0.99**	0.76	0.86
Topic Information Features			
Topic-based Features + LR	0.89	0.7	0.78
CNN (branch A)	0.79	0.81	0.80
Profiling Features			
Profiling Features + LR	0.92	0.85	0.88
CNN (branch B)	0.81	0.88	0.84
All Features			
All Features + RF	**0.99**	0.78	0.88
All Features + NN	0.90	0.89	0.90
All Features + LR	0.93	0.88	**0.91**
CNN	0.86	**0.90**	0.88

tag game extensively in their tweets, the *Tweet2vec* baseline is not able to identify them. The results of both *Botometer* and *Still Out There* [113] are superior to the other baselines, but still lower comparing to our proposed approaches.

7.4.3 Detecting Conspiracy Propagators using Psycho-linguistic Characteristics

In this subsection we focus on the role of users in the propagation of conspiracy theories that is a specific type of disinformation. We compare psycho-linguistic patterns of online users that tend to propagate posts that support conspiracy theories and of those who propagate posts that refute them. To this end, we perform a comparative analysis over various psychological and linguistic characteristics using social media texts of the users that share posts about conspiracy theories. Then, we compare the effectiveness of those characteristics for predicting if a user tends to share posts that support conspiracy theories. In addition, we propose ConspiDetector, a model that is based on a CNN and which combines word embeddings with psycho-linguistic characteristics extracted from the tweets of users to differentiate between users that tend to share posts that support conspiracy theories and those who tend to share posts that refute them.

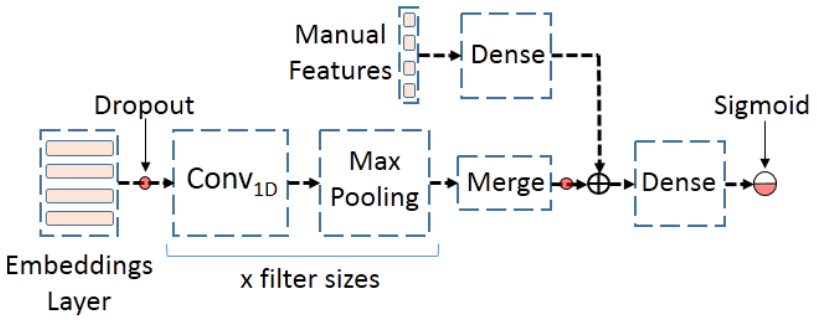


Figure 7.12: Architecture of ConspiDetector.

7.4.3.1 Methodology

In this subsection we present our proposed model, called ConspiDetector, that is based on a CNN and psycho-linguistic features that are extracted from the users' tweets with the aim to classify a user as conspiracy or anti-conspiracy propagator. The model consists of two branches, a content-based and a lexicon-based. The content-based consists of an embeddings layer followed by convolutional, max pooling, and dense layers as shown in Figure 7.12.

Since the classification task is binary conspiracy propagators vs. anti-conspiracy propagators), as output we use a Sigmoid layer. In our implementation we use dropout both after the embeddings layer and before the dense layer. With regards to multiple sizes of convolutional filters, we concatenate their outputs in one vector after the max pooling layer. To feed user's tweets to this branch, we concatenate all of her tweets into a single document. It is worth to mention that we choose CNN rather than Long Short Term Memory (LSTM) network since the process of concatenating all the tweets discards the sequential nature of the input document. This was also confirmed from the fact that when we applied LSTM on our data we obtained a lower performance compared to CNN.

The second branch is based on the four groups of psycho-linguistic features (i.e., personality traits, emotions, sentiment and linguistic patterns) extracted from the user's tweets. Given a tweet of a user, first we count how many words from the categories/lexicons appear in that tweet (count frequency feature vector). We do this for all the tweets of the user, and then we calculate an average vector for that user by summing the tweets vectors and dividing them by the number of tweets. The final averaged vector is fed into the second branch of ConspiDetector.

Table 7.14: Hashtags used to collect the tweets and statistics about the collection.

	Pro-conspiracy	**Anti-conspiracy**
Hashtags	#vaccinesCauseAutism #antiVax #climateChangeIsNotReal #flatEarth #nasaLies #nasaFake #spaceIsFake #moonLandingFake #bigPharmaFraud #ebolaconspiracy #antiFluoridation	#vaccinesWork #vaccinessavelives #climateChangeIsReal #earthisnotflat #nasatruth #nasaIRreal #spaceIsReal #moonlandingisreal
Users	977	950
Tweets	912,735	992,798

7.4.3.2 Dataset

Several collections have been developed for the task of fake news detection [232, 201]. The majority of these datasets contain fake and real articles and can be used for the evaluation of systems developed to detect fake news. However, to the best of our knowledge, there is no available collection that has focused on the conspiracy theories and that could be used for the classification of users into conspiracy and anti-conspiracy propagators. Therefore, we developed our own dataset.

To create our data collection, we used Twitter API and we collected tweets about some of the most well-known conspiracy theories[26]. Table 7.14 shows the list of hashtags that we used to collect tweets that refer to conspiracy theories as well as some statistics with regard to the collection. Initially, we collected 25,975 tweets using the hashtags shown in Table 7.14. At this stage, we had two groups of hashtags, those that are likely used to support a conspiracy and those that refute them. We should note that these hashtags were used only to collect the initial set of tweets that refer to conspiracy theories and were not used for the final annotation of the tweets as supporting/refuting a conspiracy theory. For every hashtag that supports a conspiracy theory, we tried to have one that likely refutes the conspiracy (e.g., #vaccinesCauseAutism vs #vaccinesWork). However, this was not possible for all the hashtags.

Since we focus on user level, next we take users that have posted between 2 and 10 tweets in any of those hashtags of each group. This step filters out users that are posting a lot of tweets and which are likely bots. Given that the hashtags do not always reflect the content of the tweet, we decided

[26] https://en.wikipedia.org/wiki/List_of_conspiracy_theories

Table 7.15: Examples of tweets that support and refute a conspiracy theory.

Tweets that support a conspiracy theory	Tweets that refute a conspiracy theory
- My babies will grow up to be healthier than all of you because I'm not injecting that poison into them **#antivax**	- DID YOU KNOW? being **#antivax** #antivaxx decreases the chances of your child surviving adulthood.
- Spread my newborn with peanut butter, better than vaxxing that little thing! **#antivax** #vaxxer #antivaxxer #antivacccine_movement	- **#Antivax** groups spreading lies to immigrant communities & harming children are despicable. #vaccineswork
- I don't vaccinate my children but one of them caught measles and the school sent him home! Really rude of them to try and compromise his education!! #DoctorsUnderOpression #antivaccine_movement **#antivax** #angry #EducationFest	- #Measles Outbreak in Minnesota, US, Caused by **#antivax** Campaign, Officials Say http://www.livescience.com/59105-measles-outbreak-minnesota.html #antivaxx #vaccineswork
- Hoping to learn more about the dangers of vaccines and ways to keep my babies healthy. I'll never let some doctor inject poison into them! But this is all new for me, so any advice from other parents on ways to keep them healthy would be appreciated! Thank you! **#antivax**	- You can't get autism from a vaccine. Antivaxxers clearly don't understand how the brain works. #vaccinate #vaccines **#antivax** #vaccinateyourkids #antivaxxersareidiots #vaccineswork #measles #measlesoutbreak
- I mean really, NASA lost the original tapes, telemetry data AND specs for how to get back to the moon?? I guess [they] figured if we fell for this contraption,we would fall for anything. **#SpaceIsFake** #MoonLandingHoax	- If space is fake, how can we experience night and day? Solar and lunar eclipses? The northern lights? Stars and planets? Solar storms? Sunburns? Comets and asteroids. The moon? **#spaceisfake**
- Still think the images brought to you by NASA and the other space agencies are real? Many are waking up to the deception. **#FlatEarth** #SpaceIsFake #EarthIsNotAGlobe	- Then why is it so hard to produce a working map? Or to find inaccuracies in the globe map! That alone already proves the earth is a globe! **#flatearth** fail!

to manually annotate those tweets to reassure if they indeed support or refute the conspiracy theory. In total, we manually annotated 6,385 tweets. The manual annotation was necessary since some of the hashtags are used to support as well as to refute a conspiracy theory. During the manual annotation, we noticed that the *#flatEarth* and *#antiVax* hashtags were the most controversial ones in the means that they were used to support as well as to refute the conspiracy. Table 7.15 shows some examples of tweets that support and refute a conspiracy theory. Some of those examples refer to vaccination and others to the flat earth conspiracy theory. For example, we observe that in the first tweet that supports the antiVax theory the user is referring to the vaccines as a poison that do not want to give it to his/her children. It is clear that the user believes that vaccination can be harmful. On the contrary, the first tweet in the refuting column is against the theory by saying that if you believe in this theory you "decrease the chances of your child surviving adulthood".

After the annotation of the tweets as *supporting*, *refuting* or *uncertain* to a conspiracy theory, we proceed with the annotation of the users. Let $u_{support}$ and u_{refute} be the number of tweets that a user u posted and which support or refute any of the conspiracy theories respectively. Then for every user we calculated the ratio of tweets that support a conspiracy as $u_{ratio} = u_{support}/(u_{support} + u_{refute})$. In case u_{ratio} was larger than 0.5 the user was annotated as a *conspiracy propagator*, otherwise the user was annotated as *anti-conspiracy propagator*.

After the annotation at the user level, we randomly selected 977 conspiracy propagators and 950 anti-conspiracy propagators. Finally, we collected the 1,000 most recent tweets of those users. Our analysis is based on those tweets that refer to 912,735 and 992,798 tweets for conspiracy and anti-conspiracy propagators, respectively. Here, we should mention, that we did not do any further manual annotation of the tweets collected for each user since these tweets are only used to infer and calculate the profile and the psycho-linguistic characteristics of the users.

7.4.3.3 Analysis of Conspiracy Propagators

Here we focus on answering: Which are the psycho-linguistic characteristics of users that are more likely to share posts that support/refute conspiracy theories? by analysing and comparing various psycho-linguistic patterns extracted from the tweets of the users. In particular, the *psycho-linguistic characteristics* include:

- *Personality traits*: the inferred personality traits of the user.
- *Sentiment*: the sentiment polarity expressed in the user's tweets (i.e., positive, negative).

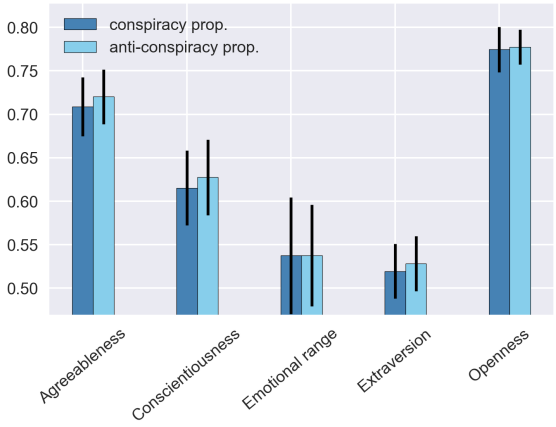
- *Emotions*: the amount of emotions expressed by the user in the tweets.
- *Linguistic patterns*: the amount of different linguistic patterns expressed in the tweets.

Personality traits: For the personality traits, we use the IBM Personality Insights API[27] to analyse users from multiple aspects based on their tweets. Figure 7.13 shows the personality traits regarding different aspects between conspiracy and anti-conspiracy propagators. In particular, we show the personality traits regarding the *Big Five*, *Values* and *Needs*. The Big-Five model [63] is one of the most well-studied and well-known models and identifies five dimensions of personality traits of people, agreeableness, conscientiousness, emotional range (also known as neuroticism), extroversion and openness. *Values* refers to motivating factors that influence the user's decision-making process and includes conservation, hedonism, openness to change, self-enhancement and self-transcendence. *Needs* includes 12 categories (i.e., challenge, closeness, curiosity, excitement, harmony, ideal, liberty, love, practicality, self-expression, stability and structure).

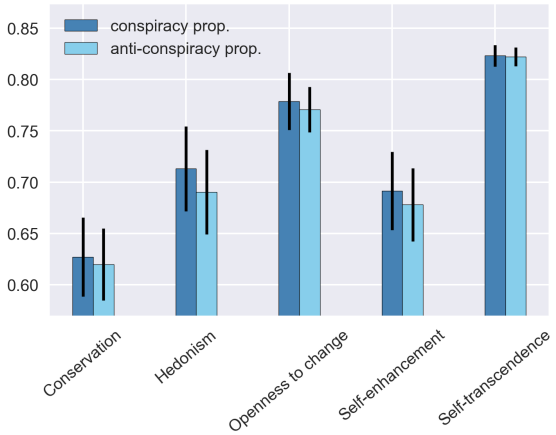
We observe that users that share posts that refute conspiracy theories have a higher score in *agreeableness* ($\mu = 0.72$) compared to the conspiracy propagators ($\mu = 0.708$, $p < 0.001$) as well as in *conscientiousness* ($p < 0.001$) and *extroversion* ($p < 0.001$). On the other hand, regarding *Values*, conspiracy propagators have higher scores in *conservation* ($\mu = 0.627$, $p < 0.001$), *hedonism* ($\mu = 0.713$, $p < 0.001$), *openness to change* ($\mu = 0.779$, $p < 0.001$), and *self-enhancement* ($\mu = 0.692$, $p < 0.001$) compared to anti-conspiracy propagators. With regard to *Needs*, conspiracy propagators have higher scores in *excitement* ($\mu = 0.637$, $p < 0.001$), *harmony* ($\mu = 0.802$, $p < 0.001$), *ideal* ($\mu = 0.682$, $p < 0.001$) and *liberty* ($\mu = 0.716$, $p < 0.001$) compared to anti-conspiracy propagators.

