Using Knowledge Graph To Detect And Explain Misinformation Spread On The Web

Youri Peskine

EURECOM
Sophia Antipolis

SORBONNE UNIVERSITÉ

PHD THESIS

In Partial Fulfilment of the Requirements for the
Degree of Doctor of Philosophy from Sorbonne University
Specialization: Data Science

Using Knowledge Graph To Detect And Explain Misinformation Spread On The Web

Youri PESKINE

Defended on 28/03/2025 before a committee composed of:

Reviewer	**BONTCHEVA Kalina**, University of Sheffield, Sheffield, UK
Reviewer	**VILLATA Serena**, Université Côte d'Azur, CNRS, Inria, Sophia Antipolis, France
Examiner	**ALAM Mehwish**, Télécom Paris, Institut Polytechnique de Paris, France
Jury President	**DUGELAY Jean-Luc**, EURECOM, Sophia-Antipolis, France
Thesis Director	**PAPOTTI Paolo** , EURECOM, Sophia Antipolis, France
Thesis Co-Supervisor	**TRONCY Raphaël**, EURECOM, Sophia Antipolis, France

dedicated to my friends and family

Acknowledgements

This thesis would not have been possible without the guidance, support, and encouragement of many people. It is with the deepest gratitude that I acknowledge their contributions.

First and foremost, I want to thank my supervisors, Dr. Paolo Papotti and Dr. Raphaël Troncy for their guidance throughout the years. Their insights, expertise, patience and valuable feedback have inspired me to push beyond my perceived limits. I would also like to thank my thesis committee members, Dr. Kalina Bontcheva, Dr. Serena Villata, Dr. Mehwish Alam and Dr. Jean-Luc Dugelay for dedicating their time and expertise to evaluating this thesis.

I would also like to thank all of my colleagues at Eurecom and within the CIMPLE project. While pursuing a PhD can be intimidating as a lot of the time is spent alone, having daily support from you has made it much more enjoyable. I am grateful for all the "tea-time" discussions we had, providing invaluable stress relief and emotional support, building an ideal work environment.

Finding a healthy balance between work and hobbies is essential to keep growing both professionally and personally. For this reason, I would like to thank all the people at the Eurecom Music Club for being awesome. Playing music with you to unwind after a long day is invaluable, and I am grateful for the opportunity to nurture my passion for music, through concerts and jam sessions.

To my friends, thank you for your convivial support and long-term encouragement. This journey would have been significantly more challenging and far less enjoyable without our endless discussions and activities. Your willingness to listen, and your much-needed distractions are essential to my well-being.

Above all, I want to thank my family, who has always been there for me. Their unconditional love and support provided the foundation that has allowed me to thrive, in so many ways. Thank you for your patience and dedication, your belief in me has been my greatest strength.

Abstract

The proliferation of misinformation, disinformation, and fake news is a critical global challenge, impacting diverse domains such as politics (US elections, Brexit), health (COVID-19 "infodemic"), and environmental issues (climate change denial). The rapid dissemination of false information, particularly through social media platforms, outpaces the ability of traditional fact-checking methods to effectively counter it. This thesis addresses the pressing need for scalable, automated tools to assist mitigators and researchers in combating misinformation by leveraging advancements in Natural Language Processing (NLP) and knowledge representation.

The research is guided by three central questions: (1) How to model relationships between the diverse types of data used in fact-checking, (2) How to better understand textual documents using automatic approaches, and (3) How to extend the notion of textual similarity for fact-checking applications. To address these questions, the thesis makes several key contributions.

First, we introduce Cimple KG, a continuously updated knowledge graph that integrates misinformation-related data from various sources, including social media posts, news articles, and fact-checking reports. This knowledge graph not only structures and normalizes metadata but also establishes relationships between disparate data points, addressing the challenge of scattered and heterogeneous information sources.

Second, we propose novel automatic approaches to detect and analyze textual features in misinformation-related documents, such as emotion, sentiment, political leaning, conspiracy theories, persuasion techniques, and narrative tropes. These features, termed Factors, provide deeper insights into the mechanisms underlying the spread of misinformation and are integrated into Cimple KG. We leverage BERT-based models to detect these features in tweets and memes, creating new state-of-the-art results, enhancing the understanding of textual documents and enabling more effective detection and analysis of misinformation. Furthermore, we explore the capabilities of Large Language Models (LLMs) in zero-shot classification tasks, demonstrating how improved class definitions can enhance their performance in understanding textual content.

Third, we extend the concept of textual similarity by introducing novel similarity measures tailored for fact-checking applications. These measures evaluate documents based on their ability to fact-check other documents, their relatedness through entities and concepts, and their granularity (e.g., comparing claims with news articles). We provide annotated datasets

and retrieval methods to validate these approaches, offering tools that go beyond traditional semantic textual similarity. Our work also includes practical applications of these similarity measures, such as in the Community Notes program on X (formerly Twitter), showcasing their utility in real-world scenarios.
This work contributes to the broader field of NLP and misinformation research by providing scalable tools and methodologies that empower fact-checkers, researchers, and policymakers to better understand and combat the spread of false information in the digital age.

Abrégé

La prolifération des désinformations et des « fake news » est un défi mondial majeur, touchant divers domaines tels que la politique (les élections américaines, le Brexit), la santé (l'« infodémie » liée à la COVID-19) et les questions environnementales (la négation du changement climatique). La diffusion rapide de fausses informations, en particulier à travers les plateformes de réseaux sociaux, surpasse la capacité des méthodes traditionnelles de vérification des faits à y faire face de manière efficace. Cette thèse aborde le besoin d'outils automatisés pour aider les mitigateurs et les chercheurs à lutter contre la désinformation, en utilisant des progrès en Traitement du Langage Naturel (TLN) et en représentation des connaissances.

La recherche est guidée par trois questions centrales : (1) Comment modéliser les relations entre les différents types de données utilisées dans la vérification des faits, (2) Comment mieux comprendre les documents textuels à l'aide d'approches automatiques, et (3) Comment étendre la notion de similarité textuelle pour les applications de vérification des faits. Pour répondre à ces questions, la thèse propose plusieurs contributions majeures. Tout d'abord, nous présentons Cimple KG, un graphe de connaissances continuellement mis à jour qui intègre des données liées à la désinformation provenant de diverses sources, comme des publications sur les réseaux sociaux, des articles de presse et des rapports de vérification des faits. Ce graphe de connaissances ne se contente pas de structurer et normaliser les métadonnées, il établit également des relations entre des points de données disparates, abordant ainsi le défi des sources d'informations éparses et hétérogènes.

Ensuite, nous proposons de nouvelles approches automatiques pour détecter et analyser les caractéristiques textuelles dans les documents liés à la désinformation, telles que l'émotion, le sentiment, les tendances politiques, les théories du complot, les techniques de persuasion et les tropes narratifs. Ces caractéristiques, appelées Factors, fournissent une compréhension plus approfondie des mécanismes sous-jacents à la propagation de la désinformation et sont intégrées dans Cimple KG. Nous utilisons des modèles basés sur BERT pour détecter ces caractéristiques dans des tweets et des mèmes, créant ainsi de nouveaux résultats à la pointe de la technologie, améliorant la compréhension des documents textuels et permettant une détection et une analyse plus efficaces de la désinformation. De plus, nous explorons les capacités des Grands Modèles de Langage (*Large Language Models*, LLM) dans les tâches de classification sans apprentissage ("*zero-shot*"), démontrant comment de meilleures définitions de classes peuvent améliorer leur performance dans la compréhension du contenu textuel.

Troisièmement, nous étendons le concept de similarité textuelle en introduisant de nouvelles mesures de similarité adaptées aux applications de vérification des faits. Ces mesures évaluent les documents en fonction de leur capacité à vérifier d'autres documents, de leur relation à travers les entités et les concepts, et de leur granularité (par exemple, en comparant des affirmations avec des articles de presse). Nous fournissons des ensembles de données annotées et des méthodes de recherche pour valider ces approches, offrant des outils qui vont au-delà de la similarité textuelle sémantique traditionnelle. Notre travail inclut également des applications pratiques de ces mesures de similarité, telles que dans le programme Community Notes sur X (anciennement Twitter), démontrant leur utilité dans des scénarios réels.
Ce travail contribue au domaine plus large du TLN et de la recherche sur la désinformation en fournissant des outils et des méthodologies évolutifs qui permettent aux vérificateurs de faits, aux chercheurs et aux décideurs de mieux comprendre et lutter contre la propagation des fausses informations à l'ère numérique.

Contents

Acknowledgements i

Abstract iii

List of Figures xi

List of Tables xiii

1 Introduction 1

1.1 Motivation . . . 1

1.1.1 Fake News, Misinformation, Disinformation . . . 1

1.1.2 Fact Checking . . . 2

1.1.3 Natural Language Processing . . . 4

1.1.4 Challenges . . . 4

1.2 Research Questions . . . 6

1.3 Contributions . . . 6

1.4 Outline of the Thesis . . . 7

2 Related Work 11

2.1 Natural Language Processing . . . 11

2.1.1 Language Models . . . 11

2.2 Knowledge Graphs . . . 15

2.2.1 Definition . . . 15

2.2.2 Resource Description Framework . . . 15

2.2.3 ClaimReview . . . 15

2.2.4 Relevant KGs in the literature . . . 16

2.3 Misinformation . . . 17

2.3.1 Tasks . . . 17

2.3.2 Datasets . . . 18

2.3.3 Methods . . . 22

3 Detection of Misinformation-related Factors **27**
3.1 Detecting Conspiracy Theories . . . 27
3.1.1 Related Work . . . 28
3.1.2 Approach . . . 28
3.1.3 Results and Analysis . . . 31
3.2 Detecting Persuasion Techniques . . . 33
3.2.1 System Description . . . 33
3.2.2 Results . . . 37
3.2.3 Discussion . . . 38
3.3 Conclusion . . . 40

4 Definitions Matters **41**
4.1 Methodology . . . 41
4.2 Definition Understanding . . . 42
4.3 Results . . . 43
4.3.1 Conspiracy Theory Classification . . . 43
4.3.2 Definition Understanding Tests . . . 44
4.4 Additional experiments . . . 45
4.4.1 Results . . . 46
4.5 Conclusion . . . 47

5 Automatic Detection of Factors in Social Media Posts **51**
5.1 Detecting Emotion, Sentiment, Political Leaning . . . 51
5.1.1 Related Work . . . 52
5.1.2 Methodology . . . 53
5.1.3 Results . . . 53
5.2 Detecting Tropes . . . 59
5.2.1 Introduction . . . 59
5.2.2 Task Definition . . . 63
5.2.3 Dataset . . . 65
5.2.4 Models . . . 68
5.2.5 Experiments . . . 68
5.2.6 Related Work . . . 72
5.3 Conclusion . . . 73

6 CimpleKG: a Knowledge Graph For Explaining Misinformation **77**
6.1 The CIMPLE KG . . . 77
6.1.1 Connecting Misinformation, Reviews, Factors and Entities . . . 78
6.1.2 The Cimple KG Data Model . . . 79
6.1.3 Collecting and Integrating Newly Published Fact-Checks . . . 79

6.1.4 Integrating Static Datasets with the Fact-checks . . . 84
6.1.5 CIMPLE KG Statistics . . . 84
6.1.6 Fact-checkers and Language Statistics . . . 84
6.1.7 CIMPLE KG Use Case and Usage . . . 85
6.2 Cimple KG Explorer . . . 88
6.3 Conclusion . . . 90

7 Novel Textual Similarity Concepts **91**
7.1 Fact-Checking Similarity - Narrative vs Implication . . . 91
7.2 Entity-Based Similarity - Entity vs Concept . . . 92
7.3 Comparing automatic approaches for document matching . . . 95
7.3.1 Using Sentence-BERT . . . 95
7.3.2 Using Graphs . . . 95
7.3.3 Comparison of the approaches . . . 96
7.4 Comparing Documents With Different Granularity - Long vs Short . . . 97
7.4.1 Data . . . 98
7.4.2 Method . . . 98
7.4.3 Results . . . 99
7.5 Conclusion . . . 99

8 Conclusion and Perspectives **101**
8.1 Conclusion . . . 101
8.2 Perspectives . . . 103

A Appendix **105**
A.1 Appendix of Chapter 3 . . . 105
A.2 Appendix of Chapter 4 . . . 106
A.2.1 Examples of Definitions . . . 106
A.2.2 Prompt Description . . . 108
A.2.3 Recommendations For Practical Use . . . 109
A.3 Appendix of Chapter 5 . . . 110
A.3.1 Reproducibility . . . 110
A.3.2 Data Collection . . . 112
A.3.3 Additional Experimental Results . . . 112
A.3.4 Error Analysis . . . 115

Publications list **119**

Résumé en français **121**
1.1 Introduction . . . 121
1.2 Travail Connexe . . . 123

1.2.1 Traitement Automatique du Langage Naturel (TALN) 123
1.2.2 Graphes de Connaissances (KG) 124
1.3 Détecter les caractéristiques de désinformation 125
1.3.1 Détection de théorie du complot dans les tweets 125
1.3.2 Détection de techniques de persuasion dans les memes 125
1.4 Les définitions comptent 126
1.5 Détection automatique de caractéristiques textuelles dans les publications sur les réseaux sociaux 127
1.5.1 Émotion, Sentiment et Biais politique 127
1.5.2 Tropes 128
1.6 Cimple KG 129
1.7 Nouvelles concepts de similarité textuelles 130
1.8 Conclusion et Perspectives 131
1.8.1 Conclusion 131
1.8.2 Perspectives 132