Emotions: We extract the emotions expressed in the tweets of a user. We follow Plutchik's model [167] and focus on the following eight emotions: anger, anticipation, disgust, fear, joy, sadness, surprise, and trust. To extract the emotions, we use NRC emotions lexicon [148] that contains around 14K words labeled with regards to these emotions. We calculate the scores for the eight different emotions. We observe that the prevalent emotion expressed in the posts of both, conspiracy propagator and anti-conspiracy propagator, is trust. In general, anti-conspiracy propagators express more emotions in their tweets compared to conspiracy propagators for all the emotions. Studies showed that emotions are very important for tasks such as author profiling [182], credibility detection [89] and the detection of false information in general [86]. However, a key difference in our study is that we analyse the emotions that are expressed by the users in their posts and not only in the posts that support/refute a conspiracy theory.

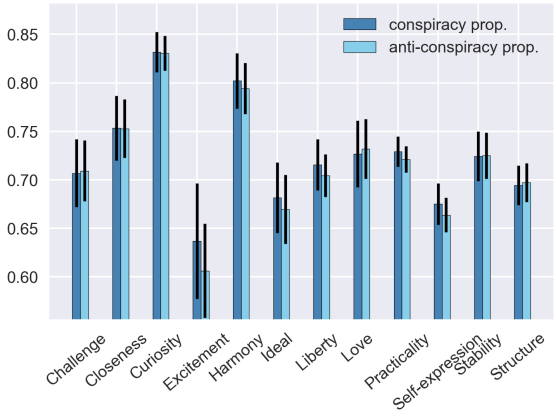
[27] https://personality-insights-demo.ng.bluemix.net/



(a) Big Five



(b) Values



(c) Needs

Figure 7.13: Personality related characteristics for conspiracy and anti-conspiracy propagators.

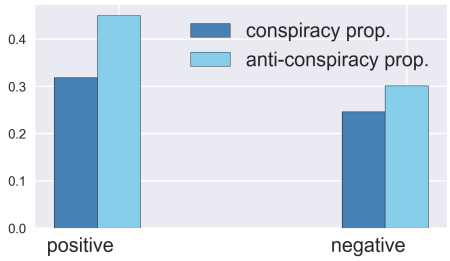


Figure 7.14: Average sentiment scores for conspiracy and anti-conspiracy propagators.

Sentiment: To extract the sentiment, we use NRC emotions lexicon [148] that additionally to emotions provided also annotated for sentiment. Figure 7.14 shows that the tweets of users that tend to refute conspiracies express a larger amount of sentiment compared to the conspiracy propagators' tweets. This difference is smaller for the negative sentiment compared to the positive. In particular, anti-conspiracy propagators show a high usage regarding positive sentiment with an average score of $\mu = 0.45$ in comparison to conspiracy propagators ($\mu = 0.319$, $p < 0.001$) as well as regarding negative sentiment ($p < 0.001$).

Linguistic patterns: For the linguistic patterns we employed LIWC [162], a standard approach for mapping text to 73 psychologically-meaningful categories, for comparing the psychological characteristics between conspiracy and anti-conspiracy propagators. In particular, we extract pronouns (I, we, you, she/he, they), personal concerns (work, leisure, home, money, religion, death), time focus (past, present, future), informal language (swear, assent, nonfluencies, fillers), cognitive processes (causation, discrepancy, tentative, certainty), affective processes (anxiety). For the linguistic analysis, we apply the following process. Given one tweet of a user, first we count how many words of the LIWC category appear in that tweet. We do this for all the tweets of the user, and then we calculate the average score for that user by dividing the total count with the number of tweets. Finally, we calculate the average of the scores for the conspiracy and anti-conspiracy propagators list.

Figure 7.15 shows the scores of the different psycho-linguistic characteristics between the conspiracy and anti-conspiracy propagators. Figure 7.15(a) shows that the users that refute conspiracy theories exhibit a higher usage of the third singular (i.e., she/he) and the first plural person (i.e., we) in comparison to the users that tend to support the conspiracy theories. In Figure 7.15(b) we can see that users who share posts refuting conspiracy theories exhibit higher usage of personal concerns in comparison to users that tend to support conspiracies. In particular, anti-conspiracy propagators show a high usage regarding *work* (e.g., work, class, boss) with an aver-

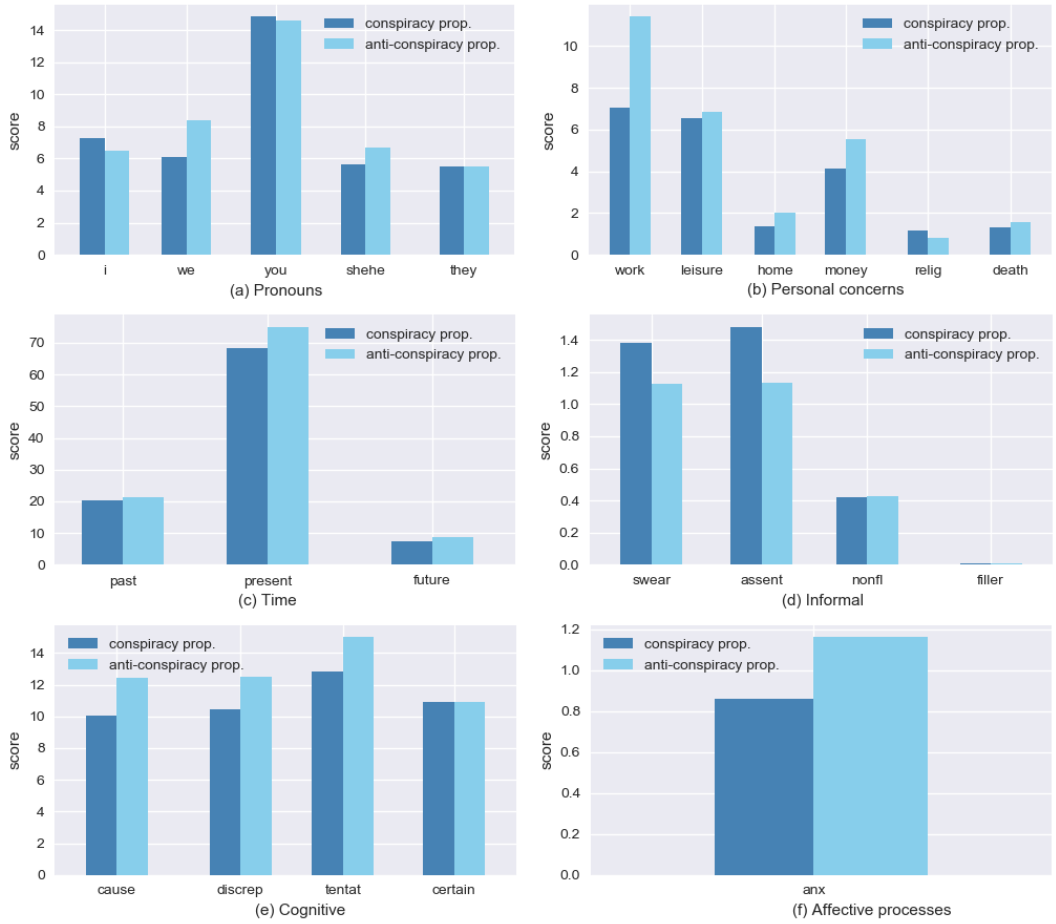


Figure 7.15: LIWC categories for conspiracy and anti-conspiracy propagators.

age score of μ= 11.411 in comparison to conspiracy propagators (μ= 7.058, $p < 0.001$). An example of a tweet that refers to work is *In the office today after being gone for a week. Greeting me was a pile of cards, letters and flowers.* In addition, anti-conspiracy propagators exhibit a statistical significant higher usage regarding *leisure* (e.g., house, TV, music), *money* (audit, cash, owe), *home* (e.g., house, kitchen, lawn) and *death*. On the other hand, we observe that the conspiracy propagators have more concerns regarding *religion* (μ= 1.175, $p < 0.001$). For example, the tweet *RT @TIME: Pope Francis opens the door for future female deacons* was posted by a conspiracy propagator. With regard to time focus, in Figure 7.15(c) we observe that both conspiracy and anti-conspiracy propagators focus more on *present* than *past* or *future*. This can be explained by the type of medium that is used to post what is happening at a specific time (i.e., tweeting). Also, users who tend to support conspiracies focus less on *present* (μ= 68.161, $p < 0.001$) and *future* (μ= 7.382, $p < 0.001$) in comparison to those that tend to refute the conspiracies. From Figure 7.15(d) we observe that the conspiracy propagators tend to use more *swear* words (μ= 1.381) compared to anti-conspiracy propagators (μ= 1.126, $p < 0.001$). Also, conspiracy propagators tend to use more *assent* words (e.g., agree, yup, okey) (μ= 1.481) compared to anti-conspiracy propagators (μ= 1.131, $p < 0.001$). Finally, regarding the cognitive processes, Figure 7.15(e) shows that anti-conspiracy propagators exhibit a higher usage in *causation* (because, effect, hence) in comparison to conspiracy propagators (μ= 10.079, $p < 0.001$). That is explained by the fact that users that refute the conspiracy theories use more explanations and arguments in their posts. Similarly, anti-conspiracy propagators show a statistical significant higher usage regarding *discrepancy* (should, would, could) and *tentative* (e.g., maybe, perhaps).

7.4.3.4 Experiments and Results

We use 25% of the users from our corpus for validation, 15% for test and the rest for the training. We initialize our embedding layer with the 300-dimensional pre-trained GloVe embeddings [164]. In addition, we evaluate the performance of Majority class, Random classifier, CNN with only embeddings (CNN), Universal Sentence Encoder (USE) [37], and ConspiDetector with profile features (e.g. verified, registration time, number of statuses, likes, followers, etc.) instead of the psycho-linguistic features. We also evaluated the performance of the BERT model [59]. However, we decided not to present the results of BERT because it achieved a very low performance that can be explained from the fact that BERT has been trained on contextual sentences whereas in our experiments the different tweets of a user are semantically unrelated. Also, when we concatenate all the tweets of a user, the final document length becomes very large[28], where BERT input

[28]The average length of the final documents in our collection is larger than 4000 tokens.

Table 7.16: Performance of the different combinations on the conspiracy and anti-conspiracy propagators detection.

	Precision	Recall	F1
Majority Class	0.51	1.00	0.34
Random	0.50	0.47	0.50
USE	0.70	0.69	0.69
CNN	0.68	0.82	0.68
CNN + Profile	0.61	0.78	0.58
CNN + Personality	0.75	0.79	0.73
CNN + LIWC	0.73	0.78	0.71
CNN + Sentiment	0.67	0.76	0.66
CNN + Emotion	0.77	0.58	0.67
ConspiDetector (Psycho-linguistic)	0.77	0.76	**0.74**
CNN + Psycho-linguistic + Profile	0.72	0.70	0.68

length is limited to 512 tokens. For the USE baseline, we represent the final concatenated documents using USE embeddings[29] and then we feed them to a Logistic Regression classifier, which achieved the highest performance among the other tested classifiers (Random Forest, Support Vector Machine, and Naïve Bayes). Also, we evaluate the performance of CNN using word embeddings and one group of features each time. For the evaluation, we report precision, recall and macro averaged F1 score.

Table 7.16 shows the results of our experiments. We observe that ConspiDetector (CNN + psycho-linguistic) achieves the best performance. In particular, ConspiDetector manages to improve the performance by 8.82% compared to the CNN baseline. Regarding using individual groups of features, the most effective is the IBM personality traits with a performance of 0.73 with regard to F1. The lowest performance is achieved with the profile characteristics (CNN + Profile) that is lower than the CNN baseline. Also, we observe that sentiment and emotion are not helpful and similar to the profile characteristics, they obtain a lower performance compared to the CNN baseline. This is an interesting observation since sentiment and emotion have been shown to be important in the detection of false information [89, 86]. However, in the previous studies, the emotions were extracted from the false claims, whereas in our study we analyse the emotional and sentimental language used by the users and therefore we use all the available published tweets of the users. Finally, we observe that the result of the USE shows comparable performance to the CNN.

[29] https://tfhub.dev/google/universal-sentence-encoder-large/3

7.4.4 PAN 2020: Profiling Fake News Spreaders

Although the detection of fake news, and credibility in general, has received a lot of research attention [88], there are only few studies that have addressed the problem from a user or author profiling perspective. For example, Shu et al. [203] analyzed different features, such as registration time, and found that users that share fake news have more recent accounts than users who share real news. Vo and Lee [225] analyzed the linguistic characteristics (e.g., use of tenses, number of pronouns) of fact-checking tweets and proposed a deep learning framework to generate responses with fact-checking intention. In our work in Section 7.4.1, we employed a model based on a Convolutional Neural Network that combines word embeddings with features that represent users' personality traits and linguistic patterns, to discriminate between fake news spreaders and fact-checkers.

Recently, we started addressing the problem of fake news detection from the author profiling perspective, with the aim of profiling those users that have shared some fake news in the past. The objective is to identify possible fake news spreaders on Twitter as a first step towards preventing fake news from being propagated among online users. This should help for their early detection and, therefore, for preventing their further dissemination. Thus, we organised a shared task on *Profiling fake news spreaders on Twitter* at the PAN Lab [20]. Following, we present an overview of the baselines that we used, together with their results, and the collected dataset.

7.4.4.1 Models

As baselines to compare the performance of the participants with, we have selected: (1) an LSTM that uses fastText[30] embeddings to represent texts; (2) a Neural Network (NN) with word n-grams (size 1-3) and (3) a Support Vector Machine (SVM) with char n-grams (size 2-6); (4) an SVM with Low Dimensionality Statistical Embeddings (LDSE) [185] to represent texts; (5) the Emotionally-Infused Neural (EIN) network[31] described in Chapter 5, and (6) a Random prediction.

7.4.4.2 Dataset Collection

We built a dataset of fake and real news spreaders, i.e. discriminating authors that have shared some fake news in the past from those that, to the best of our knowledge, have never done it. Table 7.17 presents the statistics of our dataset that consists of 500 authors for each of the two languages, English and Spanish. For each author we retrieved via the Twitter API her last 100

[30] https://fasttext.cc/docs/en/crawl-vectors.html

[31] For Spanish, we use the Spanish Emotion Lexicon [204] to extract emotions from tweets.

Table 7.17: Statistics of the PAN-AP-20 dataset for the shared task on profiling fake news spreaders on Twitter.

Language	Training	Test	Total
English	300	200	500
Spanish	300	200	500

Tweets. The dataset for each language is balanced, with 250 authors for each class (fake and real news spreaders).

7.4.4.3 Experiments and Results

We represent each author in the dataset by concatenating her tweets into one document and then we feed this document to the above models. In Table 7.18 we present the results. Whereas for the task of false information detection the EIN model obtained better results than those of the baselines (see Chapter 5), for profiling fake news spreaders its performance, although better than the one of a simple LSTM without emotional features, was not better than those obtained by classical machine learning classifiers based on n-grams. Somehow these results are in line with those obtained in previous author profiling shared tasks where classical approaches based on n-grams features obtained the best results [184]. The lower performance of EIN and LSTM is likely due to the small size of the training data that does not allow deep neural models to generalise. EIN aims at taking into account affective information but the way that spreaders write tweets that do not contain fake news does not seem to be much different than not spreaders. Therefore, spreaders seem to employ more emotion trigger words only in case of tweets containing fake news. At the moment, we are evaluating the performance of the models of the participants. A complete description of the shared task will be available in [181].

7.4.5 Ethical Concerns

Our previous work have some ethical concerns. First, we should mention that the aim of our previously mentioned systems that can differentiate between potential legitimate and suspicious users should be used by no means to stigmatize the users that have shared in the past fake news. On the contrary, such tools should be used only for the benefit of the users. For example, they could be used as a supportive tool to prevent the propagation of fake news, but not to judge users. Also, another use would be to raise the awareness of users. Moreover, there are also some ethical concerns regarding the collection and the release of the data; we plan to make those collections available only for research purpose. To protect the privacy of users, we anonymized the data (e.g., user names). Also, in accordance with the EU General Data

Table 7.18: Results on the PAN-AP-20 dataset.

Lang.	Method	$F1_{Fake\_news}$	Accuracy
English	Random	0.51	0.51
	NN + word ngrams	0.73	0.69
	SVM + char ngrams	0.70	0.68
	LDSE	**0.74**	**0.75**
	LSTM	0.55	0.56
	EIN	0.67	0.64
Spanish	Random	0.50	0.50
	NN + word ngrams	0.76	0.70
	SVM + char ngrams	**0.78**	**0.79**
	LDSE	0.77	**0.79**
	LSTM	0.60	0.60
	EIN	0.64	0.64

Protection Regulation [183], we used neutral annotation labels regarding the two classes (i.e., 0 and 1 instead of, for instance, checker and spreader) since we do not want to stigmatize specific users.

7.5 Conclusions

In this chapter we tried to connect the contents of the main parts of this thesis which have been published as research articles. We include new experiments in order to present a complete picture to the reader.

Regarding the first part of this thesis, we presented further experiments for the task of fact-checking false claims. Here we presented another system for the task but in a monolingual setup. We showed that extracted evidences from search engines are helpful to fact-check a given claim. In addition, since online claims may not be factual in some cases (do not contain facts to be verified), we conducted further experiments for identifying claims that are worthy of checking. This step can improve the efficiency of fact-checking systems by discarding unworthy claims in a real-life scenario. In this work, we proposed an approach based on a text distortion technique that was inspired by a work on authorship attribution [210]. The overall results showed that our model is very competitive comparing to other systems. Finally, we proposed another system for fact-checking news by including stance information. We used a set of lexicons that represent stance categories (e.g. Belief, Denial, Knowledge, Negation, etc.) to extract our feature set, and we combined them with other word-based and semantic-based features. The system showed promising results comparing to state-of-the-art models.

As one of the main objectives of this thesis is to understand the role of emotions in false information, in this part we presented further experiments

on understanding how emotions vary across the text of a fake news article. In our work in Chapter 5, we did not consider the position of affective information in a false information text, in other words, having the joy emotion at the beginning of an article's text was considered totally equal of having the same emotion at the end of the text. In our experiments in this part, we proposed a model for using the chronological order of affective features for the detection of false information. The results demonstrated the effectiveness of our proposed model, and the analysis presented valuable insights on how authors of false information take advantage of false stories to manipulate readers' options.

As false information spreaders are one of the main reasons for the massive spread of false content in social media, in Part III we studied those spreaders mainly from stylistic and semantic perspectives in social media. In this part we extended our experiments by proposing different models that include psychological, topical, linguistic, etc. features. Moreover, in this part we compared false information spreaders to fact checkers, in addition to our comparison to real news spreaders. The obtained results showed that our proposed models clearly improved the detection of false information spreaders considering several datasets. We presented a comprehensive analysis of the used features to understand their role in the detection process. Last but not least, we presented an overview on our shared task on Profiling Fake News Spreaders on Twitter, and we showed initial experiments on the used dataset.

Chapter 8

Conclusions and Future Work

8.1 Contributions

The rise of online Websites and social media platforms has contributed to the process of information sharing. The rapid and vast sharing of online information has helped the appearance of information disorder. This issue refers to the situation when we have information environments that are polluted. Polluted information does not always mean that the information is false, but it might be authentic information used in threatening, offensive, or misleading ways. Figure 8.1 shows some examples of these false information. To this end, the work in this thesis has focused on several types of polluted information.

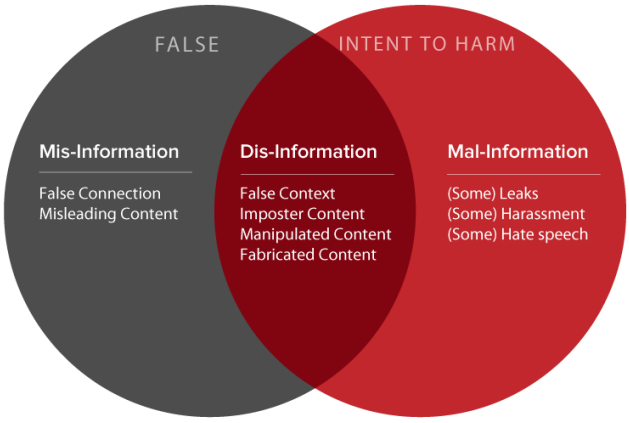


Figure 8.1: Examples of information disorder [234].

Our research works in the previous parts of this thesis addressed false information from several perspectives. In Part I we investigated the importance of verifying factuality in the information we may receive. In fact, information cannot be factual because of false claims or because of irony

was used, for instance in satirical news. Both tasks have been studied previously but a monolingual setting. In both work, we proposed cross-lingual approaches and we compared them to several baselines to validate their effectiveness. In Part II, we focused on investigating the roles of emotions in false information. Precisely, we studied the potential of employing emotions features in a rumor verification system. Moreover, we investigated the role of emotions in several types of false news and we proposed an emotionally-infused neural model to detect those types. Next, in Part III we studied the language of false information spreaders using several features. We proposed a model that incorporates psycho-linguistic features extracted from the profiles of those users that could be considered fake news spreaders because, looking at their Twitter timeline, they propagated false information. We also provided a comprehensive analysis using the social media parameters, personality traits, etc., in order to show the differences between false information spreaders and genuine users. Finally, in Part IV we further analysed the results that were obtained in the parts mentioned above, and we described further experiments we have done.

Our research work studied false information from several perspectives. The results we obtained allow us to answer the research questions that we introduced in the introduction of this thesis:

- **RQ1** *Can we detect false information from a cross-lingual perspective?* Our experiments in Part I demonstrated that cross-lingual approaches are capable of detecting and verifying false information. In Chapter 2, we proposed an approach that uses cross-lingual word embeddings to detect English, French, and Arabic ironic not factual messages and we compared its performance to a monolingual one. The results showed that despite the language and cultural differences between the different languages, the obtained results were competitive. In Chapter 3, the results of our cross-lingual fact-checking system proved that false information can be verified from a cross-lingual perspective. The performance of our proposed systems achieved the best result when compared to monolingual systems in the CheckThat! fact-checking shared task.

- **RQ2** *Can affective information improve the detection of false information?* For answering this research question, we investigated the role of emotions, and affective information in general, in two different works. In Chapter 4, the results of our rumour verification and stance detection system showed that affective information like emotions, sentiment, etc., improved the performance of our proposed method. Similarly, in Chapter 5, we proposed an emotionally-infused model that employs emotions in a deep neural network and we compared it to several baselines on two different datasets, one from social media messages and another from online news articles. The results demonstrated

that emotions effectively improved the performance of our model. This conclusion was evident when we carried out an ablation test excluding emotions from our architecture. Furthermore, in Section 7.3 we carried out additional experiments to investigate if affective information has the same chronological distribution in fake news with respect to texts containing truthful information. We found that fake news articles exaggerate while presenting false events in the first part of their text by using negative emotions, e.g., fear, sadness, etc. On the other hand, after taking the attention of the reader and manipulating her emotions with their false information, they ended the story by using a lower amount of negative emotions.

- **RQ3** *Can we detect spreaders of false information from a textual perspective?* In Chapter 6 we proposed a neural network model that combines several text-based features like stylistic, emotions, etc., with word embeddings, to classify social media accounts as either false information spreaders or not. The proposed system utilizes the sequential order of the accounts' tweets to detect the changes in the used features. We compared the performance of our proposed system to other robust baselines to validate it and the results showed that we are able effectively to identify false information spreaders. Moreover, we conducted an ablation test for a better understanding of the impact of the used features. In Section 7.4, we present further experiments to profile such users in social media. Also, we provide an overview on the shared task we organized at the PAN Lab at CLEF on profiling fake news spreader in social media.

To sum up, we believe that the answers to the aforementioned questions show the potentials of limiting false information in different ways, either from a cross-lingual or considering affective information. Besides, the potentials of profiling false information spreaders should limit the propagation of false information in social media.

8.2 Future Work

Recently, due to conflicts, disasters, and health issues that our world is facing, the effect of false information in social media became more critical than ever. Moreover, false information can contain also fake images and videos and needs to be addressed from a multimodal perspective. Although there are previous work that address the problem of the generation of fake content in media [220][1], very few works have been proposed by the research community [250, 32] to detect multimodal fake content [90]. This emphasizes

[1] https://en.wikipedia.org/wiki/Deepfake

the importance of investigating multimodal fake news detection as one of future directions [91]. Last but not least, we would like to focus on improving the explainability aspect of our proposed models. Recent research works on false information highlighted the importance of having explainable false information detectors and not only an accurate performance [222].