Bibliography **149**

List of Figures

1.1 Example of a fact-check from the Snopes organization 2
1.2 Notes per month from the Community Notes program 3

2.1 Hierarchical structure of the persuasion techniques 20
2.2 Example of textual annotation using ChatGPT 25

3.1 Graphical representation of our model to detect conspiracy theories in tweets . 29
3.2 t-SNE visualization of node embeddings. Stars represent average position of each class. 30

4.1 MCC score on the test set. Error bars show the minimum and maximum values (5 random seeds) 44
4.2 Average MCC for different LLMs on different definitions settings 46
4.3 Classifications results of Zephyr-β using different definitions for each conspiracy theory 47
4.4 Average MCC for Zephyr-β using different classification settings 48
4.5 F1 score for propaganda classification using GPT-3.5-turbo 48

5.1 Distribution of the labels for the sentiment feature 56
5.2 Distribution of the labels for the emotion feature 57
5.3 Distribution of the labels for the political bias feature 58
5.4 Correlations between tropes using the Pearson coefficient. 67

6.1 Illustration of the CimpleKG data model. 80
6.2 Data collection and processing pipeline for gathering ClaimReviews. 80
6.3 Amount of fact-checks created for each country. 87
6.4 A screenshot of explorer.cimple.eu, an exploratory search engine to browse Cimple KG 89
6.5 A screenshot of the detailed view of the Cimple KG explorer 89
6.6 A screenshot of the document view in the Cimple KG explorer 90

7.1 Example of a use-case for entity and concept similarity 94

7.2 Pipeline for computing similarity between short and long documents, using local-global coefficient x and chunk i. 99
7.3 Average rank of the different types of claims depending on the coefficient x . . 100

A.1 Correlations between tropes using the Pearson coefficient on the Vaccine subset. 113
A.2 Correlations between tropes using the Pearson coefficient on the Immigration subset. 114

List of Tables

2.1 Size of various Language Models. 14
2.2 Datasets related to misinformation detection 23

3.1 MCC results for each run on the test set 31
3.2 Datasets considered for training our models. 35
3.3 Results on the dev set of some of the models we tried. Other models with different combination of parameters are used in the ensembling and not showed here due to space, but obtain similar performances. 37
3.4 Results on the test set with our ensembling model, translating non-English languages to English. 38
3.5 Results of our ensembling model on the dev set, per-class. 39

4.1 Performance of the LLM and transformer models using macro-averaging. . . . 43
4.2 Results of the two definition understanding tests based on semantic similarity and classification results. Top row contains Spearman's correlations of similarity between EG and HW definitions, and performance of EG zero-shot classifiers. The bottom row contains correlations of similarity between pairs of EG definitions, and Cohen's kappa of their classification. 45

5.1 F1-Score of the models on a validation set 54
5.2 Examples of tweets from all the datasets. Ground Truth indicates the label of the tweet in its original dataset, while Predicted Label is the output of one of the trained models 60
5.3 (Cont.) Examples of tweets from all the datasets. Ground Truth indicates the label of the tweet in its original dataset, while Predicted Label is the output of one of the trained models 61
5.4 Examples of tropes occurring in tweets and frequency of their presence in our dataset. 74
5.5 F1-score results for our models for each trope, the 'None' class, and weighted average across the dataset. 75

5.6 Proportions of conspiracy and tropes in respective datasets. Ground truth in italics. 75
5.7 Proportions of persuasion techniques and tropes in respective datasets. Ground truth in italics. 75

6.1 Recollected percentages from the top 30 fact-checkers. Total recollection percentage: 71.07%, average recollection percentage: 50.87% 83
6.2 Statistics of the static datasets integrated into CimpleKG. 84
6.3 Distribution of ClaimReview languages for the fact-checkers found in continuously updated fact-checkers data. 85
6.4 Top 10 countries with the most fact-checkers. 86
6.5 Top 10 countries with the most fact-checks. 86
6.6 Top 15 fact-checking organizations with the most fact-checks. 86

7.1 Examples of pairs using the labels defined in 7.1 and 7.2 93
7.2 Results for retrieving Fact-checking matches 96
7.3 Results for retrieving Narrative matches 96
7.4 Results for retrieving entity matches 96
7.5 Results for retrieving concept matches 97

A.1 F1-score per class on subsets of the test data. Models are trained on different subset of the train set. V stands for Vaccine and I for Immigration. 112
A.2 F1-score per class on subsets of the test data. Models are trained on different subset of the train set. V stands for Vaccine and I for Immigration. 113
A.3 (continued) F1-score per class on subsets of the test data. Models are trained on different subset of the train set. V stands for Vaccine and I for Immigration. . . 114

List of Abbreviations

AFP Agence France Presse.

AI Artificial Intelligence.

BERT Bidirectional Encoder Representations from Transformers.

CT-BERT CovidTwitter-BERT.

GCN Graph Convolutional Network.

GNN Graph Neural Network.

GPT Generative Pre-trained Transformer.

GRU Gated Recurrent Unit.

IFCN International Fact-Checking Network.

KB Knowledge Base.

KG Knowledge Graph.

LLM Large Language Model.

LM Language Model.

LSTM Long-Short Term Memory.

NLP Natural Language Processing.

NN Neural Network.

OPT Open Pre-trained Transformer.

PPO Proximal Policy Optimization.

RDF Resource Description Framework.

RLFH Reinforcement Learning with Human Feedback.

RM Reward Model.

RNN Recurrent Neural Networks.

STS Semantic Textual Similarity.

SVM Support Vector Machine.

TF-IDF Term Frequency-Inverse Document Frequency.

URI Uniform Resource Identifier.

URL Uniform Resource Locator.

WHO World Health Organization.

Chapter 1

Introduction

1.1 Motivation

1.1.1 Fake News, Misinformation, Disinformation

Misinformation has become a major concern around the world during the last few years, especially in online media. It has impacted a variety of topics, including politics (US Elections [17, 43, 52], Brexit [62], etc.), nature (climate change [140], Australian bushfires [149], etc.) and health (Vaccine [11, 151], COVID-19 [45, 89], etc.). Indeed, the spread of fake or misleading content about the Coronavirus has been described as an *infodemic*[1] by the World Health Organization (WHO). In 2019, Americans rated fake-news as the 5th biggest problem in the country, ahead of crime, climate change, racism or sexism [94].

The traffic of information shared on the internet, on social media or news articles for example, is increasing[2]. On X (formerly Twitter), for example, there are over 300 million posts per day [109]. This information can take many forms, such as text, image, video, sound, or metadata. The multi-modal nature of the data makes the task of fact-checking even more complex [3].

Many definitions of misinformation exist in the literature. In most works, *misinformation* refers to sharing false information, without the intent to harm [148]. On the other hand, *disinformation* is defined as sharing false information with the intent to harm by the user. In our work, we will use the term *misinformation* as a broad term regarding false or misleading information shared online.

[1] https://www.who.int/health-topics/infodemic
[2] https://www.domo.com/learn/infographic/data-never-sleeps-5

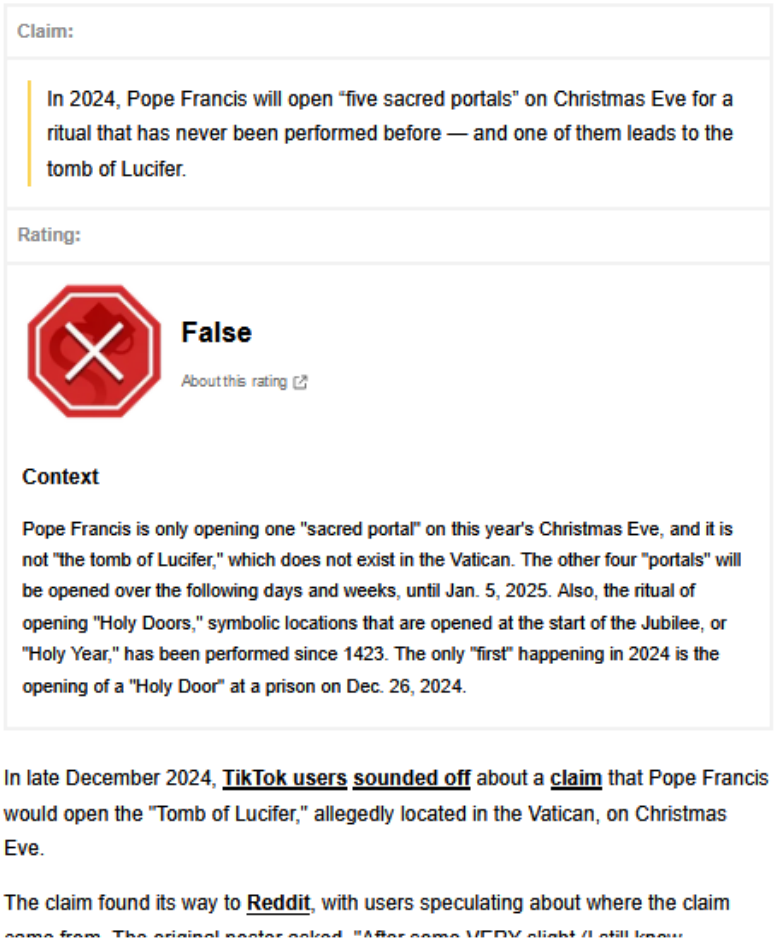


Figure 1.1: Example of a fact-check from the Snopes organization

1.1.2 Fact Checking

Nowadays, some organizations are dedicated to fact-checking, and debunk viral fake-news or fabricated content, with trusted sources. However, while it is easy to spread misinformation, it is much harder to detect and debunk it. According to [146], fake news spread six times faster than the corrected claims.

The International Fact-Checking Network (IFCN) is a worldwide community of fact-checkers. It supports many organizations such as AFP fact checking, Snopes, Politifact or Africa check with resources and networking. This community of fact-checkers publish multiple articles per day to verify the accuracy of some information (text, image, video, rumors) that originated on social media, political debates, TV, etc. The organization usually issues a rating along with an explanation and trusted sources to justify it. Figure 1.1 shows an example of a fact-checking article from the Snopes.com organization. We can see the claim being fact-checked, the rating and an explanation of the rating.

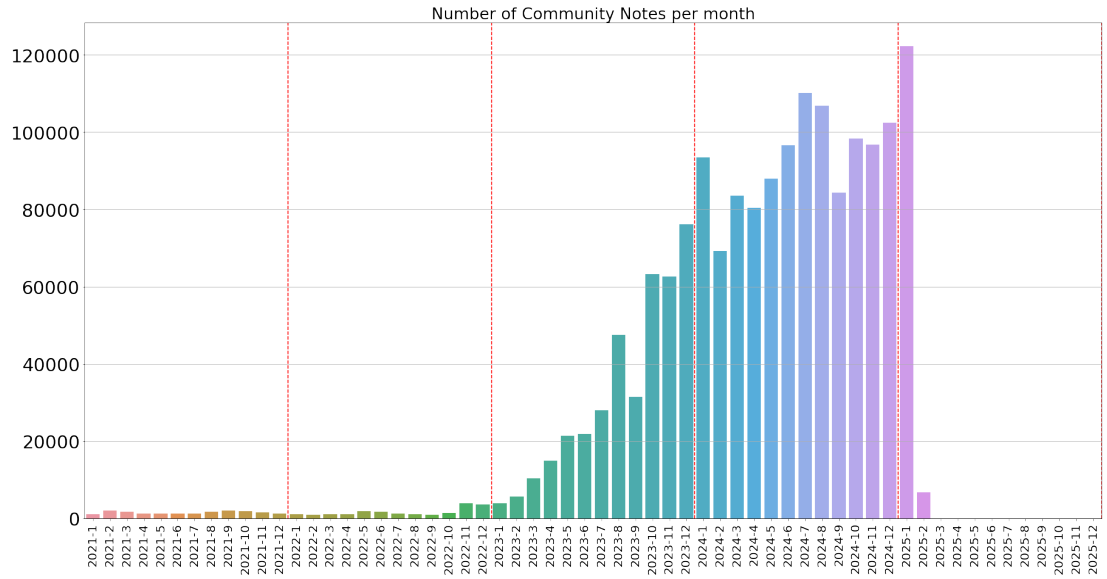


Figure 1.2: Notes per month from the Community Notes program

Some social media platforms have also implemented their own fact-checking approaches. For example, Meta (Facebook, Instagram and Threads) works in collaboration with the IFCN[3] to address misinformation spread on its platforms, by tagging media items to inform other users of misinformation content.[4] Alphabet (Google, YouTube, etc.) also proposes solutions to reduce the impact of fake-news on its services. For example, during the COVID-19 pandemic, they removed videos that posed harmful risks by contradicting health recommendations suggested by the WHO[5]. Lastly, X launched its community-driven fact-checking program named Community Notes (formerly Birdwatch) in 2021. The program allows enrolled users to create 'notes' reviewing the veracity of a tweet, while other enrolled users rate the 'note'. If a 'note' receives enough positive ratings, it is officially published and appears as additional context to the reviewed tweet for all users of the platform to see. The program has shown some positive results, with decent agreement with fact-checkers on claim verification [123]. However, it also raises concerns as the rating system could be abused by partisans of a common group, challenging content from those with whom they disagree politically [5, 123]. The program has seen a steady increase of popularity since its release, now counting more than 100 thousand notes per month (see Figure 1.2). This growth illustrates both the scaling challenge and public engagement potential.

[3] https://www.facebook.com/business/help/2593586717571940?id=673052479947730

[4] As of January 2025, this program will however be discontinued in US and replaced by a programe named Community Notes, https://transparency.meta.com/en-us/features/how-fact-checking-works/

[5] https://safety.google/intl/en_uk/stories/fighting-misinformation-online/

1.1.3 Natural Language Processing

Natural Language Processing (NLP) is a computer science field focusing on understanding, generating and processing human language. It leverages advancements in Artificial Intelligence (AI) to provide applications in a wide range of topics, such as health, finance or entertainment. The task of understanding language is complex, as natural human language can be ambiguous and heavily context-dependent. AI models often require human annotations during their supervised training. However, humans rely on intuition to infer missing context from conversations, which makes it difficult for models to reach human-like performances. The need for training model without explicit supervision has led researchers to focus on unsupervised or self-supervised methods.

Recent breakthroughs in NLP include the development of so-called foundation models, such as ChatGPT, which have completely shaken-up the landscape of AI-based applications and NLP-focused research. These conversational agents consist of very large models trained on an enormous amount of data that can perform a wide range of downstream tasks. They have shown good performance in annotating data [49], and are even outperforming crowd-workers in some annotation tasks [49].

Naturally, NLP has also seen numerous usage to specifically combat online misinformation. For example, research has focused on analyzing the spread of fake-news, misinformation detection or content moderation. As the whole ecosystem surrounding misinformation rely on data from many sources, machines are able to learn useful patterns. For example, AI has been used to analyze massive amount of social media posts to show how homogeneity and polarization in communities play an important role in the spread of misinformation [145].

1.1.4 Challenges

As stated in the previous sections, most platforms rely in some ways on fact-checkers, either directly (Meta through the IFCN) or indirectly (X with the crowd which is often citing fact-checkers as sources). However, professional fact-checking is time-consuming and does not scale well. While fake-news is easy to produce, countering it is time-consuming. Organizations specialized in fact-checking usually focus their effort on the most viral claims. This is why fact-checkers need tools to assist them to scale up their ability to fact-check. However, automatic fact-checking is a complex problem that uses scattered data from many independent actors, such as fact-checking organizations, community-driven fact-checking or knowledge bases. These sources also have different kind of metadata attached which makes it difficult to analyze content. Indeed, even fact-checkers make use of different sets of labels to describe the veracity of a claim.

Moreover, as the number of textual documents related to misinformation rises, the need of au-

tomatic approaches becomes clearer. Understanding the subtle aspects of text is key to better understanding the spread of misinformation online. Recent automatic breakthrough have created approaches that can learn patterns from massive amount of information. However, these tasks rely heavily on human annotations, which are costly and challenging to produce. Also, current automatic approaches lack explainability [8], as neural networks are often cited as *"opaque"*. The aim of automatic fact-checking approaches is evolving to not only classify a claim in terms of veracity but also to explain why and how this result has been obtained. For example, the dataset and challenge FEVEROUS aims at retrieving evidence information from Wikipedia, and assessing how this information is used to provide a verdict [6]. This approach is closer to the one of fact-checkers, as they provide sources and reasoning behind their verdict.

LLMs have shown very good "out-of-the-box" performances. However, their ability remains vastly untested for very specific tasks, and it is useful to benchmark their performance. These large models are also "opaque", and trying to understand how they function is a large part of current research. As society uses LLMs more and more due to their powerful reasoning capabilities, we have to analyze how they work. Indeed, models are used in numerous ways around us, but little attention is paid to how they achieve their results. As more powerful models are gated behind API usage, it becomes even harder to identify their functioning process. Prompting such models represents ways to probe their reasoning aspects but require analysis to draw insightful conclusions.

As fact-checker face time-constraints, they cannot focus on every false claim online, and only focus on the most viral ones. Before fact-checking a claim, they also have to spend some time making sure the claim has not already been fact-checked before. In a 2020 survey[6], more than 44% of fact-checkers needed a tool that will help them identify previously-checked claims. According to [128], *"viral claims often come back after a while in social media, and politicians are known to repeat the same claims over and over again"*. Automatically identifying previously fact-checked claims rely heavily on textual similarity research using textual embeddings. However, training models for that task is difficult as it relies a lot on human intuitions. Moreover, this approach is not fine-tuned to fact-checking applications. For example, it is not suited for long documents, such as news article, which appear regularly in the fact-checking pipeline.

We can summarize the main issues around three challenges:

- **Structural Fragmentation**: Misinformation-related data exists in isolated silos across platforms, organizations, and formats. This fragmentation creates barriers to comprehensive analysis and prevents the development of unified approaches to misinformation detection and analysis.
- **Contextual Understanding**: Current automated approaches often fail to capture the

[6] https://fullfact.org/media/uploads/coof-2020.pdf

nuanced contextual factors that influence how information spreads and is interpreted. This includes emotional, political, and narrative dimensions that are crucial for understanding misinformation dynamics.

- **Adaptive Similarity**: Traditional similarity measures struggle to capture the complex relationships between misinformation instances, particularly across different content formats and temporal contexts. This limitation hinders effective tracking of misinformation evolution and reuse.

These fundamental challenges create an ecosystem where misinformation thrives in the gaps between detection, understanding, and tracking. Our work addresses these limitations through computational approaches that provide both foundational infrastructure and innovative analysis methods.

1.2 Research Questions

Given the challenges described in the previous section, we have defined the following research questions:

- **RQ1:** ***How to model relationships between all the different types of data used for fact-checking.*** As explained in section 1.1.4, misinformation-related documents are scattered from different sources. Tools are needed to structure data and normalize the metadata attached to it, as well as creating relationships between the different data-points.
- **RQ2:** ***How to better understand textual documents with the use of automatic approaches.*** Understanding the spread of misinformation starts from understanding the intricate aspects of each textual documents. Automatic approaches are suited for learning patterns from large amount of data, enabling us to analyze the levers of misinformation.
- **RQ3:** ***How to extend the notion of textual similarity for fact-checking applications.*** Semantic textual similarity is widely used for matching texts but could be improved for the retrieval of previously fact-checked claims. While it has shown some use in misinformation detection, it lacks explainability and is not suited for all out-of-the-box usages.

1.3 Contributions

To answer the research questions at hand, we have made multiple scientific contributions. We propose Cimple KG, a continuously-updated knowledge graph of misinformation-related data,

that regroups all the different documents used in fact-checking, as well as misinformation posts. It includes different types of data, such as social media posts, news articles, memes or fact-checking articles. This work directly addresses **RQ1** by not only modeling the relationships between all the different types of data used for fact-checking, but also integrating many interesting textual features of the documents.

We present novel automatic approaches to detect textual features in the textual documents previously mentioned. We focus on emotion, sentiment, political-leaning, conspiracy-theories, persuasion techniques and tropes. We define these textual features as *Factors*, representing dimensions that play an important role in our understanding of textual documents. These textual features are also added to Cimple KG, and we believe that the detection and analysis of these *factors* allow actors, such as researchers or fact-checkers, to better understand textual documents, addressing **RQ2**.

Lastly, we present novel similarity measures for textual documents, especially useful for fact-checking applications. As such, we propose to measure documents based on their ability to fact-check other documents, but also on their relatedness based on entities and concepts mentioned. We then compare documents with different granularity, by comparing claims and news articles. For those three novel similarity measures, we propose annotated datasets and methods to retrieve documents. This work mostly addresses **RQ3** as we go beyond standard semantic textual similarity and propose complementary approaches.

1.4 Outline of the Thesis

The remaining of this manuscript is structured as follows. In Chapter 2, we introduce relevant related works for understanding the content of this thesis. It covers research topics such as natural language processing, knowledge graphs and misinformation. We present models, datasets, frameworks and methods that are currently used in the state-of-the-art.

In Chapter 3, we use transformer-based approaches to compete in research challenges in the detection of misinformation-related textual features in social media posts. We participate in MediaEval 2021 and MediaEval 2022 on the detection of conspiracy theories in Tweets, published in [104] and [106] respectively. We propose an ensembling system that combines multiple CT-BERT models to reach state-of-the-art results and share the code on GitHub[7]. Moreover, we study the graph of user interactions to detect misinformation spreaders. We also participate in SemEval-2024 Task 4: Multilingual Detection of Persuasion Techniques in Memes, where we extensively explore different BERT-based models, loss functions, training data, etc. We obtain our best results when leveraging the hierarchical structure of the persuasive techniques. This work has been published in [108] and the code is available on

[7] https://github.com/D2KLab/mediaeval-fakenews

GitHub[8].

In Chapter 4, we study both the conspiracy theory and the persuasion technique detection tasks by exploring the performance of Large Language Models (LLMs). We compare LLMs such as gpt-3.5-turbo, gpt-4o, Llama2 and Zephyr in zero-shot classification. Then, we explore how class definitions impact the classification results of LLMs. Lastly, we propose a method to generate class definitions based on examples. We find that improving definitions of the class labels has a direct consequence on the downstream classification results. This work has led to a publication [105] and we share our code on GitHub[9].

In Chapter 5, we focus on using transformer-based models to detect emotion, sentiment and political-leaning in social media posts. Additionally, we propose a dataset annotating a novel textual feature representing easily recognizable devices used in narratives to convey a specific theme or idea, called Tropes. We release an annotated dataset and propose approaches to detect Tropes on social media. We also explore correlations between the aforementioned textual features, revealing useful insights. For example, we find that conspiracy theories are usually promoted with negative sentiment and right political bias, which might reflect the inclination of conservatives towards anti-science information. The content of this chapter has been published in two publications [107] and [44], while we share our code on GitHub[10][11].

In Chapter 6, we release Cimple KG, a daily-updated knowledge graph of misinformation-related data. It contains multiple datasets containing social media posts, news articles and fact-checking documents. Cimple KG is additionally enhanced with the extraction of entities, or the factors present in the textual documents. The knowledge graph contains more than 16 million triples and represents, to the best of our knowledge, the largest up-to-date resource of misinformation research. Cimple KG has been the subject of a publication [22], and is shared under multiple forms[12][13][14].

In Chapter 7, we extend the notion of similarity measure in textual documents by creating novel datasets and showcasing their application. We first decompose notions of similarity useful for fact-checking, focusing on textual entities, concepts or narratives. We annotate pairs of tweets/claims, and we compare two different retrieval methods. We also define a notion of textual similarity for documents with different granularity (long vs short documents), by experimenting with previously fact-checked claim retrieval in the context of news articles. We propose a retrieval approach that uses local or global information in a long text to consistently

[8] https://github.com/D2KLab/semeval-2024-task-4
[9] https://github.com/dkorenci/gpt-def-zeroshot
[10] https://github.com/D2KLab/covid-twitter-discourse-analysis
[11] https://github.com/Tireswind/ADTIST24
[12] KG on GitHub: https://github.com/CIMPLE-project/knowledge-base
[13] Review data on GitHub https://github.com/MartinoMensio/claimreview-data
[14] SPARQL endpoint: https://data.cimple.eu/sparql

retrieve the most useful claim. We also release an annotated dataset to measure the performance of our system. Lastly, we present applications of textual similarity in the context of the matching of tweets/claims using the Community Notes program from the X platform as an example.

Finally, Chapter 8 concludes this work by summarizing each contribution. It explains how research question have been answered, and proposes some perspectives for future works to extend its use and tackle its flaws.

Chapter 2

Related Work

In this chapter, we cover the fundamental concepts useful for understanding the remaining of the thesis. We introduce recent research breakthrough in Natural Language Processing (Section 2.1), Knowledge Graphs (Section 2.2) and Misinformation detection (Section 2.3).

2.1 Natural Language Processing

Natural Language Processing (NLP) is a research topic that focuses on understanding textual language with the use of computational resource. It has seen a serious rise in interest in the last few years with the release of powerful language models with convincing reasoning capabilities, such as ChatGPT.

2.1.1 Language Models

Language models (LMs) are probabilistic models that solve many textual-related tasks, such as predicting the next word in a paragraph, textual translation, or classifying the emotion used in a sentence. Historically, LMs used n-grams to represent textual information, and inferred knowledge from the limited context. Other approaches used recurrent neural networks (RNNs) to take advantage of the sequential structure of text. For example, Long-Short Term Memory (LSTM) and Gated Recurrent Units (GRUs) models are typically effective at leveraging long term context. In 2016, they had been established as the state-of-the-art approaches for many natural processing tasks.

Neural approaches also allowed the production of vector representations of words, named embeddings. In particular, the Word2Vec [90] model is able to capture the semantic meaning of words in the embeddings, allowing mathematical operations on the representation of the words and computation of distance between the vectors. Indeed, the embeddings of two words are close if the words have a similar meaning. A popular example showcase the subtraction

of the embedding of the word 'King' with the embedding of the word 'Man', followed by the addition of the embedding of the word 'Woman'. The word with the closest embedding to the resulting embedding would be the word 'Queen', showcasing the ability of Word2Vec at representing words.

The attention mechanism allows the modeling of relationships between words regardless of the distance between them. In particular, the 'multi-head self-attention' was used as the only building block of the "Transformer" architecture [144] introduced in 2017. This model has an encoder-decoder structure, i.e. the encoder transforms the input sequence (text) into a representation, and the decoder generates the output sequence from the latent representation. While the Transformer architecture is not the first to introduce this notion of encoder-decoder architecture, it is certainly the first to be entirely based on the attention mechanism. This architecture reached state-of-the-art results in different tasks while also allowing faster training times.

BERT

BERT is a transformer-based model designed to learn representations in a bidirectional fashion, from both the left and right contexts. This allows for a more robust and finer training that out-performed many other approaches in many different tasks. BERT is also a great model to fine-tune on specific tasks. While it has been trained on masked language model and next sentence prediction tasks, it can be fine-tuned on sentence classification tasks, like emotion detection for example. This ability to be fine-tuned on any domain combined with its high performances make it the go-to approach for almost any textual-based task.