The research work in this thesis has focused on detecting false information and profiling its spreaders. We studied false information from several aspects, although the core of our research was on investigating the importance of taking into account affective information. Moreover, regarding the fake news spreaders, we studied the language they employed, and we used mainly textual feature to profile them. Following, we identify further research directions we intend to explore to extend the research carried out in the framework of our Ph.D. In Part I, we proposed two approaches for verifying if the information was factual from a cross-lingual perspective. Both methods use word embedding vectors aligned in an embedding vector space to unify similar terms across several languages. However, these cross-lingual embeddings perform worse than the recently proposed multilingual BERT, XLM [129], or other multilingual pretrained models. Thus, in order to produce more coherent systems, as future work, we are interested in investigating the performance of those models in our tasks.

In Part II we focused on the role of affective information in detecting false information, e.g., emotions. In our proposed approaches, we took into account the affective information from the text using affective lexicons. Nonetheless, it would be interesting to investigate more recent methods for extracting affective information from the text. The lexicons we used have two main disadvantages; firstly, they are monolingual, and this prevents applying our approaches to other languages. Secondly, they are limited in terms of the size since they have been created either by manual annotation or in a semi-supervised way. Although the latter should overtake the limitation in size by including more words, it usually includes also more noise. As a consequence, it makes any analysis based on its output less accurate.

Finally, in Part III, we proposed an approach for detecting false news spreaders on Twitter using several lexical, stylistic, psycholinguistic, and semantic features. For future work, we will keep working on investigating further possible features to detect false information spreaders. In our analysis about the language used by fake news spreaders and suspicious online users (e.g., trolls), we found that most of those users use slang, bad and sexual words in their tweets to offend some of their victims (malinformation). For that, we included in our proposed approach in Section 7.4.2 as features the existence of the previous word categories. We observed also that those users frequently use figurative language in their messages (e.g., sarcasm) to mock their victims. Thus, modeling the existence of figurative language should help detect them.

8.3 Research Publications

In this section we list our publications during the Ph.D. by grouping them into two main research categories: 1) Misinformation and Disinformation, and 2) Malinformation:

8.3.1 Misinformation and Disinformation

A) **Rumors and Stance**

1) **Ghanem B.**, Rosso P., Rangel F. (2018). Stance Detection in Fake News: a Combined Feature Representation. In: First Workshop on Fact Extraction and Verification (FEVER 2018), co-located with EMNLP, Brussels, Belgium, November 1st, pp. 66-71.
2) **Ghanem B.**, Cignarella A. T., Bosco C., Rosso P., Pardo, F. M. R. (2019). 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 (SemEval-2019), co-located with the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019), Minneapolis, Minnesota, USA, June 6-7, pp. 1125-1131.

B) **False Information and Fact-Checking**

3) **Ghanem B.**, Montes-y-Gómez M., Rangel F., Rosso P. (2018). UPV-INAOE-Autoritas - Check That: An Approach based on External Sources to Detect Claims Credibility. In: Cappellato L., Ferro N., Nie J.L., Soulier L. (Eds.) CLEF 2018 Labs and Workshops, Notebook Papers, Avignon, France, September 10th, CEUR Workshop Proceedings, CEUR-WS.org, vol. 2125.
4) **Ghanem B.**, Montes-y-Gómez M., Rangel F., Rosso P. (2018). UPV-INAOE-Autoritas - Check That: Preliminary Approach for Checking Worthiness of Claims. In: Cappellato L., Ferro N., Nie J.L., Soulier L. (Eds.) CLEF 2018 Labs and Workshops, Notebook Papers, Avignon, France, September 10th, CEUR Workshop Proceedings, CEUR-WS.org, vol. 2125.
5) **Ghanem B.**, Glavas G., Giachanou A., Ponzetto S., Rosso P., Rangel F. (2019). UPV-UMA at CheckThat! Lab: Verifying Arabic Claims using Crosslingual Approach. In: L. Cappellato,

N. Ferro, D. E. Losada and H. Müller (eds.) CLEF 2019 Labs and Workshops, Notebook Papers, Lugano, Switzerland, September 9th, CEUR Workshop Proceedings, CEUR-WS.org, vol. 2380.

C) **False Information and Emotions**

6) **Ghanem B.**, Rosso P., Rangel, F. (2020). An Emotional Analysis of False Information in Social Media and News Articles. ACM Transactions on Internet Technology (TOIT), 20(2), pp. 1-18. (Impact Factor: 2.382, Q2)

7) Rosso P., **Ghanem B.**, Giachanou A. (2020). On the Impact of Emotions on the Detection of False Information. In European Conference on Natural Language Processing and Information Retrieval.

D) **False Information and Figurative Language**

8) **Ghanem B.**, Rangel F., Rosso P. (2018). LDR at SemEval-2018 Task 3: A Low Dimensional Text Representation for Irony Detection. In: Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval 2018), Co-located with NAACL, New Orleans, Louisiana, June 5-6. Association for Computational Linguistics, pp. 531–536.

9) **Ghanem B.**, Karoui J., Benamara F., Moriceau V., Rosso P. (2019). IDAT@FIRE2019: Overview of the Track on Irony Detection in Arabic Tweets. In: Notebook Papers of FIRE 2019, FIRE-2019, Kolkata, India, December 12-15, CEUR Workshop Proceedings. CEUR-WS.org, vol. 2517, pp. 380-390.

10) **Ghanem B.**, Karoui J., Benamara F., Rosso P., and Moriceau V. (2020). Irony Detection in a Multilingual Context. In Advances in Information Retrieval. ECIR-2020. Lecture Notes in Computer Science, vol. 12036, pp. 141-149. Springer, Cham. (Core A Conference)

E) **False Information Spreaders**

11) Rangel F., Rosso P., Charfi A., Zaghouani W., **Ghanem B.**, Sánchez-Junquera J. (2019). Overview of the Track on Author Profiling and Deception Detection in Arabic. In: Notebook Papers of FIRE 2019, FIRE-2019, Kolkata, India, December 12-15, CEUR Workshop Proceedings. CEUR-WS.org, vol. 2517, pp. 70-83.

12) Giachanou A., **Ghanem B.** (2019). Bot and Gender Detection using Textual and Stylistic Information. In: L. Cappellato, N. Ferro, D. E. Losada and H. Müller (eds.) CLEF 2019 Labs and Workshops, Notebook Papers, Lugano, Switzerland, September 9th, CEUR Workshop Proceedings, CEUR-WS.org, vol. 2380.

13) Giachanou A., Ríssola E. A., **Ghanem B.**, Crestani F., Rosso P. (2020). The Role of Personality and Linguistic Patterns in Discriminating between Fake News Spreaders and Fact Checkers. In: Proceedings of the 23rd International Conference on Applications of Natural Language to Information Systems, NLDB-2020, Springer, LNCS (12089), pp. 181–192. (Core C Conference; and best paper award)

14) Bevendorff J., **Ghanem B.**, Giachanou A., Kestemont M., Manjavacas E., Potthast M., Rangel F., Rosso P., Specht G., Stamatatos E., Stein B., Wiegmann M., Zangerle E. (2020). Shared Tasks on Authorship Analysis at PAN 2020. In Advances in Information Retrieval. ECIR-2020. Lecture Notes in Computer Science, vol. 12036, pp. 508-516. Springer, Cham. (Core A Conference)

15) Bevendorff J., **Ghanem B.**, Giachanou A., Kestemont M., Manjavacas E., Potthast M., Rangel F., Rosso P., Specht G., Stamatatos E., Stein B., Wiegmann M., Zangerle E. (2020). Overview of PAN 2020: Authorship Verification, Celebrity Profiling, Profiling Fake News Spreaders on Twitter, and Style Change Detection. In: Experimental IR Meets Multilinguality, Multimodality, and Visualization, Proc. 11th Int. Conf. of the CLEF Association, CLEF 2020, September 22–25, Springer, LNCS (12260).

16) **Ghanem, B.** and Ponzetto, S. P. and Rosso, P. (2020). FacTweet: Profiling Fake News Twitter Accounts. (eds) Statistical Language and Speech Processing (SLSP). Lecture Notes in Computer Science. Springer, Cham.

17) Rangel F., Giachanou A., **Ghanem B.**, Rosso P. (2020). Overview of the 8th Author Profiling Task at PAN 2020: Pro-filing Fake News Spreaders on Twitter. In: Linda Cappellato, Carsten Eickhoff, Nicola Ferro, and Aurélie Névéol, CLEF 2020 Labs and Work-shops, Notebook Papers, September 22-25. CEUR-WS.org.

18) **Ghanem, B.** and Buscaldi, Davide and Rosso, P. (2020). TexTrolls: Identifying Trolls on Twitter with Textual and Affective Features. In: Workshop on Online Misinformation- and Harm-Aware Recommender Systems, co-located with RecSys, Rio de Janeiro, Brazil, September 25th.

8.3.2 Malinformation

19) Frenda S., **Ghanem B.**, Guzmán-Falcón E., Montes-y-Gómez M., Villasenor-Pineda L. (2018). Automatic Lexicons Expansion for Multilingual Misogyny Detection. In: Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018), co-located with the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Turin, Italy, December 12-13, CEUR Proceedings, vol. 2263.

20) Frenda S., **Ghanem B.**, Montes-y-Gómez M. (2018). Exploration of Misogyny in Spanish and English Tweets. In: Proceedings of the Third Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018), co-located with 34th Conf. of the Spanish Society for Natural Language Processing (SEPLN 2018), Seville, Spain, September 18th, CEUR Proc., vol. 2150, pp. 260-267.

21) Frenda S., **Ghanem B.**, Montes-y-Gómez M., Rosso P. (2019). Online Hate Speech against Women: Automatic Identification of Misogyny and Sexism on Twitter. In: Journal of Intelligent & Fuzzy Systems, vol. 36, num. 5, pp. 4743–4752. (Impact Factor: 1.637, Q3)

Bibliography

[1] Ahmet Aker, Leon Derczynski, and Kalina Bontcheva, (2017), Simple Open Stance Classification for Rumour Analysis. *arXiv preprint arXiv:1708.05286*.

[2] Ahmet Aker, Vincentius Kevin, and Kalina Bontcheva, (2019), Credibility and Transparency of News Sources: Data Collection and Feature Analysis.

[3] Ahmet Aker, Vincentius Kevin, and Kalina Bontcheva, (2019), Predicting News Source Credibility.

[4] Magda B Arnold, (1960), *Emotion and Personality*. Columbia University Press.

[5] Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho, (2017), Unsupervised Neural Machine Translation. *arXiv preprint*.

[6] Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein, (2020), Generating fact checking explanations. *arXiv preprint arXiv:2004.05773*.

[7] Salvatore Attardo, (2000), Irony as Relevant Inappropriateness. *Journal of Pragmatics*, 32(6):793–826.

[8] Adam Badawy, Kristina Lerman, and Emilio Ferrara, (2019), Who Falls for Online Political Manipulation? In: *Companion Proceedings of The 2019 World Wide Web Conference*. pp. 162–168. ACM.

[9] Hareesh Bahuleyan and Olga Vechtomova, (2017), 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)*. pp. 461–464.

[10] Hareesh Bahuleyan and Olga Vechtomova, (2017), 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)*. pp. 461–464.

[11] Sean Baird, Doug Sibley, and Yuxi Pan. *Talos Targets Disinformation with Fake News Challenge Victory.* http://blog.talosintelligence.com/2017/06/talos-fake-news-challenge.

[12] Alexandra Balahur and Marco Turchi, (2014), Comparative Experiments Using Supervised Learning and Machine Translation for Multilingual Sentiment Analysis. *Comput. Speech Lang.*, 28(1):56–75.

[13] Ramy Baly, Georgi Karadzhov, Dimitar Alexandrov, James Glass, and Preslav Nakov, (2018), Predicting Factuality of Reporting and Bias of News Media Sources. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pp. 3528–3539.

[14] Ramy Baly, Georgi Karadzhov, Abdelrhman Saleh, James Glass, and Preslav Nakov, (2019), Multi-Task Ordinal Regression for Jointly Predicting the Trustworthiness and the Leading Political Ideology of News Media. 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)*. pp. 2109–2116.

[15] Jeremy Barnes, Roman Klinger, and Sabine Schulte im Walde, (2018), Bilingual Sentiment Embeddings: Joint Projection of Sentiment Across Languages. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. pp. 2483–2493. Association for Computational Linguistics.

[16] Alberto Barrón-Cedeno, Giovanni Da San Martino, Israa Jaradat, and Preslav Nakov, (2019), Proppy: A System to Unmask Propaganda in Online News. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33. pp. 9847–9848.

[17] Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Rangel, Paolo Rosso, and Manuela Sanguinetti, (2019), Semeval-2019 task 5: Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter. In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. pp. 54–63.

[18] Elisa Bassignana, Valerio Basile, and Viviana Patti, (2018), Hurtlex: A Multilingual Lexicon of Words to Hurt. In: *5th Italian Conference on Computational Linguistics, CLiC-it 2018*, volume 2253. pp. 1–6. CEUR-WS.

[19] Farah Benamara, Cyril Grouin, Jihen Karoui, Véronique Moriceau, and Isabelle Robba, (2017), Analyse d'opinion et langage figuratif

dans des tweets présentation et résultats du Défi Fouille de Textes DEFT2017. In: *Actes de DEFT@TALN2017*. Orléans, France.

[20] Janek Bevendorff, Bilal Ghanem, Anastasia Giachanou, Mike Kestemont, Enrique Manjavacas, Martin Potthast, Francisco Rangel, Paolo Rosso, Günther Specht, Efstathios Stamatatos, Benno Stein, Matti Wiegmann, and Eva Zangerle, (2020), Shared Tasks on Authorship Analysis at PAN 2020. In: Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins, *Advances in Information Retrieval*. pp. 508–516. Cham.

[21] Mudit Bhargava, Pulkit Mehndiratta, and Krishna Asawa, (2013), Stylometric Analysis for Authorship Attribution on Twitter. In: *International Conference on Big Data Analytics*. pp. 37–47. Springer.

[22] Gaurav Bhatt, Aman Sharma, Shivam Sharma, Ankush Nagpal, Balasubramanian Raman, and Ankush Mittal, (2018), Combining Neural, Statistical and External Features for Fake News Stance Identification. In: *Companion of the The Web Conference 2018 on The Web Conference 2018*. pp. 1353–1357. International World Wide Web Conferences Steering Committee.

[23] Daniel Bikel and Imed Zitouni, (2012), *Multilingual Natural Language Processing Applications: From Theory to Practice*. IBM Press, 1st edition.

[24] Prakhar Biyani, Kostas Tsioutsiouliklis, and John Blackmer, (2016), "8 Amazing Secrets for Getting More Clicks": Detecting Clickbaits in News Streams Using Article Informality. In: *Thirtieth AAAI Conference on Artificial Intelligence*.

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

[26] Brandon C Boatwright, Darren L Linvill, and Patrick L Warren, (2018), Troll Factories: The Internet Research Agency and State-sponsored Agenda Building. *Resource Centre on Media Freedom in Europe*.

[27] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov, (2017), Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

[28] Cristina Bosco, Dell'Orletta Felice, Fabio Poletto, Manuela Sanguinetti, and Tesconi Maurizio, (2018), Overview of the Evalita 2018

Hate Speech Detection Task. In: *EVALITA 2018-Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian*, volume 2263. pp. 1–9. CEUR.

[29] Ryan L Boyd, Alexander Spangher, Adam Fourney, Besmira Nushi, Gireeja Ranade, James Pennebaker, and Eric Horvitz, (2018), Characterizing the Internet Research Agency's Social Media Operations During the 2016 US Presidential Election using Linguistic Analyses.

[30] Erin E Buckels, Paul D Trapnell, and Delroy L Paulhus, (2014), Trolls Just Want to have Fun. *Personality and individual Differences*, 67:97–102.

[31] Erik Cambria, Daniel Olsher, and Dheeraj Rajagopal, (2014), SenticNet 3: a Common and Common-sense Knowledge Base for Cognition-driven Sentiment Analysis. In: *Proceedings of the 28th AAAI Conference on Artificial Intelligence*.

[32] Juan Cao, Peng Qi, Qiang Sheng, Tianyun Yang, Junbo Guo, and Jintao Li, (2020), Exploring the role of visual content in fake news detection. *arXiv preprint arXiv:2003.05096*.

[33] Paula Carvalho, Luís Sarmento, Mário J Silva, and Eugénio De Oliveira, (2009), Clues for Detecting Irony in User-Generated Contents: Oh...!! It's "so easy";-). In: *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*. pp. 53–56. ACM.

[34] Sonia Castelo, Thais Almeida, Anas Elghafari, Aécio Santos, Kien Pham, Eduardo Nakamura, and Juliana Freire, (2019), A Topic-Agnostic Approach for Identifying Fake News Pages. In: *Companion Proceedings of The 2019 World Wide Web Conference*. pp. 975–980.

[35] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete, (2011), Information Credibility on Twitter. In: *Proceedings of the 20th international conference on World Wide Web*. pp. 675–684.

[36] Irish Internet Safety Awareness Centre. Explained: What is False Information (Fake News)? https://www.webwise.ie/teachers/what-is-fake-news/. [Online; accessed 03-March-2020].

[37] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil, (2018), Universal Sentence Encoder. *CoRR*, abs/1803.11175.

[38] Raymond Chakhachiro, (2007), Translating Irony in Political Commentary Texts from English into Arabic. *Babel*, 53(3):216–240.

[39] Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly, (2016), Stop Clickbait: Detecting and Preventing Clickbaits in Online News Media. In: *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. pp. 9–16.

[40] Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, Hui Jiang, and Diana Inkpen, (2016), Enhanced LSTM for Natural Language Inference. *arXiv preprint arXiv:1609.06038*.

[41] Yimin Chen, Niall J Conroy, and Victoria L Rubin, (2015), Misleading Online Content: Recognizing Clickbait as "False News". In: *Proceedings of the 2015 ACM on workshop on multimodal deception detection*. pp. 15–19.

[42] Lesley Chiou and Catherine Tucker. Fake News and Advertising on Social Media: A Study of the Anti-vaccination Movement. Technical report, National Bureau of Economic Research.

[43] Yoonjung Choi and Janyce Wiebe, (2014), +/-effectwordnet: Sense-level Lexicon Acquisition for Opinion Inference. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pp. 1181–1191.

[44] Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini, (2015), Computational Fact Checking from Knowledge Networks. *PloS one*, 10(6).

[45] Andrea Cimino and Felice Dell'Orletta, (2017), Stacked Sentence-Document Classifier Approach for Improving Native Language Identification. In: *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*. pp. 430–437.

[46] Eric M Clark, Jake Ryland Williams, Chris A Jones, Richard A Galbraith, Christopher M Danforth, and Peter Sheridan Dodds, (2016), Sifting Robotic from Organic Text: a Natural Language Approach for Detecting Automation on Twitter. *Journal of Computational Science*, 16:1–7.

[47] Herbert L Colston. Irony as Indirectness Cross-Linguistically: On the Scope of Generic Mechanisms. In: *Indirect Reports and Pragmatics in the World Languages*, pp. 109–131. Springer.

[48] Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou, (2017), Word Translation Without Parallel Data. *arXiv preprint*.

[49] Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R Bowman, Holger Schwenk, and Veselin Stoyanov, (2018), Xnli: Evaluating Cross-lingual Sentence Representations. *arXiv preprint arXiv:1809.05053*.

[50] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi, (2017), The Paradigm-shift of Social Spambots: Evidence, Theories, and Tools for the Arms Race. In: *Proceedings of the 26th international conference on world wide web companion*. pp. 963–972.

[51] Giovanni Da San Martino, Alberto Barrón-Cedeño, Preslav Nakov, Seunghak Yu, and Israa Jaradat. Propaganda Analysis Project. https://propaganda.qcri.org/. [Online; accessed 03-March-2020].

[52] Giovanni Da San Martino, Alberto Barron-Cedeno, and Preslav Nakov, (2019), 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*. pp. 162–170.

[53] Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeno, Rostislav Petrov, and Preslav Nakov, (2019), Fine-Grained Analysis of Propaganda in News Article. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. pp. 5640–5650.

[54] Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer, (2016), Botornot: A System to Evaluate Social Bots. In: *Proceedings of the 25th International Conference Companion on World Wide Web*. pp. 273–274. International World Wide Web Conferences Steering Committee.

[55] Jorge Carrillo De Albornoz, Laura Plaza, and Pablo Gervás, (2012), SentiSense: An Easily Scalable Concept-based Affective Lexicon for Sentiment Analysis. In: *LREC*. pp. 3562–3567.

[56] Marco L Della Vedova, Eugenio Tacchini, Stefano Moret, Gabriele Ballarin, Massimo DiPierro, and Luca de Alfaro, (2018), Automatic Online Fake News Detection Combining Content and Social Signals. In: *2018 22nd Conference of Open Innovations Association (FRUCT)*. pp. 272–279. IEEE.

[57] Hossein Derakhshan and Claire Wardle, (2017), Information Disorder: Definitions. *AA. VV., Understanding and Addressing the Disinformation Ecosystem*, pp. 5–12.

[58] Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga, (2017), SemEval-2017 Task 8: RumourEval: Determining Rumour Veracity and Support for Rumours. *arXiv preprint arXiv:1704.05972*.

[59] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, (2018), Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*.

[60] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, (2019), BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 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)*. pp. 4171–4186.

[61] Bhuwan Dhingra, Zhong Zhou, Dylan Fitzpatrick, Michael Muehl, and William W Cohen, (2016), Tweet2Vec: Character-Based Distributed Representations for Social Media. In: *The 54th Annual Meeting of the Association for Computational Linguistics*. p. 269.

[62] John P Dickerson, Vadim Kagan, and VS Subrahmanian, (2014), Using Sentiment to Detect Bots on Twitter: Are Humans more Opinionated than Bots? In: *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*. pp. 620–627.

[63] J M Digman, (1990), Personality Structure: Emergence of the Five-Factor Model. *Annual Review of Psychology*, 41(1):417–440.

[64] Sebastian Dungs, Ahmet Aker, Norbert Fuhr, and Kalina Bontcheva, (2018), Can Rumour Stance Alone Predict Veracity? In: *Proceedings of the 27th International Conference on Computational Linguistics*. pp. 3360–3370.

[65] Paul Ekman, (1992), An Argument for Basic Emotions. *Cognition & emotion*, 6(3-4):169–200.

[66] Tamer Elsayed, Preslav Nakov, Alberto Barrón-Cedeño, Maram Hasanain, Reem Suwaileh, Giovanni Da San Martino, and Pepa Atanasova, (2019), Overview of the CLEF-2019 CheckThat! Lab: Automatic Identification and Verification of Claims. In: *International Conference of the Cross-Language Evaluation Forum for European Languages*. pp. 301–321. Springer.

[67] EMC. The Digital Universe of Opportunities: Rich Data and the Increasing Value of the Internet of

Things. https://www.emc.com/leadership/digital-universe/2014iview/executive-summary.htm. [Online; accessed 03-March-2020].

[68] Andrea Esuli and Fabrizio Sebastiani, (2007), SentiWordNet: a High-coverage Lexical Resource for Opinion Mining. *Evaluation*, 17:1–26.

[69] Facebook. Facebook Safety Check. https://www.facebook.com/about/safetycheck. [Online; accessed 10-March-2020].

[70] Delia Irazú Hernańdez Farías, Viviana Patti, and Paolo Rosso, (2016), Irony Detection in Twitter: The Role of Affective Content. *ACM Transactions on Internet Technology (TOIT)*, 16(3):1–24.

[71] Ethan Fast, Binbin Chen, and Michael S. Bernstein, (2016), Empath: Understanding Topic Signals in Large-scale Text. In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. pp. 4647–4657. ACM.

[72] Simona Frenda, Bilal Ghanem, Estefanıa Guzmán-Falcón, Manuel Montes-y Gómez, Luis Villasenor-Pineda, and Optica y Electrónica, (2018), Automatic expansion of lexicons for multilingual misogyny detection. In: *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018) co-located with the Fifth Italian Conference on Computational Linguistics*, volume 2263. pp. 188–193. CEUR.

[73] Simona Frenda, Bilal Ghanem, and Manuel Montes-y Gómez, (2018), Exploration of Misogyny in Spanish and English Tweets. In: *Third Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018)*, volume 2150. pp. 260–267. CEUR Workshop Proceedings.

[74] Simona Frenda, Bilal Ghanem, Manuel Montes-y Gómez, and Paolo Rosso, (2019), Online Hate Speech against Women: Automatic Identification of Misogyny and Sexism on Twitter. *Journal of Intelligent & Fuzzy Systems*, 36(5):4743–4752.

[75] Jie Gao, Sooji Han, Xingyi Song, and Fabio Ciravegna, (2020), RP-DNN: A Tweet Level Propagation Context Based Deep Neural Networks for Early Rumor Detection in Social Media. *arXiv preprint arXiv:2002.12683*.

[76] Bilal Ghanem, Labib Arafeh, Paolo Rosso, and Fernando Sánchez-Vega, (2018), HYPLAG: Hybrid Arabic Text Plagiarism Detection System. In: *International conference on applications of natural language to information systems*. pp. 315–323. Springer.

[77] Bilal Ghanem, Davide Buscaldi, and Paolo Rosso, (2020), TexTrolls: Identifying Trolls on Twitter from a Textual Perspective. In: *Proceedings of the Workshop on Online Misinformation- and Harm-Aware Recommender Systems.*

[78] Bilal Ghanem, Alessandra Teresa Cignarella, Cristina Bosco, Paolo Rosso, and Francisco Rangel, (2019), 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.* pp. 1125–1131.