The popularity of the BERT model led to many follow-up architectures that reuses some of the main building blocks. RoBERTa [83] improves the pre-training approach by changing some hyper-parameters to make it more robust. AlBERT [74] focuses on reducing the number of parameters of BERT to increase the training speed an lower memory requirements. DistilBERT [124] uses knowledge distillation during pre-training to reduce the overall size of the model. DeBERTa [59] improves on BERT and RoBERTa by introducing a disentangled attention mechanism and an enhanced mask decoder. The research community has also released a multitude of pre-trained BERT models on some specific tasks. For example, BERT-HarMe[1] is fine-tuned on multiple datasets[2] [70, 134] about harmful/hateful speech in memes. COVID-Twitter-BERT (CT-BERT) [95] is a pre-trained model on large corpus of Twitter data on the topic of COVID-19. It is specifically designed to be used on downstream tasks that require additional knowledge on those topics.

[1] https://huggingface.co/limjiayi/bert-hateful-memes-expanded
[2] https://github.com/di-dimitrov/harmeme

Lastly, BERT and its successors have been used successfully on sentence-pairs tasks like Semantic Textual Similarity (STS), even if the architectures were not designed to accommodate such task. Indeed, the task requires the computation of all pairs of sentences, which is very inefficient. This problem has been addressed by Sentence-BERT [119], a BERT-like model that uses siamese and triplet network structures to generate sentence embeddings. Those embeddings can then be compared to measure the similarity between sentences, reducing the computation time significantly while maintaining the performance.

Large Language Models

While BERT is a popular choice for researchers for classification tasks, it does not allow for generation of texts. In 2018, OpenAi proposed a generative model based on the transformer architecture called 'Generative Pre-trained Transformer' (GPT) [115], using a large corpus of text with a novel training approach, using both unsupervised training and supervised fine-tuning. This model led the way for many subsequent models. GPT-2 (2019) [116] scaled-up the number of parameters and training data tenfold and was designed as a general-purpose learner. Indeed, the model performs well on tasks without any explicit supervision during training, including question-answering, summarization and translation. Ultimately, GPT-3 (2020) [19] showcased an even larger model, with even greater task-agnostic performances. The GPT-2 model was only released partially at first, as OpenAI suggested the model could be misused. It was eventually released fully, as the last open-weight model from OpenAI.

However, OpenAi is not the only actor in LLM research. For example, EleutherAI proposes GPT-Neo [14] in 2021, an open-source GPT-3 inspired model trained on a very large corpus of text called "The Pile" [47]. They also released GPT-J (2021) [147], and GPT-NeoX (2022) [13] as follow-up open-source models able to compete with closed-weight models from OpenAi. In 2022, Meta also released OPT [155] as another open-weight models, and BigScience publicly released the code of their model BLOOM [153] to help future research of LLMs.

In late 2022, OpenAi released ChatGPT, a model trained to generate human-like conversational responses. Powered by GPT-3.5, it is not trained to complete text, but rather to behave like a chat-bot. This is done following a three-step training framework. First, the GPT-3.5 completion model is fine-tuned on a prompt/output dataset handcrafted by humans using supervised learning. Second, Reinforcement Learning with Human Feedback (RLHF) [100] is used to train a Reward Model (RM) that can evaluate the output of the generative model based on a prompt. In practice, humans are presented with a prompt and multiple outputs, and rank the outputs from best to worst. The RM is then trained to learn the human preference. Lastly, the GPT-3.5 supervised model in step 1 is fine-tuned using Proximal Policy Optimization (PPO) [127] to generate outputs that will receive a high reward by the RM. This method is used to train both GPT-3.5-turbo and GPT-4 models. Training models to generate human-like conversations is

Model	Size (# of parameters)	Year
BERT-base	110 M	2019
BERT-large	340 M	2019
GPT-1	117 M	2018
GPT-2	1.5 B	2019
GPT-3	175 B	2020
GPT-Neo	2.7 M	2021
GPT-J	6 B	2021
GPT-NeoX	20 B	2022
OPT	175 B	2022
BLOOM	175 B	2022
GPT-3.5-turbo	*undisclosed*	2022
GPT-4	*undisclosed*	2023
Llama-1	65 B	2023
Llama-2	70 B	2023
Llama-3.1	405 B	2024
Claude	*undisclosed*	2023
Mistral	7 B	2023
Zephyr	7 B	2023
Alpaca	7 B	2023

Table 2.1: Size of various Language Models.

the most popular approach today, and has been adopted by many actors. Companies, such as Google (Gemini [137], 2023), Anthropic (Claude [7], 2023) or Meta (Llama-1 [138], 2023) all propose models with these capabilities. While some are closed-sources, other models have their weights accessible to allow further research contributions to the domain. For example, Meta's Llama or Mistral's self-titled model [66] have been used to create Alpaca (2023) [135] and Zephyr (2023) [142], smaller models (7 Billion parameters) with good capabilities.

As stated previously, an interesting feature of LLMs is their ability to perform well on tasks without explicit supervision during training. This is called "zero-shot learning", and is tested directly by prompting the model. Similarly, LLMs are proficient at understanding a task from very few examples. Usually, some examples of a task are shared in the prompt before asking the model to annotate a data point. This technique is often called "in-context few-shot learning", and has been studied extensively in the literature [19, 37].

With the constant increase in computational power available comes the ability to train larger and larger models, on larger and larger corpus. Scalability is an important factor of the success of LLMs. As displayed in Table 2.1, the number of parameters of language models grow exponentially. In this work, we refer to 'Large' language models (LLMs) for language models with more than a billion parameters.

2.2 Knowledge Graphs

2.2.1 Definition

Knowledge Graphs (KGs) are a type of data-structure that uses graphs to represent relationships between data-points. KGs propose a unique way of structuring data, that allows for some unique usage. For example, they are broadly used by the semantic web community to interlink web-pages, entities or abstract concepts for applications such as search engines [156] or recommender systems [54]. They are also widely used in social network research, where nodes represent users and edges the interaction among them. The KG allows computing metrics such as the reach of a user, or its centrality, which helps determine the spread of information through the network.

2.2.2 Resource Description Framework

The Resource Description Framework (RDF) is a method to model and represent KGs. It is based on a triple statement format that represent any object in the graph. The statement represent the subject, the predicate and the object. Both subject and object are nodes in the KG, while the predicate is a directed edge. The nodes can represent objects, represented by Unique Resource Identifiers (URIs) or values, while edges represent the relationship between both nodes. For example,

Schema.org

Schema.org[3] is a website that propose a standardized data model to re-use in many contexts. It defines **types** and **properties** that model existing objects and relationships. For example, it defines a type *SocialMediaPosting*[4], having several properties such as *sharedContent* that can be used to link an instance of a social media post (e.g. a Tweet) with the content shared (e.g. a video). The SocialMediaPosting also inherit from properties of its parent classes, *Article*, *CreativeWork* and *Thing*, such as *wordCount*, *author*, *dateCreated* or *url*.

2.2.3 ClaimReview

ClaimReview is a structured data markup format that is part of the Schema.org vocabulary. It is used by fact-checking organizations to publish specific details about the claim being examined, the fact-checking verdict (such as true, false, or misleading), the source of the claim, the date reviewed, and other relevant information. The ClaimReview format makes it possible to link fact-checking articles to fact-checked claims, commonly in the form of URL pairs, claim

[3] https://schema.org/
[4] https://schema.org/SocialMediaPosting

descriptions and ratings as well as information about what organization verified the claims. The structured nature of ClaimReview makes it an ideal format for consuming fact-checks, and it is used by search engines and social media platforms.

Google is one of the main sponsors of the ClaimReview project and provides both a user interface and an API for searching and retrieving ClaimReview data. The interface enables users to search claims using keywords and to navigate to the full fact-check article. An API is also available which enables programs to search ClaimReview data.[5]

The Google Fact Check explorer[6] is designed for exploring ClaimReview data using query terms and often returns a subset of ClaimReview objects and values. For example, the numerical value of `reviewRating.ratingValue` attribute is not usually returned, and instead only the value of `ClaimReview.textualRating` is provided. The numerical value is useful for comparing the level of the factuality of claims, whereas the textual rating is sometimes filled with textual descriptions in various languages and hence is more difficult to parse and compare. Other ClaimReview fields not returned through the Google Fact Check API are `appearances` and `firstAppearance`, which are used by fact-checkers to indicate where the claim appeared. This information is valuable for propagating claim assessments to the URLs where they appeared which can help determine the credibility of the source as a whole. This enables us to establish, for example, how many misinforming claims appeared on a certain news source or by a specific social media account.

2.2.4 Relevant KGs in the literature

Researchers have used KGs to store fact-checking related data, mostly centered on verified claims. However, claims are inherently tied to the context in which they appear, such as time-range, related social media posts, fact-checking articles, named-entities etc. The structured aspect of graphs allows representing the relationships between the documents. We present here the different KGs that focus on connecting claims to their context.

ClaimsKG [136]

ClaimsKG is one of the first KG datasets to provide a collection of fact-checked content. Their database relies on the ClaimReview data published by fact-checkers. The last release of ClaimsKG was in January 2023 and consisted of just under 75 thousand claims collected from 13 popular fact-checking websites.[7] The limitations of this resource are centered around the small number of fact-checking websites included in the ClaimReview crawl, infrequent

[5] Google ClaimReview API, https://developers.google.com/fact-check/tools/api/reference/rest/v1alpha1/claims/search.

[6] Google Fact Check Tools, https://toolbox.google.com/factcheck.

[7] ClaimsKG, https://data.gesis.org/claimskg.

updates at long intervals, and the narrow scope of the KG. Considering the rapid pace at which misinformation emerges and spreads, it is critical for any supporting dataset to include the most recent claims and their verification results and include a large variety of data sources.

The Database of Known Fakes (DBKF)

The Database of Known Fakes[8] is a more recent initiative aiming at enabling users to browse through previously fact-checked documents by known organizations. It collects new fact-checks and displays them in a web-based user interface, allowing to query the database with relevant filters, such as date, language, concepts, or authors. While DBKF shares daily-updated data, it still lacks a significant amount of fact-checks and does not allow search based on textual factors, or review label.

2.3 Misinformation

As stated in the introduction, misinformation has become a major research problem in recent years. Major research works study the spread of misinformation in social media networks or automatic fact-checking, with a strong emphasis on claims. In this section we preset useful misinformation-related research datasets that will be re-used in our work, as well as methods to perform claim verification, retrieval etc.

2.3.1 Tasks

Misinformation research can take many forms, for many different purposes. We discuss here the main tasks that are tackled by the scientific community.

Claim Worthiness Estimation

As social media generate an incredible amount of information, fact-checkers have to first filter-out claims that are not worthy of being verified. These claims could be simply non-factual, like opinions, or factual but unimportant, i.e. have no impact on society or not interesting to the general public. The first sub-task of the CLEF Check-That! 2022 challenge [96] consist of estimating the check-worthiness of a claim spanning COVID-19 and political topics. This is a multi-class classification problem, and is usually solved using transformer-based models.

[8] https://www.ontotext.com/company/news/the-database-of-known-fakes-a-valuable-eu-research-result/

Claim Verification

Claim verification consist of automatically verifying the veracity of claims. These claims usually correspond to those made online on social media, or by politicians during debates. This research field typically leverages online document databases, such as Wikipedia, or fact-checker organization websites to verify the veracity of the claims. Solutions tend to use retrieval systems to select candidates, and entailment models to infer the veracity. The latter is essentially a multi-class classification problem as the veracity labels are fixed (true, false, partially false, not enough information, etc.). Popular methods use TF-IDF in combination with transformer models to retrieve relevant candidates and pre-trained language models (BERT, GPT, etc.) for classification.

Claim Retrieval

To help fact-checkers, researchers have studied how to retrieve previously fact-checked claims that re-appear in other contexts, such as on social media. Indeed, as viral claims come back regularly, fact-checkers need to efficiently filter-out previously fact-checked claims for time constraints. In the CLEF Check-That! 2022 challenge [96], the goal of the second sub-task is to detect previously fact-checked claims in tweets, with respect to a collection of claims from different fact-checking organizations. Most approaches consider this task as a matching or a ranking problem. Researchers usually leverage algorithms such as BM25 for ranking all the claims for each posts, in combination with embeddings models such as Sentence-BERT [119].

Text Classification

Text classification is arguably the most popular NLP task. It usually refers to the use of models for the detection of textual features, for example emotion or sentiment. In the context of misinformation, those features can take the form of conspiracy theories [12, 75] or persuasion techniques [27, 34, 35]. Most often, the tasks are not binary and multiple classes have to be detected. For example, the Media-Eval-2021 [112] tasks consist of detecting nine COVID-19 related conspiracy theories in tweets. For persuasion technique detection, a sub-task of the Sem-Eval-2022-Task-3 [110] challenge consist of identifying the persuasion techniques used in each paragraph of a news article, among 23 proposed techniques.

2.3.2 Datasets

We present here the datasets related to misinformation on the web that will be used later in the following chapters. Table 2.2 summarizes the different dataset sources, sizes and tasks.

COCO: an annotated Twitter dataset of COVID-19 conspiracy theories [75]

This dataset is composed of 3,459 tweets annotated with regard to 12 different named conspiracy theories related to COVID-19. The annotation is broken down into mentions and support of the conspiracy, and each tweet can be related to multiple conspiracies. The dataset was built by searching keywords (e.g. '#coronarvirus', 'plandemic', 'microchip' or 'chemtrails') on Twitter during the period January 2020 to June 2021. The different labels annotated are: Suppressed cures, Behavior control, Anti vaccination, Fake virus, Intentional pandemic, Harmful radiation, Depopulation, New world order, Esoteric misinformation, Satanism, Other conspiracy theory and Other misinformation. The authors also provide the definition of each label that was given to the annotators.

SemEval-2024 Task 4: Multilingual Detection of Persuasion Techniques in Memes [34]

This dataset presents the annotation of persuasion techniques in online memes. It contains a total of 10,000 memes annotated with regard to 22 different persuasion techniques[9]. The dataset was created by scraping public Facebook posts in groups about political, health or societal topics, as well as Instagram for memes in non-English languages (North Macedonian and Arabic). In the context of this dataset, a 'meme' is defined as "a photograph style image with a short text on top of it". Memes can be annotated with multiple persuasion techniques. Lastly, persuasion techniques in this dataset belong to a hierarchical structure (see Figure 2.1).

COVID-19 Twitter Dataset with Latent Topics, Sentiments and Emotions Attributes (COVID LTSE Attributes) [56]

The COVID LTSE Attributes dataset contains 252 million of tweets from January 2020 to June 2021 related to the COVID-19 pandemic. The data was searched using keywords, such as 'wuhan' or 'corona'. This dataset is labeled with 17 attributes, such as topics or emotions, using probabilistic topic modeling and pre-trained models. Most notably, it contains the emotion attribute, which has been labeled using the *CrystalFeel*[10] pre-trained machine learning algorithm.

COVIDSenti: A Large-Scale Benchmark Twitter Data Set for COVID-19 Sentiment Analysis [99]

COVIDSenti is a dataset with labels for sentiment in COVID-related tweets. The tweets have been crawled from February 2020 to March 2020, using keywords such as 'coronavirus' or

[9] The full list of persuasion techniques and their definition can be accessed here: https://propaganda.math.unipd.it/semeval2024task4/definitions22.html.

[10] https://socialanalyticsplus.net/crystalfeel/

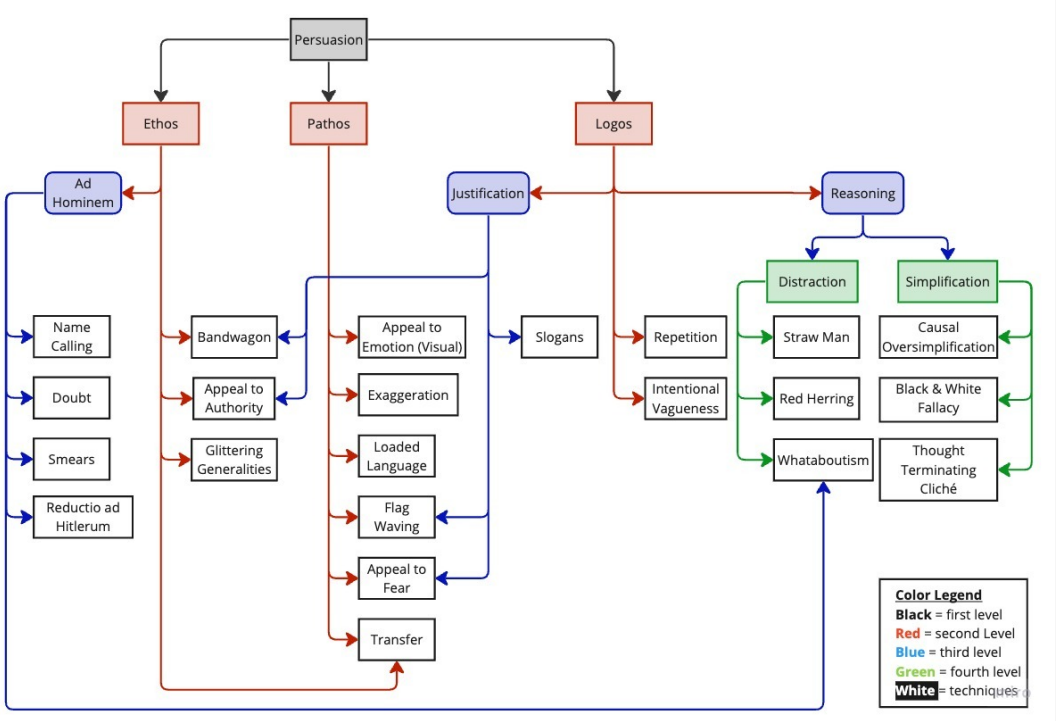


Figure 2.1: Hierarchical structure of the persuasion techniques

'Corona Outbreak'. The data is annotated with *TextBlob*[11] using the methodology described in [9] resulting in 90,000 annotations.

Troll Factories: Manufacturing Specialized Disinformation on Twitter (Russian Troll) [79]

The Russian Troll dataset contains 2.9 million of tweets from February 2012 to May 2018. The tweets are from accounts associated with the Internet Research Agency, which interfered during the U.S. 2016 presidential elections. A detailed analysis of the disinformation tactics used by this group of people is available in [78]. The data is labeled at the account level using five main categories ('Right troll', 'Left troll', 'Fearmonger', 'HashtagGamer' and 'NewsFeed').

COVID-19 Stance [50]

In the COVID-19 Stance dataset, tweets are labeled with a stance towards a topic related to the pandemic. The data was crawled from February 2020 to August 2020 using keywords ('coronavirus', 'covid-19', etc.) or hashtags ('#lockdown', '#washhands', '#socialdistancing', etc.). The topics are 'Anthony S. Fauci', 'keeping schools closed', 'stay at home orders' and 'wearing a face mask', and the annotation was done with Amazon Mechanical Turk. Since the release

[11] https://textblob.readthedocs.io/en/dev/

of this dataset, numerous tweets have been deleted or removed, and we were only able to retrieve 3,616 tweets to re-use in our work.

Community Notes

As described in the introduction (Chapter 1), many social media platforms have developed their own fact-checking tools. 'Community Notes' is the program launched by X, which features crowd-sourced fact-checking. It allows X users to identify potentially misleading tweets by writing 'notes', and other users to review 'notes'. This dataset is growing every day as the program receives more notes and reviews. As of late September 2024, the dataset contained more than millions of notes.

Community Notes Matching Dataset [123]

The dataset used in [123] contains a total of 9,851 tweets that have been labeled by Community Notes users with 'Misleading' or 'Not Misleading' labels, from January 2021 to September 2021. It also contains claims coming from professional fact-checker sources, including multiple information about the claim, such as the date, or the veracity label. This dataset contains pairs of tweets/claims, labeling if the tweet contains the professionally reviewed claim.

CLEF CheckThat! 2022 [96]

The CLEF CheckThat! 2022 dataset contains social media posts linked to fact-checking articles. It has 14,000 verified claims, collected on the Snopes fact-checker website, which cite the social media post being debunked. The goal of the dataset is to perform previously fact-checked claim retrieval.

AFP

The AFP dataset contains news articles collected through Agence France Presse (AFP). It has around 200k news articles about all journalistic topics (health, politics, events, sports, etc.) written in English. It contains some metadata such as the data published, the author, etc.

Propaganda corpus [25]

Lastly, the Propaganda corpus focuses on propaganda detection in news articles. It originally contains more than 451 articles annotated with 18 propaganda techniques[12], similarly to the SemEval 2024 dataset. The data was collected from 47 news outlets previously labeled by

[12] QCRI propaganda techniques, https://propaganda.qcri.org/annotations/definitions.html.

Media Bias Fact Check as being 'non-propagandist' or 'propagandist'. Each sentence in those articles are labeled at the fragment level and at the sentence level.

FEVEROUS (Fact Extraction and VERification Over Unstructured and Structured information) dataset [6]

The FEVEROUS is a popular choice to benchmark automatic fact-checking approaches. It contains more than 80k claims that have been annotated with evidence from Wikipedia pages (sentences, tables, etc.), with a label explaining if the evidence refutes or supports the claim. It was used during a challenge, where teams have explored methods to perform page, sentence and table selection, as well as verdict prediction.

AVERITEC [126] dataset

The AVERITEC dataset contains around 4.5k claims annotated with evidence from fact-checking websites in the form of question-answer pairs. The goal of this process is to break down the reasoning process of the fact-checker. For example, for the claim "*The USA has succeeded in reducing greenhouse emissions in previous years* (2020.11.02) - Morgan Griffith", the questions would be "Q1: *What were the total gross U.S. greenhouse gas emissions in 2007?*", "Q2: *When did greenhouse gas emissions drop in US?*" and "Q3: *Did the total gross U.S. greenhouse gas emissions rise after 2017?*". The dataset also contains a knowledge store of useful web pages for each claim, retrieved using the Google Search API.

2.3.3 Methods

Misinformation research focuses primarily on claims. First, researchers have studied the automation of claim verification using sources, and claim retrieval from databases of previously fact-checked claims from different fact-checking organizations.

Claim verification

The FEVEROUS dataset has been used during a challenge which showcased claim verification tasks. The winning team [16] first used the BM25 algorithm for page selection. They then used a multi-hop dense passage retrieval, followed by a BM25 filtering step, before using a fine-tune RoBERTa model to re-rank candidates. This pipeline is used for both sentence and table selection steps. Using two pre-trained TAPAS [60] models with a MLP, they compute the entailment between the claim and the candidates.

The AVERITEC dataset was used during an evaluation campaign [125], and the winning

Dataset	Source	Size	Tasks
COCO	X (Twitter)	3,459k posts	Conspiracy theory classification
SemEval-2024	Facebook + Instagram	10k posts	Persuasion technique classification
COVID LTSE Attributes	X	252 million posts	Emotion, sentiment, and topic detection
COVIDSenti	X	90k posts	Sentiment detection
Russian Troll	X	2.9 million posts	Political-leaning detection
COVID-19 Stance	X	3,616 posts	Stance towards 4 topics
Community Notes	X	1 million notes	Posts verification with notes and ratings
CN matching	X + Fact-checking organizations	9,851 matches	Claim Retrieval
CLEF CheckThat!	Snopes	14k matches	Claim Retrieval
AFP	News Articles	200k articles	News topic
Propaganda corpus	News Articles	451 articles	Propaganda technique classification
FEVEROUS	Wikipedia	80k claims	Claim verification
AVERITEC	Fact-checking organizations	4.5 claims	QA pairs for fact-checking reasoning process

Table 2.2: Datasets related to misinformation detection

team [121] mostly uses GPT-4o in a multi-stage claim verification process, composed of claim interpretation, questions generation, evidence retrieval, question answering, verdict prediction and verdict justification. For the evidence retrieval stage, the LLM creates search query texts. Documents in the KB and the textual queries are passed to an embedding model (gte–base-en-v1.5) to find the 5 most semantically close documents to the query. All other stages are performed in the prompt of the GPT-4o model.

However, automatic fact-checking often relies on opaque methods and suffer from credibility, as professional fact-checking will be preferred. Hence, the need to rather help fact-checkers verify claims more efficiently than to automatically verify their veracity.

Previously fact-checked claim retrieval

In [128], the authors explore multiple approach such as the BM25 algorithm and BERT-based models to rank claims (from fact-checking articles) based on its similarity with tweets. Their best results are obtained by combining both BM25 rankings with similarity scores computed on the embeddings of BERT models, and using rankSVM to rank claims.

The CLEF-2022 CheckThat! challenge [96] proposes to retrieve previously fact-checked claims that appear in tweets. During this competition, teams [98] have used transformer based architectures (BERT-based, GPT-based, etc) in combination with some text preprocessing steps. For example, the winning team [130] used sentence transformers (Sentence-T5) to select candidates and GPT-neo as a re-ranker of the candidates.

Detection of textual features

Transformer-based models have been studied extensively to detect textual features by the research community. For example, BERT has been used to detect sentiment in customer reviews [114], social media posts [67] and even finance-related topics [93]. It has also been used extensively to detect emotion in textual documents [1,2], and more broadly in various sentence classification tasks.

Transformer-based models, such as BERT and the likes, rely on annotated data to be fine-tuned. Indeed, after being trained on large amount of data on masked language model and next sentence prediction, they are fine-tuned on smaller dataset using supervised learning. Typically, the architecture of the model is modified, with the connection of the last layer of the model to a layer of neurons proportional to the number of output classes. In most cases, the model is trained using standard Cross Entropy or Binary Cross Entropy loss functions. However, not all classes are represented equally in the training set, which creates imbalance in the training. If this imbalance is not addressed, the model will have trouble detecting the

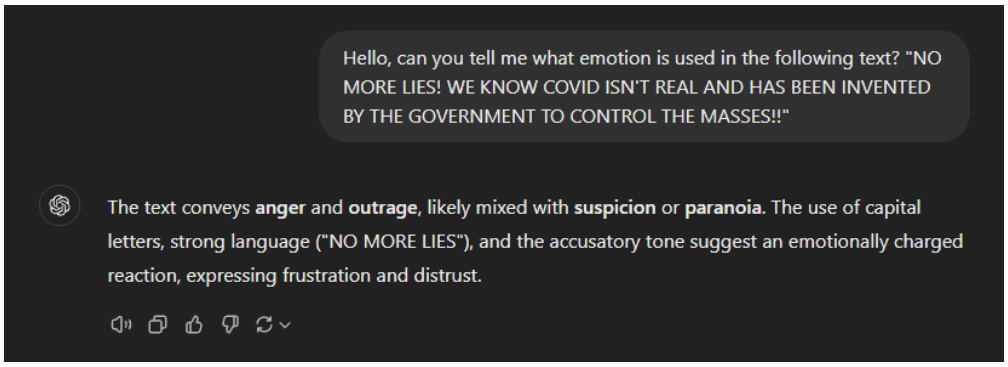


Figure 2.2: Example of textual annotation using ChatGPT

least prevalent classes. Some solutions require adding weights to the classification loss of each sample, inversely proportional to the prevalence of the target class. Another solution is to use Focal Loss [77], which down-weights the loss assigned to well-classified examples.