[79] Bilal Ghanem, Goran Glavas, Anastasia Giachanou, Simone Paolo Ponzetto, Paolo Rosso, and Francisco Rangel, (2019), UPV-UMA at checkthat! lab: Verifying arabic claims using a cross lingual approach. In: *Working Notes of CLEF 2019 - Conference and Labs of the Evaluation Forum, Lugano, Switzerland, September 9-12.*

[80] Bilal Ghanem, Jihen Karoui, Farah Benamara, Véronique Moriceau, and Paolo Rosso, (2019), IDAT at FIRE2019: Overview of the Track on Irony Detection in Arabic Tweets. In: *Proceedings of the 11th Forum for Information Retrieval Evaluation.* pp. 10–13. ACM.

[81] Bilal Ghanem, Jihen Karoui, Farah Benamara, Véronique Moriceau, and Paolo Rosso, (2019), IDAT@FIRE2019: Overview of the Track on Irony Detection in Arabic Tweets. In: *Working Notes of the Forum for Information Retrieval Evaluation (FIRE 2019). CEUR Workshop Proceedings. In: CEUR-WS.org, Kolkata, India*, volume 2517. pp. 380–390.

[82] Bilal Ghanem, Manuel Montes-y Gòmez, Francisco Rangel, and Paolo Rosso, (2018), UPV-INAOE-Autoritas - Check That: An Approach based on External Sources to Detect Claims Credibility. In: *In Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, CLEF '18*, volume 2125, Avignon, France.

[83] Bilal Ghanem, Manuel Montes-y Gómez, Francisco Rangel, and Paolo Rosso, (2018), UPV-INAOE-Check That: Preliminary Approach for Checking Worthiness of Claims. In: *Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum.*, volume 2125.

[84] Bilal Ghanem, Francisco Rangel, and Paolo Rosso, (2018), LDR at SemEval-2018 Task 3: A Low Dimensional Text Representation for Irony Detection. In: *Proceedings of The 12th International Workshop on Semantic Evaluation.* pp. 531–536.

[85] Bilal Ghanem, Paolo Rosso, and Francisco Rangel, (2018), Stance Detection in Fake News A Combined Feature Representation. In: *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*. pp. 66–71.

[86] Bilal Ghanem, Paolo Rosso, and Francisco Rangel, (2020), An Emotional Analysis of False Information in Social Media and News Articles. *ACM Transactions on Internet Technology (TOIT)*, 20(2):1–18.

[87] Anastasia Giachanou, Esteban A Ríssola, Bilal Ghanem, Fabio Crestani, and Paolo Rosso, (2020), The role of personality and linguistic patterns in discriminating between fake news spreaders and fact checkers. In: *International Conference on Applications of Natural Language to Information Systems*. pp. 181–192. Springer.

[88] Anastasia Giachanou, Paolo Rosso, and Fabio Crestani, (2019), Leveraging Emotional Signals for Credibility Detection. In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. pp. 877–880.

[89] Anastasia Giachanou, Paolo Rosso, and Fabio Crestani, (2019), Leveraging Emotional Signals for Credibility Detection. In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. (SIGIR '19 2019), pp. 877–880.

[90] Anastasia Giachanou, Guobiao Zhang, and Paolo Rosso, (2020), Multimodal Fake News Detection with Textual, Visual and Semantic Information). In: *23rd International Conference on Text, Speech and Dialogue*, September 8-11.

[91] Anastasia Giachanou, Guobiao Zhang, and Paolo Rosso, (2020), Multimodal Multi-image Fake News Detection. In: *7th IEEE International Conference on Data Science and Advanced Analytics, Special Session on Fake News, Bots, and Trolls*, October 6-9.

[92] Andrea Gianti, Cristina Bosco, Viviana Patti, Andrea Bolioli, and Luigi Di Caro, (2012), Annotating Irony in a Novel Italian Corpus for Sentiment Analysis. In: *Proceedings of the 4th Workshop on Corpora for Research on Emotion Sentiment and Social Signals, Istanbul, Turkey*. pp. 1–7.

[93] Goran Glavas, Robert Litschko, Sebastian Ruder, and Ivan Vulic, (2019), How to (Properly) Evaluate Cross-Lingual Word Embeddings: On Strong Baselines, Comparative Analyses, and Some Misconceptions. *arXiv preprint arXiv:1902.00508*.

[94] Genevieve Gorrell, Mehmet Emin Bakir, Ian Roberts, Mark Anthony Greenwood, Benedetta Iavarone, and Kalina Bontcheva, (2019), Partisanship, Propaganda and Post-Truth Politics: Quantifying Impact in Online Debate. *The Journal of Web Science*, 7.

[95] Genevieve Gorrell, Elena Kochkina, Maria Liakata, Ahmet Aker, Arkaitz Zubiaga, Kalina Bontcheva, and Leon Derczynski, (2019), SemEval-2019 task 7: RumourEval, Determining Rumour Veracity and Support for Rumours. In: *Proceedings of the 13th International Workshop on Semantic Evaluation*. pp. 845–854.

[96] Cyril Goutte and Serge Léger, (2017), Exploring Optimal Voting in Native Language Identification. In: *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*. pp. 367–373.

[97] Jesse Graham, Jonathan Haidt, and Brian A Nosek, (2009), Liberals and Conservatives Rely on Different Sets of Moral Foundations. *Journal of personality and social psychology*, 96(5):1029–1046.

[98] Ana Granados, Manuel Cebrian, David Camacho, and Francisco de Borja Rodriguez, (2010), Reducing the Loss of Information through Annealing Text Distortion. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):1090–1102.

[99] Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov, (2018), Learning Word Vectors for 157 Languages. *CoRR*, abs/1802.06893.

[100] H. Paul Grice. Logic and Conversation. In: Peter Cole and Jerry L. Morgan, *Speech Acts. Syntax and Semantics, Volume 3*, pp. 41–58. Academic Press, New York.

[101] Andreas Hanselowski, PVS Avinesh, Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M Meyer, and Iryna Gurevych, (2018), A Retrospective Analysis of the Fake News Challenge Stance-Detection Task. In: *Proceedings of the 27th International Conference on Computational Linguistics*. pp. 1859–1874.

[102] Andreas Hanselowski, Avinesh PVS, Benjamin Schiller, and Felix Caspelherr. *Team Athene on the Fake News Challenge*. https://medium.com/@andre134679/team-athene-on-the-fake-news-/challenge-28a5cf5e017b.

[103] Andreas Hanselowski, Avinesh PVS, Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M Meyer, and Iryna Gurevych, (2018), A Retrospective Analysis of the Fake News Challenge Stance Detection Task. *arXiv preprint arXiv:1806.05180*.

[104] Maram Hasanain, Reem Suwaileh, Tamer Elsayed, Alberto Barrón-Cedeño, and Preslav Nakov, Overview of the CLEF-2019 CheckThat! Lab on Automatic Identification and Verification of Claims. Task 2: Evidence and Factuality. In: *In Working Notes of CLEF 2019 - Conference and Labs of the Evaluation Forum, CLEF '19*, volume 2380.

[105] Will Haskell. People Explaining their 'Personal Paradise' is the Latest Hashtag to Explode on Twitter. https://www.businessinsider.com.au/hashtag-games-on-twitter-2015-6. [Online; accessed 15-March-2020].

[106] Cynthia Van Hee, Els Lefever, and Véronique Hoste, (2018), SemEval-2018 Task 3: Irony Detection in English Tweets. In: *Proceedings of The 12th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT, New Orleans, Louisiana, June 5-6, 2018*. pp. 39–50.

[107] Delia Irazú Hernández Farías, Cristina Bosco, Viviana Patti, and Paolo Rosso, (2017), Sentiment Polarity Classification of Figurative Language: Exploring the Role of Irony-Aware and Multifaceted Affect Features. In: *International Conference on Computational Linguistics and Intelligent Text Processing*. pp. 46–57. Springer.

[108] Sepp Hochreiter and Jürgen Schmidhuber, (1997), Long Short-Term Memory. *Neural computation*, 9(8):1735–1780.

[109] Joan B Hooper, (1974), *On Assertive Predicates*, volume 4. In J. Kimball, editor, Syntax and Semantics.

[110] Benjamin D Horne and Sibel Adali, (2017), This Just In: Fake News Packs a Lot in Title, Uses Simpler, Repetitive Content in Text Body, More Similar to Satire than Real News. In: *Eleventh International AAAI Conference on Web and Social Media*.

[111] Hen-Hsen Huang, Chiao-Chen Chen, and Hsin-Hsi Chen, (2018), Disambiguating False-Alarm Hashtag Usages in Tweets for Irony Detection. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. pp. 771–777.

[112] Yu-Hsiang Huang, Hen-Hsen Huang, and Hsin-Hsi Chen, (2017), Irony Detection with Attentive Recurrent Neural Networks. In: *European Conference on Information Retrieval*. pp. 534–540. Springer.

[113] Jane Im, Eshwar Chandrasekharan, Jackson Sargent, Paige Lighthammer, Taylor Denby, Ankit Bhargava, Libby Hemphill, David Jurgens, and Eric Gilbert, (2019), Still Out There: Modeling and Identifying Russian Troll Accounts on Twitter. *arXiv preprint arXiv:1901.11162*.

[114] Romania Insider. Measles outbreak claims another life in Romania. https://www.romania-insider.com/measles-outbreak-romania/. [Online; accessed 03-March-2020].

[115] Fredrik Johansson, (2019), Supervised Classification of Twitter Accounts Based on Textual Content of Tweets. In: *CLEF 2019 Labs and Workshops, Notebook Papers*, volume 2380. pp. 1–8. CEUR-WS.org.

[116] Oliver P. John and Sanjay Srivastava. The Big-five Trait Taxonomy: History, Measurement, and Theoretical Perspectives. In: *Handbook of personality: Theory and research*, pp. 102–138. Guilford Press, New York, USA.

[117] Georgi Karadzhov, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, and Ivan Koychev, (2017), Fully Automated Fact Checking Using External Sources. In: *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*. pp. 344–353.

[118] Alireza Karduni, Ryan Wesslen, Sashank Santhanam, Isaac Cho, Svitlana Volkova, Dustin Arendt, Samira Shaikh, and Wenwen Dou, (2018), Can you Verifi this? Studying Uncertainty and Decision-making about Misinformation using Visual Analytics. In: *Twelfth international AAAI conference on web and social media*.

[119] Jihen Karoui, Farah Benamara, and Véronique Moriceau, (2017), SOUKHRIA: Towards an Irony Detection System for Arabic in Social Media. In: *Third International Conference On Arabic Computational Linguistics, ACLING 2017, November 5-6, 2017, Dubai, United Arab Emirates*. pp. 161–168.

[120] Jihen Karoui, Farah Benamara, Véronique Moriceau, Nathalie Aussenac-Gilles, and Lamia Hadrich Belguith, (2015), Towards a Contextual Pragmatic Model to Detect Irony in Tweets. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing : Volume 2 short papers*. (ACL-IJCNLP'15 2015), pp. 644–650.

[121] Jihen Karoui, Farah Benamara, Véronique Moriceau, Viviana Patti, Cristina Bosco, and Nathalie Aussenac-Gilles, (2017), Exploring the Impact of Pragmatic Phenomena on Irony Detection in Tweets: A Multilingual Corpus Study. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. pp. 262–272. Association for Computational Linguistics.

[122] Jihen Karoui, Farah Banamara Zitoune, and Veronique Moriceau, (2017), Soukhria: Towards an Irony Detection System for Arabic in Social Media. *Procedia Computer Science*, 117:161–168.

[123] Lauri Karttunen, (1971), Implicative Verbs. *Language*, pp. 340–358.

[124] Yoon Kim, (2014), Convolutional Neural Networks for Sentence Classification. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pp. 1746–1751. Association for Computational Linguistics.

[125] Paul Kiparsky and Carol Kiparsky, (1968), *Fact*. Linguistics Club, Indiana University.

[126] Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga, (2018), All-in-one: Multi-task Learning for Rumour Verification. *arXiv preprint arXiv:1806.03713*.

[127] KP Krishna Kumar and G Geethakumari, (2014), Detecting Misinformation in Online Social Networks Using Cognitive Psychology. *Human-centric Computing and Information Sciences*, 4(1):14.

[128] Srijan Kumar, Robert West, and Jure Leskovec, (2016), Disinformation on the Web: Impact, Characteristics, and Detection of Wikipedia Hoaxes. In: *Proceedings of the 25th international conference on World Wide Web*. pp. 591–602.

[129] Guillaume Lample and Alexis Conneau, (2019), Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

[130] Kyumin Lee, Brian David Eoff, and James Caverlee, (2011), Seven Months with the Devils: A Long-term Study of Content Polluters on Twitter. In: *Fifth international AAAI conference on weblogs and social media*.