BERT-based models have been the most popular approach in many natural language processing tasks in the last few years. However, LLMs have recently seen a surge in popularity due to their ease of use and their 'out-of-the-box' performance on a multitude of tasks, including sentence classification. Indeed, researchers have extensively studied OpenAI's GPT-3 and 4 ability to detect emotion [15], sentiment [69] and more. As an example, in the SemEval-2021 Task 6 evaluation campaign [35], 80% of participants used BERT-based models to detect persuasion techniques. In a similar task proposed in the SemEval-2024 Task 4, only 65% of the proposed approaches used BERT-based models, and 52% used LLMs in their pipeline.

LLMs have a unique usage technique, known as 'prompting'. As models are trained to behave in a conversational manner, the annotation of data for text classification needs to be formatted as a discussion. Figure 2.2 shows an example of using ChatGPT to annotate emotion in text. As pictured, the model returns the answer in a full paragraph, also giving explanation of the annotation. While this is made for casual usage, it is not suited for large-scale experiment. Additional parsing or prompting techniques are required to extract a label from the model.

Many different prompting techniques exist to boost the performance of LLMs for textual classification [82][13]. For example, the task can be described to the LLMs through definitions, or using examples. The proper selection of the examples, the order in which they are shown to the model, and their number all play a significant role in the performance of the model [80]. Another prompting strategy, called "chain of thought", consist of breaking down the reasoning process of the task in a few simpler steps [150]. This technique has even been extended to consider multiple decision process in an approach called "tree of thoughts". These approaches have shown how much impact the design of the prompt has on performance of many complex

[13] In this work, the search for the best prompt is referred to "*Prompt template engineering*"

tasks, as they yield significantly better results than standard prompts. Lastly, LLMs have a limited context length and have shown decreasing performance when dealing with long prompts [81]. As a workaround, the “Retrieval-augmented generation” [76] technique is used to only give the model useful data. It consists of first querying a database to extract the relevant material and then adding it to the prompt. This method can improve results by suggesting relevant documents as well as reducing the prompt length.

Chapter 3

Detection of Misinformation-related Factors

In this chapter, we present approaches that aim at detecting textual features in social media posts. In particular, we focus on textual features related to misinformation, with the detection of conspiracy theories in Section 3.1 and persuasion techniques in Section 3.2. The content of both sections have been published in [113], [104] and [108], and the code is shared on GitHub[1][2].

3.1 Detecting Conspiracy Theories

In this section, we detail our submission to the MediaEval 2022 "FakeNews Detection" competition. This challenge is broken down into three tasks that aims at detecting 9 named conspiracy theories in tweets, as well as classifying misinformation spreaders in a user interaction graph. The full task description is detailed in [111]. The data used is a subset of the COCO dataset described in Section 2.3.2.

The first task of this challenge is a multi-label multi-classification problem in the tweet textual content. This type of problem has been studied extensively in the natural language processing (NLP) literature. Some popular baseline approaches to tackle this problem include statistical techniques, such as TF-IDF [92], or transformer-based models, such as BERT [33]. The second task is a node classification problem in a graph. This kind of task can be tackled with graph neural network based approaches, such as GraphConv [73] (GCN), or approaches that generate node embeddings, such as node2vec [53]. The third task is also a multi-label multi-classification problem, with both textual content from the tweet and information from the graph of user interaction. Our code is available on GitHub.[3]

[1] https://github.com/D2KLab/mediaeval-fakenews
[2] https://github.com/D2KLab/semeval-2024-task-4
[3] https://github.com/D2KLab/mediaeval-fakenews

3.1.1 Related Work

Transformer-based [144] models have achieved state-of-the-art performance for various NLP tasks [152]. These models are pre-trained on a large corpus of text, and can be fine-tuned for a specific task on a smaller corpus. An example is COVID-Twitter-BERT (CT-BERT) [95], which is a pre-trained model on a large corpus of Twitter data on the topic of COVID-19. This makes it suited for tasks 1 and 3 of this challenge. This year's task 1 is very similar to last year's task 3 [112], in which we participated [104]. In our previous experience, the CT-BERT model was the most performing one.

The second task requires methods that leverage graph data. Indeed, the provided data is composed of a graph of interaction between Twitter users, as well as some information about the user itself. More information about the data can be found in the task description paper [111]. This task of node classification can be tackled using node embedding techniques from sequence-based models (e.g. node2vec) or GNNs (e.g. GCN) [63]. Sequence-based models learn the embeddings of a node by using the structure of the graph and the neighbors of a node, without capturing any information about the node features. The node classification can then be done with different models, from the learned node embedding, using traditional classifier approaches such as Multi Layer Perceptron (MLP) or Random Forest (RF). The GNN-based approach optimize both the embedding and the classification task at the same time. It utilizes the structure of the graph, as well as the node features.

3.1.2 Approach

In order to tackle this challenge, we studied text-classification transformer-models for tasks 1 and 3, and node-classification models for tasks 2 and 3. Our approach leverages multiple CT-BERT models for text-classification and node2vec in combination with simple classifiers (MLP, RF) for node-classification. We also experimented with GNN without much success, and we do not report these results. In all experiments, we split the data into 5 stratified cross-validation sets.

Text Classification

First, we used some basic pre-processing on the text data. We replaced all emojis with their textual meaning using the *emoji* Python library.[4] We also removed the hashtag ('#' character) from the tweets. Next, we approached task 1 as a multi-label 3-way classification problem. We trained 5 CT-BERT models, one for each cross-validation fold, using a custom loss function. The last layer of our models has 27 dimensions, three for each of the 9 conspiracy theories (discuss, support, not related). We build 9 different Cross-Entropy losses, each measuring the

[4] https://github.com/carpedm20/emoji/

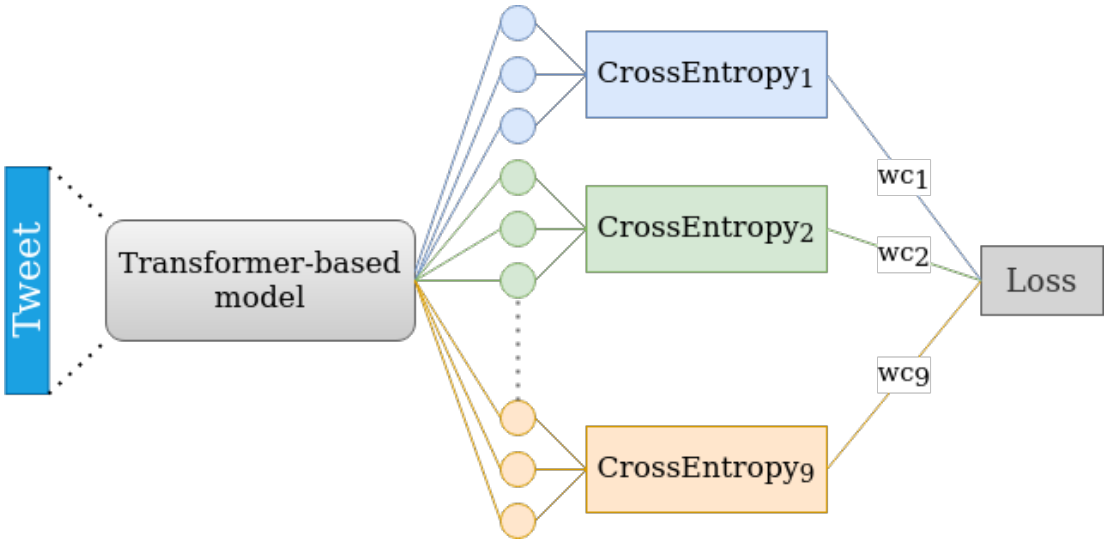


Figure 3.1: Graphical representation of our model to detect conspiracy theories in tweets

performance of the model at detecting one conspiracy theory. These Cross-Entropy losses are weighted independently, proportionally to the inverse of the frequency of the respective class in the training data. Then, the final loss is the unweighted sum of the 9 different losses. Figure 3.1 shows a graphical representation of the proposed model.

Node Classification

We used a node2vec model to generate node embeddings, and then used standard machine-learning classifiers to perform the node classification.

We first build the graph from the user-interaction data, using the *networkx* Python library [57].[5] This graph is composed of around 1.7 Million nodes (representing the Twitter users) and 270 Million directed edges (representing the interaction between the users). We run the *node2vec* algorithm on that graph, using *nodevectors* Python library. [6] We generate 10 random walks per node, of length 40, with the return parameter set to p=1 and the in-out parameter set to q=1/2. The dimension of the embeddings is 32. Figure 3.2 shows a t-SNE visualization [143] in two dimensions of the embeddings of the graph nodes. In orange we see normal users and in blue misinformation spreaders. The stars represent the average position of each class, showcasing that both classes have different distribution. We can also see clusters of users further away from the average.

Next, we train some popular machine-learning classifier algorithms available on the *scikit-*

[5] https://github.com/networkx/networkx
[6] https://github.com/VHRanger/nodevectors

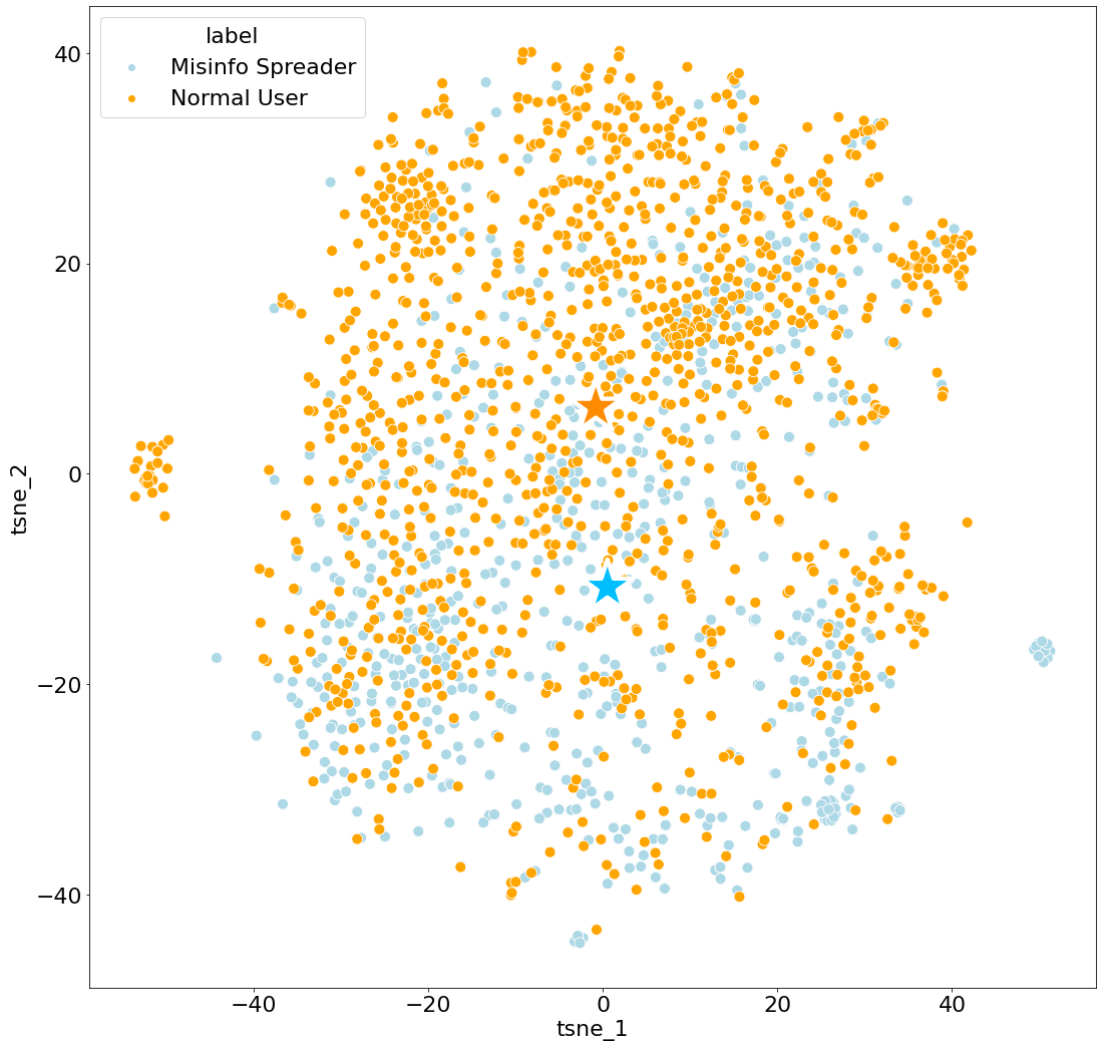


Figure 3.2: t-SNE visualization of node embeddings. Stars represent average position of each class.