[131] Young-Suk Lee, Kishore Papineni, Salim Roukos, Ossama Emam, and Hany Hassan, (2003), Language Model Based Arabic Word Segmentation. In: *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. pp. 399–406. Association for Computational Linguistics.

[132] Xian Li, Weiyi Meng, and Clement Yu, (2011), T-verifier: Verifying Truthfulness of Fact Statements. In: *Data Engineering (ICDE), 2011 IEEE 27th International Conference on*. pp. 63–74. IEEE.

[133] Christine Liebrecht, Florian Kunneman, and Bosch Antal van den, (2013), The Perfect Solution for Detecting Sarcasm in Tweets# Not. In: *Proceedings of the 4th Workshop on Computational Approaches to*

Subjectivity, Sentiment and Social Media Analysis. pp. 29–37. New Brunswick, NJ: ACL.

[134] Robert Litschko, Goran Glavaš, Simone Paolo Ponzetto, and Ivan Vulić, (2018), Unsupervised cross-lingual information retrieval using monolingual data only. In: *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. (SIGIR '18 2018), pp. 1253–1256.

[135] Mario Livio, (2017), *Why?: What Makes Us Curious*. Simon and Schuster Publishing.

[136] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha, (2016), Detecting Rumors from Microblogs with Recurrent Neural Networks. In: *IJCAI*. pp. 3818–3824.

[137] Jing Ma, Wei Gao, and Kam-Fai Wong, (2018), Detect Rumor and Stance Jointly by Neural Multi-task Learning. In: *Companion Proceedings of the The Web Conference 2018*. pp. 585–593.

[138] Jing Ma, Wei Gao, and Kam-Fai Wong, (2018), Rumor Detection on Twitter with Tree-structured Recursive Neural Networks. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. pp. 1980–1989.

[139] Laurens van der Maaten and Geoffrey Hinton, (2008), Visualizing Data Using t-SNE. *Journal of machine learning research*, 9(11):2579–2605.

[140] Suraj Maharjan, Sudipta Kar, Manuel Montes-y Gómez, Fabio A González, and Thamar Solorio, (2018), Letting Emotions Flow: Success Prediction by Modeling the Flow of Emotions in Books. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. pp. 259–265.

[141] Shervin Malmasi, Keelan Evanini, Aoife Cahill, Joel Tetreault, Robert Pugh, Christopher Hamill, Diane Napolitano, and Yao Qian, (2017), A Report on the 2017 Native Language Identification Shared Task. In: *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*. pp. 62–75.

[142] Ilia Markov, Lingzhen Chen, Carlo Strapparava, and Grigori Sidorov, (2017), CIC-FBK Approach to Native Language Identification. In: *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*. pp. 374–381.

[143] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, (2013), Efficient Estimation of Word Representations in Vector Space. *CoRR*, abs/1301.3781.

[144] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig, (2013), Linguistic Regularities in Continuous Space Word Representations. In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. pp. 746–751.

[145] Clyde R Miller, (1939), The Techniques of Propaganda. From "How to Detect and Analyze Propaganda," an Address Given at Town Hall. *The Center for learning*.

[146] Saif M. Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry, (2016), SemEval-2016 Task 6: Detecting Stance in Tweets. In: *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. pp. 31–41.

[147] Saif M. Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko, (2017), Stance and Sentiment in Tweets. *ACM Transactions on Internet Technology*, 17(3):26:1–26:23.

[148] Saif M. Mohammad and Peter D. Turney, (2010), Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon. In: *Proceedings of the NAACL HLT 2010*. pp. 26–34. ACL.

[149] Douglas Colin Muecke, (1969), *The Compass of Irony*. Routledge.

[150] Subhabrata Mukherjee and Gerhard Weikum, (2015), Leveraging Joint Interactions for Credibility Analysis in News Communities. In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. pp. 353–362.

[151] Preslav Nakov, Alberto Barrón-Cedeno, Tamer Elsayed, Reem Suwaileh, Lluís Màrquez, Wajdi Zaghouani, Pepa Atanasova, Spas Kyuchukov, and Giovanni Da San Martino, (2018), Overview of the CLEF-2018 CheckThat! Lab on Automatic Identification and Verification of Political Claims. In: *Proceedings of the International Conference of the Cross-Language Evaluation Forum for European Languages*. pp. 372–387.

[152] Yair Neuman, (2016), *Computational Personality Analysis: Introduction, Practical Applications and Novel Directions*. Springer Publishing Company, Incorporated, 1st edition.

[153] Yair Neuman and Yochai Cohen, (2014), A Vectorial Semantics Approach to Personality Assessment. *Scientific Reports*, 4(1).

[154] Alfred Ng. This was the Most Viewed Facebook ad Bought by Russian Trolls. https://www.cnet.com/news/this-was-the-most-viewed-facebook-ad-bought-by-russian-trolls/. [Online; accessed 23-March-2020].

[155] Jian Ni and Radu Florian, (2017), Improving Multilingual Named Entity Recognition with Wikipedia Entity Type Mapping. *CoRR*, abs/1707.02459.

[156] Finn Årup Nielsen, (2011), A new ANEW: Evaluation of a Word List for Sentiment Analysis in Microblogs. *arXiv preprint arXiv:1103.2903*.

[157] Brendan Nyhan and Jason Reifler, (2010), When Corrections Fail: The Persistence of Political Misperceptions. *Political Behavior*, 32(2):303–330.

[158] Reynier Ortega-Bueno, Francisco Rangel, DI Hernández Farıas, Paolo Rosso, Manuel Montes-y Gómez, and José E Medina Pagola, (2019), Overview of the Task on Irony Detection in Spanish Variants. In: *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019), co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2019). CEUR-WS. org*, volume 2421. pp. 229–256.

[159] Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti, (2018), Stance Classification for Rumour Analysis in Twitter: Exploiting Affective Information and Conversation Structure. In: *Proceedings of the 2nd International Workshop on Rumours and Deception in Social Media (RDSM 2018)*, volume 2482. pp. 1–7.

[160] W Gerrod Parrott, (2001), *Emotions in Social Psychology: Essential Readings*. Psychology Press.

[161] Gabriella Pasi, Marco Viviani, and Alexandre Carton, (2019), A Multi-Criteria Decision Making approach based on the Choquet integral for assessing the credibility of User-Generated Content. *Information Sciences*, 503:574–588.

[162] James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. The Development and Psychometric Properties of LIWC 2015. Technical report.

[163] James W. Pennebaker, Martha E. Francis, and Roger J. Booth, (2001), Linguistic Inquiry and Word Count: LIWC 2001. *Mahway: Lawrence Erlbaum Associates*, 71.

[164] Jeffrey Pennington, Richard Socher, and Christopher Manning, (2014), Glove: Global Vectors for Word Representation. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. (EMNLP '14 2014), pp. 1532–1543.

[165] Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea, (2018), Automatic Detection of Fake News. In: *Proceedings of the 27th International Conference on Computational Linguistics*, Aug. pp. 3391–3401. Association for Computational Linguistics, Santa Fe, New Mexico, USA.

[166] Warren A Peterson and Noel P Gist, (1951), Rumor and Public Opinion. *American Journal of Sociology*, 57(2):159–167.

[167] R. Plutchik, (1982), A Psychoevolutionary Theory of Emotions. *Social Science Information*.

[168] Robert Plutchik, (2001), The Nature of Emotions: Human Emotions have Deep Evolutionary Roots, a Fact that may Explain their Complexity and Provide Tools for Clinical Practice. *American scientist*, 89(4):344–350.

[169] Lahari Poddar, Wynne Hsu, Mong Li Lee, and Shruti Subramaniyam, (2018), 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)*. pp. 65–72. IEEE.

[170] Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum, (2016), Credibility Assessment of Textual Claims on the Web. In: *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. pp. 2173–2178. ACM.

[171] Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum, (2017), 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*. pp. 1003–1012.

[172] Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum, (2018), DeClarE: Debunking Fake News and False Claims using Evidence-Aware Deep Learning. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. pp. 22–32.

[173] Soujanya Poria, Alexander Gelbukh, Amir Hussain, Newton Howard, Dipankar Das, and Sivaji Bandyopadhyay, (2013), Enhanced Sentic-

Net with Affective Labels for Concept-Based Opinion Mining. *IEEE Intelligent Systems*, 28(2):31–38.

[174] Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen, (2016), Clickbait Detection. In: *European Conference on Information Retrieval*. pp. 810–817. Springer.

[175] Martin Potthast, Francisco Rangel, Michael Tschuggnall, Efstathios Stamatatos, Paolo Rosso, and Benno Stein, (2017), Overview of PAN'17. In: *International Conference of the Cross-Language Evaluation Forum for European Languages*. pp. 275–290. Springer.

[176] Rob Procter, Farida Vis, and Alex Voss, (2013), Reading the Riots on Twitter: Methodological Innovation for the Analysis of Big Data. *International Journal of Social Research Methodology*, 16(3):197–214.

[177] Tomáš Ptáček, Ivan Habernal, and Jun Hong, (2014), Sarcasm Detection on Czech and English Twitter. In: *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. pp. 213–223.

[178] Kunal Purohit. Misinformation, Fake News Spark India Coronavirus Fears. https://www.aljazeera.com/news/2020/03/misinformation-fake-news-spark-india-coronavirus-fears-200309051731540.html. [Online; accessed 10-March-2020].

[179] Vahed Qazvinian, Emily Rosengren, Dragomir R Radev, and Qiaozhu Mei, (2011), Rumor has it: Identifying Misinformation in Microblogs. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pp. 1589–1599. Association for Computational Linguistics.

[180] Colin Raffel and Daniel PW Ellis, (2015), Feed-Forward Networks with Attention Can Solve Some Long-Term Memory Problems. *arXiv preprint arXiv:1512.08756*.

[181] Francisco Rangel, Anastasia Giachanou, Bilal Ghanem, and Paolo Rosso, (2020), Overview of the 8th Author Profiling Task at PAN 2020: Profiling Fake News Spreaders on Twitter. In: Linda Cappellato, Carsten Eickhoff, Nicola Ferro, and Aurélie Névéol, *CLEF 2020 Labs and Workshops, Notebook Papers*, sep. CEUR-WS.org.

[182] Francisco Rangel and Paolo Rosso, (2016), On the Impact of Emotions on Author Profiling. *Information processing & management*, 52(1):73–92.

[183] Francisco Rangel and Paolo Rosso, (2019), On the implications of the general data protection regulation on the organisation of evaluation tasks. *Language and Law= Linguagem e Direito*, 5(2):95–117.

[184] Francisco Rangel and Paolo Rosso, (2019), Overview of the 7th Author Profiling Task at PAN 2019: Bots and Gender Profiling. In: *CLEF 2019 labs and workshops, notebook papers. CEUR Workshop Proceedings. CEUR-WS.org.*, volume 2380.

[185] Francisco Rangel, Paolo Rosso, and Marc Franco-Salvador, (2018), A Low Dimensionality Representation for Language Variety Identification. In: *In 17th International Conference on Intelligent Text Processing and Computational Linguistics, CICLing'16. Springer, LNCS(9624).* pp. 156–169.

[186] Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi, (2017), 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.* pp. 2931–2937.

[187] Marta Recasens, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky, (2013), Linguistic Models for Analyzing and Detecting Biased Language. In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, volume 1. pp. 1650–1659.

[188] CITS Research. A Brief History of Fake News. https://www.cits.ucsb.edu/fake-news/brief-history. [Online; accessed 03-March-2020].

[189] Paul Resnick, Samuel Carton, Souneil Park, Yuncheng Shen, and Nicole Zeffer, (2014), Rumorlens: A System for Analyzing the Impact of Rumors and Corrections in Social Media. In: *Proceedings of the Computational Journalism Conference.*

[190] Benjamin Riedel, Isabelle Augenstein, Georgios P Spithourakis, and Sebastian Riedel, (2017), A Simple but Tough-to-beat Baseline for the Fake News Challenge Stance Detection Task. *arXiv preprint arXiv:1707.03264.*

[191] Paolo Rosso, Francisco Rangel, Bilal Ghanem, and Anis Charfi, (2018), ARAP: Arabic Author Profiling Project for Cyber-Security. *Sociedad Española para el Procesamiento del Lenguaje Natural.*

[192] Victoria L Rubin, Yimin Chen, and Niall J Conroy, (2015), Deception Detection for News: Three Types of Fakes. In: *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research*

in and for the Community. pp. 1–4. American Society for Information Science.

[193] Victoria L Rubin, Niall Conroy, Yimin Chen, and Sarah Cornwell, (2016), Fake News or Truth? Using Satirical Cues to Detect Potentially Misleading News. In: *Proceedings of the second workshop on computational approaches to deception detection*. pp. 7–17.

[194] Natali Ruchansky, Sungyong Seo, and Yan Liu, (2017), Csi: A Hybrid Deep Model for Fake News Detection. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. pp. 797–806. ACM.