Table 3.1: MCC results for each run on the test set

	Run	Model	Test MCC
Task 1	001	CT-BERT Ensembling	**0.710**
	002	CT-BERT	0.685
	003	CT-BERT Ensembling+'CD'	0.657
Task 2	101	node2vec+MLP Ensembling	0.327
	102	node2vec+MLP	**0.355**
	103	node2vec+RF Ensembling	0.253
	104	node2vec+MLP Ensembling+'CD'	0.295
	105	node2vec+MLP+RF Ensembling	0.259
	106	node2vec+MLP Ensembling	0.327
Task 3	201	(task 1) CT-BERT Ensembling	**0.719**
	202	(task 1) CT-BERT	0.676
	203	(task 1) CT-BERT Ensembling+'CD'	0.663
	204	MLP Fusion Ensembling	0.690
	205	MLP Fusion	0.676

learn Python library [103][7] to perform node classification. Those algorithms take as input the 32-dimension vector from the node2vec model and perform a binary-classification objective. Random Forest (RF) classifier obtained the best results. We also trained a MLP classification head as well.[8]

Tweet Classification Using Both Text and Graph Data

We used both graph and textual data for the third task of the challenge. We trained a classifier from the concatenation of both text and graph features, without any form of feature space normalization. Those graph features are the 32-dimension vector from the node embeddings, and the textual features are the 27-dimension vector output of the task 1 model. The training loss is similar to the one described in Section 3.1.2.

3.1.3 Results and Analysis

In this section, we first describe each run, and then analyze the main takeaways from the results. We present our results for this challenge in Table 3.1.

[7] List of tested classifiers: KNeighborsClassifier, GaussianProcessClassifier, DecisionTreeClassifier, RandomForestClassifier, AdaBoostClassifier, GaussianNB, QuadraticDiscriminantAnalysis and GradientBoostingClassifier

[8] The hyper-parameters for the MLP model are: layers of 32-16-8-1, dropout p=0.1, ReLU activation function

Runs description

For the first task, we performed an ensembling of the 5 models trained on each fold of the cross validation split (run 001). This ensembling was done with majority voting. In case of a draw between two classes, 1 would have priority over 2 and 3, and 3 over 2. This order follows the proportion of samples we have in the data. Run 002 is the single best model of the 5 models. Run 003 is the same ensembling as run 001, but with the 'cannot determine' labeling if there is less than 4 models out of 5 in agreement. The ensembling (run 001) obtained the best results.

The second task was more challenging and results are lower compared to task 1. We propose multiple ensembling methods from the models trained on each fold of the cross-validation split. We trained 5 MLP models and 5 RF models. Run 101, 103, and 105 are ensembling of those models. Run 106 is the same as run 101 with the 'cannot determine' labeling logic as in run 003. Run 102 is the best single MLP model from run 101. This run obtained the best results, out-performing the ensembling (run 101). We also submitted our run 101 as a sixth run (106) because we did not use any pre-trained model.

For the last task, we first propose the same models as the first task, without using the graph data. Runs 201, 202 and 203 are the same as runs 001, 002 and 003. Results are slightly different because the test data is different from task 1 and 3. Then, we also propose models using both text and graph data, using the approach described in Section 3.1.2. Run 204 is an ensembling of the MLP models trained on each fold of the cross validation split, and run 205 is the single best MLP model. The best performing model for this third task is the ensembling of CT-BERT models, only on text data (same as run 001).

Main takeaways

A first takeaway from these results is that the CT-BERT model is suited for text classification tasks and obtains very good results, even if slightly lower than in 2021 [104]. The ensembling of models trained on different cross-validation fold improves the results with textual models (run 001, 201 and 204), but not for the graph-based approach (run 101, 103 and 105). Also, the 'cannot determine' labeling did not improve the results over the normal majority voting (runs 003, 104 and 203). The overall approach for task 2 gives lower results than task 1, but the task is also more challenging. The MLP classifier from the node embeddings is a baseline that can definitely be improved. For example, we could try to reduce the size of the very large graph by removing some nodes in order to remove noise. For the last task, the MLP fusion model, taking both text-based features and graph-based features (run 204), did not improve on the text-only CT-BERT model (run 201). The fusion model could still be improved to correctly use both kinds of data and improve the overall results.

Comparing these results with last year challenge, we did not see a major improvement in the

overall score, even though we had access to more data this year. Regarding the results for each conspiracy theory, the best results are obtained for the 'Harmful Radiation/Influence' (0.830), 'New World Order' (0.830), and 'Population reduction' (0.876) conspiracies. The worst result is obtained for the conspiracy theory 'Antivax' (0.563). More data does not seem to correlate to better results for each conspiracy as well, since 'Antivax' has almost four times more data than 'Harmful Radiation/influence' and still perform significantly worse. This is partially due to the weighting of the loss function, which emphasizes the classes with a smaller number of samples.

3.2 Detecting Persuasion Techniques

In this section, we present the work done during our participation to the 'SemEval-Task4: Multilingual Detection of Persuasion Techniques in Memes" task. The data is briefly described in Section 2.3.2. The task breaks down into 3 sub-tasks; sub-task 1 uses the textual content of the meme to detect the persuasion techniques, sub-task 2a uses the whole image and text to detect the persuasive techniques, while sub-task 2b only consist of binary detection. In sub-task 1, a total of 20 persuasion techniques are used. We only describe our solution to tackle this sub-task 1.

Our approach consists of an ensembling model of our top-3 best models for each persuasion techniques. In our experiments, we reached the best results leveraging the hierarchical nature of the data, with hierarchical loss, and outputting ancestor classes. Our method can be reproduced using the code at https://github.com/D2KLab/semeval-2024-task-4.

3.2.1 System Description

In this section, we describe the system used in our submission. We also present approaches that were considered but not kept in our final submission.

Models

We experimented with multiple transformer-based models to tackle persuasion detection in the textual content of the memes.

- **BERT** [33]: First introduced in 2018, this model is based on the bidirectional transformer encoder architecture [144] trained with masked language model and next sentence prediction tasks.

- **BERT-HarMe**[9]: This model is a fine-tuned version of BERT on multiple datasets[10] [70, 134] about harmful/hateful speech in memes.
- **RoBERTa** [83]: This model changes the BERT pre-training approach, making it more robust.
- **AlBERT** [74]: AlBERT focuses on reducing the number of parameters of BERT to increase the training speed and lower memory requirements.
- **DistilBERT** [124]: This model uses knowledge distillation during pre-training to reduce the size of BERT.
- **DeBERTa** [59]: DeBERTa improves on BERT and RoBERTa by introducing a disentangled attention mechanism and an enhanced mask decoder.

Datasets

In this task, we use multiple training datasets. We experimented adding the train, validation and dev sets from SemEval-2021 Task 6 [35] and the PTC corpus [26] to the training data. Table 3.2 shows the datasets and their respective sizes.

- **SemEval-2021 Task 6**: This dataset also annotates memes with regard to the same 20 persuasion techniques. The train, validation and dev sets are appended to the training set of this task without any modification.
- **PTC Corpus**: This dataset contains news articles annotated at the span level with regard to 18 propaganda techniques. We first split the articles into sentences and transfer the span-level label to sentence-level. In this dataset, some labels are the same as this year's task, and can be aligned in a straightforward manner. However, when propaganda labels are different, they often correspond to multiple persuasion techniques. To align these labels, we add all the corresponding persuasion techniques valid for the propaganda. We only appended sentences that contain a propaganda technique to the training set of this task (around 5% of the total number of sentences).

Outputting ancestor classes

In this task, the goal is to detect the 20 persuasion techniques, but they appear in a hierarchical framework (see Figure 2.1. The official metrics of the challenge are hierarchical F1 (**F1H**), hierarchical precision (**PreH**) and hierarchical recall (**RecH**), which all take into consideration

[9] https://huggingface.co/limjiayi/bert-hateful-memes-expanded
[10] https://github.com/di-dimitrov/harmeme

Dataset	Size
SemEval-2024 Train	7000
SemEval-2021 Train+Validation+Dev	951
PTC (sampled)	427

Table 3.2: Datasets considered for training our models.

the hierarchical nature of the data. Since ancestor nodes are inherently outputted when detecting child nodes, we also tried to directly detect the ancestor classes. This raises the number of classes to 28 (instead of 20). Thus, the ancestor node can still be outputted even if its child node has not been detected, resulting in better performing models.

Losses

We also experimented with different training losses, which address multiple aspects of the data. For example, balancing the class misrepresentation in the data with class weights, or using hierarchical loss to reflect the hierarchical nature of the data.

- **Binary Cross Entropy (BCE) Loss**: This loss computes BCE losses for each class, weighted with the inverse frequency of its label, and sum them. This loss requires the output layer to have the size of number of classes.
- **Cross Entropy (CE) Loss**: We used 20 different CE losses for each class, weighted according to the inverse frequency of each label. Each loss computes the performance of the model at detecting a specific class. The final loss is the sum of the 20 losses. This loss requires the output layer to have twice the size of number of classes.
- **Focal Loss (FL)** [77]: This loss addresses class imbalance by down-weighting the loss assigned to well-classified examples. We used the implementation proposed by [39]. This loss requires the output layer to have the size of number of classes.
- **Custom Hierarchical Loss (HL)**: In order to reflect the hierarchical nature of the data, we implemented a custom hierarchical loss function. This function uses max pooling on logits x^c from children classes of the same ancestor a (e.g. Name Calling, Doubt, Smears, Reductio ad Hitlerum and Whataboutism are all children of the Ad hominem ancestor). The newly created logit correspond to the output of the model on the corresponding ancestor. Thus, we can compute the BCE Loss between this output and the true label y^a of the ancestor. We can iterate by max-pooling all the logits in the next ancestor. Note that logits can correspond to children or ancestor classes (e.g. the Logos ancestor pools the logits of Justification, Repetition, Intentional Vagueness, and Reasoning, even though the logits of Justification and Reasoning are also pooled from other child classes). We then sum all these BCE losses together, which measure how

well the model performs to detect the ancestor, rather than each persuasion techniques. Before summing this loss to the original classification loss of the techniques (CE, BCE or FL), we apply a normalization factor α. In practice, we found best results when α is equal to 0.5. Equations 3.1 and 3.2 describe the computation of this loss. $\mathscr{A}$ describes the ensemble of all ancestor techniques.

$$\mathscr{L}_{HL} = \mathscr{L}_{CE,BCE} + \alpha \cdot \sum_{a \in \mathscr{A}} \mathscr{L}^{a}_{BCE} \tag{3.1}$$

$$\mathscr{L}^{a}_{BCE} = y^{a} \cdot log\sigma(\max(\{x^{c}\}_{c \in child(a)})) + (1 - y^{a}) \cdot log(1 - \sigma(\max(\{x^{c}\}_{c \in child(a)}))) \tag{3.2}$$

Data augmentation

Some persuasion techniques have very little training data available in the datasets. We tried generating new samples for the bottom 5 classes with different methods.

- **Round Translation**: We translated every sample in French and translated them back to English. This can generate new sentences similar to the original ones. However, this new data is very limited and will not be varied.
- **GPT-4-Turbo Generation** [42]: We used GPT-4-Turbo to generate completely new sentences corresponding to a persuasive technique. As showed in [105], definitions of the class label have a significant impact in the performance of GPT models. We provided the definition of the persuasive technique provided by the organizers[11] in the system prompt, along with 5 randomly selected samples. We then used few-shot prompt technique with 5 more randomly selected samples, and finally asked the model to generate a new sentence. We generated two sets of 30 and 50 examples for five classes. For reproducibility measures, the full prompt is available in Appendix A.3.

Training process

For training our models, we use the AdamW [84] optimizer with a learning rate of 2e-5, and a weight decay of 0.01. We also use a ReduceLROnPlateau Learning rate scheduler, reducing the earning rate by a factor of 0.7 if results have not improved in 4 epochs. Most experiments are done on 10 epochs, saving the best model (according to F1H) on the validation set. We also

[11] https://propaganda.math.unipd.it/semeval2024task4/definitions.html

Model	Data	Classes	Loss	F1H	PreH	RecH
BERT	2024	20	CE	0.612	0.603	0.621
BERT	2024+2021	20	BCE	0.623	0.561	**0.700**
BERT	2024+2021	28	HL	**0.640**	0.626	0.654
BERT	2024+2021+PTC	28	HL	0.633	0.647	0.618
BERT	2024+2021	28	FL	0.629	0.638	0.620
BERT	2024+2021	20	FL	0.611	0.635	0.588
BERT	2024	28	CE	0.629	0.612	0.646
RoBERTa	2024+2021	20	CE	0.619	0.610	0.628
RoBERTa	2024+2021	28	CE	0.631	0.610	0.653
BERT-HarMe	2024+2021	20	CE	0.625	0.599	0.652
BERT-HarMe	2024+2021	28	CE	0.639	**0.651**	0.627
BERT-HarMe	2024+2021	28	HL	0.634	0.634	0.634
BERT-HarMe	2024+GPT-augmented	28	CE	0.634	0.605	0.666
AlBERT	2024+2021	20	CE	0.604	0.600	0.607
DeBERTa	2024+2021	20	CE	0.617	0.617	0.618
DistilBERT	2024+2021	20	CE	0.602	0.622	0.584
Ensembling	Top-3 best models			**0.675**	**0.650**	**0.702**

Table 3.3: Results on the dev set of some of the models we tried. Other models with different combination of parameters are used in the ensembling and not showed here due to space, but obtain similar performances.

experimented with freezing the first few layers of the pre-trained BERT-based model to keep its acquired knowledge when trained on massive amount of data.

Ensembling

We trained many models according to different combinations of the previous parameters. Our final submission consists of a majority voting among the top-3 models for each persuasion technique evaluated on the dev set and according to the F1-score. These models are not necessarily the best models overall according to hierarchical F1, but demonstrate effectiveness in detecting specific persuasion technique. We also perform majority-voting on ancestor classes with models that output them (Section 3.2.1).