[195] Sebastian Ruder, (2017), A survey of cross-lingual embedding models. *CoRR*, abs/1706.04902.

[196] Motaz Saad and Basem O Alijla, (2017), Wikidocsaligner: An off-the-shelf Wikipedia Documents Alignment Tool. In: *Proceedings of the 2017 Palestinian International Conference on Information and Communication Technology*. pp. 34–39.

[197] Hassan Saif, Yulan He, Miriam Fernandez, and Harith Alani, (2016), Contextual Semantics for Sentiment Analysis of Twitter. *Information Processing & Management*, 52(1):5–19.

[198] Shota Sasaki, Shuo Sun, Shigehiko Schamoni, Kevin Duh, and Kentaro Inui, (2018), Cross-Lingual Learning-to-Rank with Shared Representations. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. pp. 458–463.

[199] Chengcheng Shao, Giovanni Luca Ciampaglia, Onur Varol, Alessandro Flammini, and Filippo Menczer, (2017), The Spread of Fake News by Social Bots. *arXiv preprint arXiv:1707.07592*, pp. 96–104.

[200] Elisa Shearer and Katerina Eva Matsa. News Use Across Social Media Platforms 2018. https://www.journalism.org/2018/09/10/news-use-across-social-media-platforms-2018/. [Online; accessed 03-March-2020].

[201] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu, (2018), FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media. *arXiv preprint arXiv:1809.01286*.

[202] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu, (2017), Fake News Detection on Social Media: A Data Mining Perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36.

[203] Kai Shu, Suhang Wang, and Huan Liu, (2018), Understanding User Profiles on Social Media for Fake News Detection. In: *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. pp. 430–435. IEEE.

[204] G Sidorov, S Miranda-Jiménez, F Viveros-Jiménez, A Gelbukh, N Castro-Sánchez, F Velásquez, I Dıaz-Rangel, S Suárez-Guerra, A Trevino, and J Gordon, (2012), Empirical Study of Opinion Mining in Spanish Tweets. *Lect Notes Comput Sci*, 7629:1–14.

[205] A Sigar and Z Taha, (2012), A Contrastive Study of Ironic Expressions in English and Arabic. *College of Basic Education Researchers Journal*, 12(2):795–817.

[206] Ian Simpson. Man Pleads Guilty in Washington Pizzeria Shooting Over Fake News. https://www.reuters.com/article/us-washingtondc-gunman/man-pleads-guilty-in-washington-pizzeria-shooting-over-fake-news-idUSKBN16V1XC. [Online; accessed 10-may-2019].

[207] Stephen Skalicky, Nicholas Duran, and Scott A Crossley, (2020), Please, Please, Just Tell Me: The Linguistic Features of Humorous Deception.

[208] Samuel L. Smith, David H.P. Turban, Steven Hamblin, and Nils Y. Hammerla, (2017), Offline Bilingual Word Vectors, Orthogonal Transformations and the Inverted Softmax. In: *Proceedings of ICLR*.

[209] Abu Bakr Soliman, Kareem Eissa, and Samhaa R. El-Beltagy, (2017), AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP. In: *Third International Conference On Arabic Computational Linguistics, ACLING 2017, November 5-6, 2017, Dubai, United Arab Emirates*. pp. 256–265.

[210] Efstathios Stamatatos, (2017), Authorship Attribution using Text Distortion. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. pp. 1138–1149.

[211] Madeena Sultana, Padma Polash, and Marina Gavrilova, (2017), Authorship Recognition of Tweets: A Comparison Between Social Behavior and Linguistic Profiles. In: *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. pp. 471–476. IEEE.

[212] Sahil Swami, Ankush Khandelwal, Vinay Singh, Syed Sarfaraz Akhtar, and Manish Shrivastava, (2018), A Corpus of English-Hindi Code-Mixed Tweets for Sarcasm Detection. In: *19th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*.

[213] Eugenio Tacchini, Gabriele Ballarin, Marco L Della Vedova, Stefano Moret, and Luca de Alfaro, (2017), Some Like it Hoax: Automated Fake News Detection in Social Networks. In: *Proceedings of the Second Workshop on Data Science for Social Good*, volume 1960. pp. 1–15.

[214] Marcella Tambuscio, Giancarlo Ruffo, Alessandro Flammini, and Filippo Menczer, (2015), Fact-checking Effect on Viral Hoaxes: A Model of Misinformation Spread in Social Networks. In: *Proceedings of the 24th international conference on World Wide Web*. pp. 977–982.

[215] Yi-jie Tang and Hsin-Hsi Chen, (2014), Chinese Irony Corpus Construction and Ironic Structure Analysis. In: *COLING 2014, 25th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, August 23-29, 2014, Dublin, Ireland*. pp. 1269–1278.

[216] Mariona Taulé, Maria Antònia Martí, Francisco Rangel, Paolo Rosso, Cristina Bosco, and Viviana Patti, (2017), Overview of the Task on Stance and Gender Detection in Tweets on Catalan Independence. In: *Proceedings of the 2nd Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2017)*, volume 1881. pp. 157–177. CEUR-WS.org.

[217] Yla R Tausczik and James W Pennebaker, (2010), The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods. *Journal of language and social psychology*, 29(1):24–54.

[218] Yi Tay, Anh Tuan Luu, Siu Cheung Hui, and Jian Su, (2018), Reasoning with Sarcasm by Reading In-Between. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. pp. 1010–1020.

[219] Mike Thelwall, Kevan Buckley, Georgios Paltoglou, Di Cai, and Arvid Kappas, (2010), Sentiment Strength Detection in Short Informal Text. *Journal of the American Society for Information Science and Technology*, 61(12):2544–2558.

[220] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner, (2016), Face2face: Real-time Face Capture and Reenactment of RGB Videos. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 2387–2395.

[221] Lin Tian, Xiuzhen Zhang, Yan Wang, and Huan Liu, (2020), Early Detection of Rumours on Twitter via Stance Transfer Learning. In: *European Conference on Information Retrieval*. pp. 575—-588. Springer.

[222] Tian Tian, Yudong Liu, Xiaoyu Yang, Yuefei Lyu, Xi Zhang, and Binxing Fang, (2020), QSAN: A Quantum-probability based Signed Attention Network for Explainable False Information Detection. In: *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*.

[223] Onur Varol, Emilio Ferrara, Clayton A Davis, Filippo Menczer, and Alessandro Flammini, (2017), Online Human-bot Interactions: Detection, Estimation, and Characterization. In: *Eleventh international AAAI conference on web and social media*.

[224] Tony Veale, (2011), Creative Language Retrieval: A Robust Hybrid of Information Retrieval and Linguistic Creativity. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*. (HLT '11 2011), pp. 278–287.

[225] Nguyen Vo and Kyumin Lee, (2019), Learning from Fact-checkers: Analysis and Generation of Fact-checking Language. In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. pp. 335–344.

[226] Svitlana Volkova, Stephen Ranshous, and Lawrence Phillips, (2018), Predicting Foreign Language Usage from English-Only Social Media Posts. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 2. pp. 608–614.

[227] Svitlana Volkova, Kyle Shaffer, Jin Yea Jang, and Nathan Hodas, (2017), Separating Facts from Fiction: Linguistic Models to Classify Suspicious and Trusted News Posts on Twitter. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2. pp. 647–653.

[228] Soroush Vosoughi, Mostafa 'Neo' Mohsenvand, and Deb Roy, (2017), Rumor Gauge: Predicting the Veracity of Rumors on Twitter. *ACM transactions on knowledge discovery from data (TKDD)*, 11(4):1–36.

[229] Soroush Vosoughi, Deb Roy, and Sinan Aral, (2018), The Spread of True and False News Online. *Science*, 359(6380):1146–1151.

[230] Takashi Wada and Tomoharu Iwata, (2018), Unsupervised Cross-lingual Word Embedding by Multilingual Neural Language Models. *arXiv preprint*.

[231] Po-Ya Angela Wang, (2013), # Irony or # Sarcasm – A Quantitative and Qualitative Study Based on Twitter. In: *Proceedings of the 27th*

Pacific Asia Conference on Language, Information, and Computation (PACLIC 27). pp. 349–356.

[232] William Yang Wang, (2017), "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. pp. 422–426.

[233] Claire Wardle and Hossein Derakhshan, (2017), Information Disorder: Toward an Interdisciplinary Framework for Research and Policy Making. *Council of Europe report*, 27.

[234] Claire Wardle and Hossein Derakhshan. One year on, we are still not recognizing the complexity of information disorder online. https://firstdraftnews.org/latest/coe_infodisorder/. [Online; accessed 15-March-2020].

[235] Helena Webb, Pete Burnap, Rob Procter, Omer Rana, Bernd Carsten Stahl, et al., (2016), Digital Wildfires: Propagation, Verification, Regulation, and Responsible Innovation. *ACM Transactions on Information Systems (TOIS)*, 34(3):15.

[236] Adina Williams, Nikita Nangia, and Samuel R Bowman, (2017), A broad-coverage Challenge Corpus for Sentence Understanding Through Inference. *arXiv preprint arXiv:1704.05426*.

[237] Michael Wilson, (1988), MRC Psycholinguistic Database: Machine-usable dictionary, version 2.00. *Behavior research methods, instruments, & computers*, 20(1):6–10.

[238] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann, (2005), Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In: *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[239] Fan Yang, Arjun Mukherjee, and Eduard Dragut, (2017), Satirical News Detection and Analysis using Attention Mechanism and Linguistic Features. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. pp. 1979–1989.

[240] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy, (2016), Hierarchical Attention Networks for Document Classification. In: *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*. pp. 1480–1489.

[241] Savvas Zannettou, Tristan Caulfield, Emiliano De Cristofaro, Michael Sirivianos, Gianluca Stringhini, and Jeremy Blackburn, (2019), Disinformation Warfare: Understanding State-Sponsored Trolls on Twitter and their Influence on the Web. In: *Companion Proceedings of The 2019 World Wide Web Conference*. pp. 218–226. ACM.

[242] Savvas Zannettou, Michael Sirivianos, Jeremy Blackburn, and Nicolas Kourtellis, (2019), The Web of False Information: Rumors, Fake News, Hoaxes, Clickbait, and Various Other Shenanigans. *Journal of Data and Information Quality (JDIQ)*, 11(3):1–37.

[243] Qiao Zhang, Shuiyuan Zhang, Jian Dong, Jinhua Xiong, and Xueqi Cheng. Automatic Detection of Rumor on Social Network. In: *Natural Language Processing and Chinese Computing*, pp. 113–122. Springer.

[244] Yigeng Zhang, Fan Yang, Yifan Zhang, Eduard Dragut, and Arjun Mukherjee, (2020), Birds of a feather flock together: Satirical news detection via language model differentiation. *arXiv preprint arXiv:2007.02164*.

[245] Zhe Zhao, Paul Resnick, and Qiaozhu Mei, (2015), Enquiring Minds: Early Detection of Rumors in Social Media from Enquiry Posts. In: *Proceedings of the 24th International Conference on World Wide Web*. pp. 1395–1405. International World Wide Web Conferences Steering Committee.

[246] Vitalii Zhelezniak, Aleksandar Savkov, April Shen, Francesco Moramarco, Jack Flann, and Nils Y Hammerla, (2019), Don't Settle for Average, Go for the Max: Fuzzy Sets and Max-Pooled Word Vectors. *arXiv preprint arXiv:1904.13264*.

[247] Guineng Zheng, Subhabrata Mukherjee, Xin Luna Dong, and Feifei Li, (2018), Opentag: Open Attribute Value Extraction from Product Profiles. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. pp. 1049–1058.

[248] Rong Zheng, Jiexun Li, Hsinchun Chen, and Zan Huang, (2006), A Framework for Authorship Identification of Online Messages: Writing-Style Features and Classification Techniques. *Journal of the American society for information science and technology*, 57(3):378–393.

[249] Xinyi Zhou and Reza Zafarani, (2020), A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys (CSUR)*, 53(5):1–40.

[250] Dimitrina Zlatkova, Preslav Nakov, and Ivan Koychev, (2019), Fact-checking meets fauxtography: Verifying claims about images. In: *Pro-*

ceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 2099–2108.

[251] Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter, (2018), Detection and Resolution of Rumours in Social Media: A Survey. *ACM Computing Surveys (CSUR)*, 51(2):32.

[252] Arkaitz Zubiaga, Elena Kochkina, Maria Liakata, Rob Procter, Michal Lukasik, Kalina Bontcheva, Trevor Cohn, and Isabelle Augenstein, (2018), Discourse-aware Rumour Stance Classification in Social Media using Sequential Classifiers. *Information Processing & Management*, 54(2):273–290.

[253] Arkaitz Zubiaga, Maria Liakata, Rob Procter, Kalina Bontcheva, and Peter Tolmie, (2015), Towards Detecting Rumours in Social Media. In: *AAAI Workshop: AI for Cities*.