3.2.2 Results

We share our results on the dev set provided by the organizers in Table 3.3. These results show the performance of some single models as well as the performance of the ensembling used in the final submission. Table 3.5 shows the performance of each class on the dev set, using the ensembling model for classification. Table 3.4 shows the results of our final submission on the 4 test languages: English, Bulgarian, North Macedonian and Arabic. We translate non-English

Language	F1H	PreH	RecH
English	0.655	0.628	0.685
Bulgarian	0.345	0.367	0.325
North Macedonian	0.442	0.520	0.384
Arabic	0.177	0.343	0.119
Arabic (unofficial)	0.439	0.369	0.544

Table 3.4: Results on the test set with our ensembling model, translating non-English languages to English.

languages using `py-googletrans`[12] to English in order to run our models and obtain the predictions. We would like to note that our official submission for the Arabic language was incorrect, due to Arabic-to-English translation errors on our end. We corrected the error and also show the performance of the model, albeit being an unofficial result.

3.2.3 Discussion

Model-wise, our best results were obtained using BERT, RoBERTa and BERT-HarMe. We ultimately did not use any of AlBERT, DeBERTa and DistilBERT models in our final submission as those were not in any top-3 best performing models of any persuasion techniques. The BERT-HarMe models were the best-performing on the detection of 'Slogans', 'Appeal to Authority', 'Flag-waving', 'Appeal to fear/prejudice', 'Black-and-white Fallacy/Dictatorship', 'Thought-terminating cliché', 'Presenting Irrelevant Data (Red Herring)', 'Glittering generalities (Virtue)', 'Doubt', 'Logos', 'Justification' and 'Distraction' classes. RoBERTa models were the best-performing for 'Repetition', 'Bandwagon', 'Ethos'.

We also noticed a slight performance increase by adding the 2021 dataset during training, which was not necessarily true when adding the PTC corpus. This is probably due to the fact that the PTC Corpus is about news articles and not memes. Our data-augmentation experiments on round-translation did not improve the results at all, while the GPT-4-Turbo augmentation experiments provided a very slight boost, but not for the augmented classes.

The hierarchical nature of the task and the evaluation metrics were reflected in the results, as most of our best performing models are outputting 28 classes by including the ancestors and/or are trained with Hierarchical Loss (**HL**). However, best models at detecting 'Causal-Oversimplification' are using BCE Loss.

We can see in Table 3.2 that some persuasive techniques are easier to detect than others. For example, 'Appeal to authority' seems to be the easiest class to detect, and 'Obfuscation, Intentional vagueness, Confusion' the hardest. Training data seems to lightly correlate with

[12] https://github.com/ssut/py-googletrans

Technique	F1H
Repetition	0.516
Obfuscation	0.000
Slogans	0.495
Bandwagon	0.583
Appeal to authority	0.891
Flag-waving	0.623
Appeal to fear/prejudice	0.425
Causal Oversimplification	0.304
Black-and-white Fallacy	0.549
Thought-terminating cliché	0.330
Straw Man	0.286
Red Herring	0.182
Whataboutism	0.442
Glittering generalities (Virtue)	0.562
Doubt	0.437
Name calling/Labeling	0.617
Smears	0.583
Reductio ad hitlerum	0.526
Exaggeration/Minimisation	0.492
Loaded Language	0.682
Logos	0.773
Reasoning	0.552
Justification	0.727
Simplification	0.496
Distraction	0.389
Ethos	0.810
Ad Hominem	0.742
Pathos	0.704

Table 3.5: Results of our ensembling model on the dev set, per-class.

performance results, with some strong outliers like 'Smears' under-performing comparing to its high number of training samples, and 'Bandwagon' over-performing. As for the ancestor classes, the highest-level 'Logos', 'Ethos' and 'Pathos' have the highest performance, while those composed of the hardest persuasive techniques to detect like 'Simplification', 'Distraction' and 'Reasoning' have lower performance.

3.3 Conclusion

In this chapter, we tackle the detection of misinformation-related factors. This work addresses RQ2 by enabling the detection and the analysis of conspiracy theories and persuasion techniques.

We propose a transformer-based method to detect COVID-19-related conspiracy theories in tweets, composed of an ensembling of CT-BERT models. We also propose a node embedding-based techniques to detect misinformation spreader in the user-interaction graph, using node2vec and an MLP classification head. Our best model obtains a MCC score of 0.719 on the test data.

We also describe the system our team EURECOM used for sub-task 1 at SemEval-2024 Task 4: Multilingual Detection of Persuasion Techniques in Memes. We explore multiple BERT-based models, training datasets, losses, data augmentation procedures, and training process. Our final submission consists of an ensembling model that performs majority voting between our top-3 best performing models for each persuasive technique. We find that some pre-trained models on harmful meme data are competitive, and that incorporating hierarchical information in the training process, such as outputting the whole 28 classes (including the ancestors) or using a hierarchical loss, significantly improves the results. We obtain a hierarchical F1 score of 0.675 on the dev set and 0.655 (English), 0.345 (Bulgarian), 0.442 (North Macedonian), 0.177 (Arabic) on the test set.

As factors play a significant role in the spread of misinformation, we propose models to detect them, answering RQ2. Future work could focus on finding better usage of the user-graph in the detection of conspiracy theories, and detect other misinformation-focused factors.

Chapter 4

Definitions Matters

In this chapter, we explore the ability of LLMs at detecting textual features. We again analyze the tasks of Conspiracy Theory and Persuasion Techniques classification. We also explore how class definitions impact the classification results, giving insights about how LLMs 'understand' prompts. Lastly, we show a method to generate class definitions based on examples, and we find that improving definitions of the class labels has a direct consequence on the downstream classification results. The work of this chapter has been published in [105] and on GitHub[1].

4.1 Methodology

We leverage GPT-3 to perform multi-label zero-shot conspiracy theory detection on the test set of the MediaEval data (see Section 3.1). In particular, we perform binary classification for each conspiracy category, labeling each tweet as either mentioning the conspiracy or not.

Our baseline method relies on zero-shot (ZS) conspiracy theory classification from the textual label of the classes only (e.g. 'Anti-vaccination', 'Harmful Radiation', 'Satanism', etc). This assesses if the knowledge encoded in GPT-3 is able to differentiate between similar conspiracy theories.

Our next two approaches aim at improving the model's understanding of the label by providing more context in the prompt, specifically with a short definition of the label. We compare two types of definitions: Human-Written (HW) and Example-Generated (EG).

The HW definitions are given in the dataset overview paper [75], and are part of the guidelines that were given to the human annotators of the data. Despite the definitions being well-written, annotators had to regularly discuss their understanding of the categories, suggesting the difficulty of the task at hand[2].

[1] https://github.com/dkorenci/gpt-def-zeroshot

[2] The authors report a 92% inter-annotator agreement and more than half tweets had at least one disagreement.

The EG definitions are generated with GPT-3 from the training set, by providing GPT-3 with 25 examples of tweets mentioning a given conspiracy theory and 25 examples of tweets not related to the conspiracy theory. We use 5 different random seeds to randomly select the example tweets[3], resulting in 45 definitions generated in total. We then ask the model to come up with a short textual description that could separate the sets of tweets. In this setting, we do not provide the textual label of the conspiracy theory, but we only give example tweets to the model. This prevents the model to rely on some of its pre-trained knowledge from reading the textual label. Examples of definitions which have been generated are in Appendix A.2.1.

For prompting the model, we rely on simple prompts, using both OpenAI's 'system' and 'user' roles in our request. The 'system' message contains a description of the task, while the 'user' message contains the tweet's content to be classified. For the classification of the tweets, the definition is appended at the end of the 'system' message. Example prompts used to generate EG definitions and to annotate conspiracy theories are provided in Appendix A.2.2.

4.2 Definition Understanding

The approach of definition-based zero-shot classification leads to whether GPT-3 is able to correctly "interpret" definitions and "apply" them to text classification, which, in our case, amounts to detection of conspiracy categories in texts. We propose two tests aimed at assessing if GPT-3 indeed "understands" the definitions given in the prompts.

The general approach is to use semantic similarity to measure how similarity between definitions correlates with the output of the definition-based classifiers, which we view as a result of GPT-3's "interpretation" and "application" of a definition. For example, one expectation is that similar definitions should lead to similar outputs. We perform the tests using the 45 example-generated definitions, which represent a challenging test case of mutually close definitions – randomly varied and derived from related categories. We define the semantic similarity of two definitions as cosine similarity of their embeddings, using state-of-art[4] sentence embedding model [119].

The first test of GPT-3's "understanding" of the definitions measures whether EG definitions more similar to HW ones guide the model to produce better classification results. This is achieved by correlating the similarity between the EG definitions and the corresponding HW ones, and the performance of the classifiers based on the generated definitions.

The second test measures whether mutually similar EG definitions guide the model to produce similar predictions. This is achieved by correlating the similarity between two EG definitions on one side, and the similarity of the corresponding classifiers' predictions on the other

[3] Tweets in both sets can also support other conspiracy theories (multi-label classification problem)

[4] We use top-ranked `all-mpnet-base-v2` model: https://www.sbert.net/docs/pretrained_models.html

side. Similarity between two sets of predicted binary labels is calculated using Cohen's κ, a chance-corrected measure of annotator agreement.

4.3 Results

4.3.1 Conspiracy Theory Classification

In this section, we discuss the results of the different approaches on the classification of the full test set, totaling 823 tweets. Average results are in Table 4.1, and per-category results are in Figure 4.1. We use Matthews correlation coefficient, Precision, Recall and F1 score to compute the classification performance.

Approach	MCC	Precision	Recall	F1
Zero-shot	0.398	0.331	**0.852**	0.440
w/ Example-generated definitions	0.442	0.371	0.831	0.485
w/ Human-written definitions	**0.516**	**0.464**	0.823	**0.555**
CT-BERT ensembling	0.780	0.779	0.849	0.810

Table 4.1: Performance of the LLM and transformer models using macro-averaging.

Results show that both EG and HW definitions outperform the ZS baseline. It supports the claim that GPT-3 is capable of leveraging the knowledge provided via the definitions to perform classification and, therefore, that definitions of the labels can be used to guide the model to better perform NLP tasks. While EG definitions do not reach the same performance as HW ones, they can still be used to significantly improve classification accuracy, especially in cases where the HW definition is not available. Our method shows that we can infer a textual description from examples and that GPT-3 can use it to better annotate future samples. Indeed, the usage of EG definitions leads to an average relative gain of around +10% in MCC, Precision and F1 scores compared to the ZS baseline. HW definitions see an even greater improvement of around +30% in average, showing the importance of a well-defined definition. However, these results are still far from the state-of-the-art CT-BERT fine-tuning methods.

Figure 4.1 reports the performances for all approaches per conspiracy theory. We observe a general trend, with definitions having a positive impact on the performance for most conspiracy theories. However, some concepts are seemingly harder for GPT-3 to produce useful definitions, such as Satanism, where the EG definitions lead to worse results than the ZS baseline. Also, some conspiracies are more robust to the EG definitions, as the variance is low and changing the samples lead to similar results, such as Intentional Pandemic, or Fake Virus. Lastly, some EG definitions lead to better results than the HW ones, suggesting that with a better sampling of the examples, this method could generate higher-performing definitions.

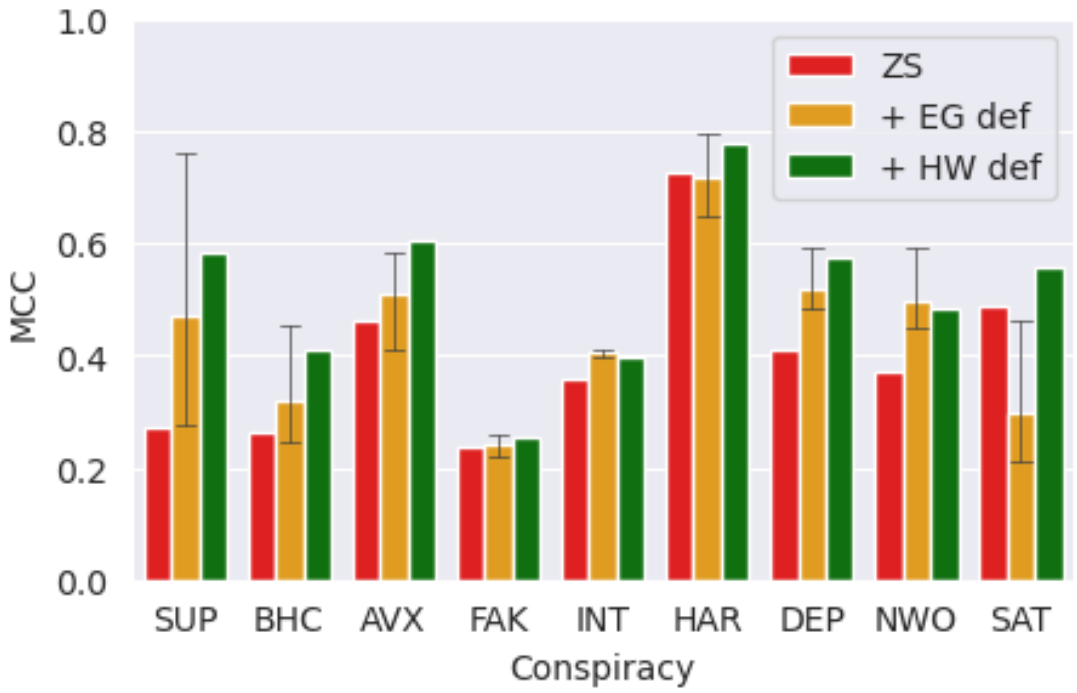


Figure 4.1: MCC score on the test set. Error bars show the minimum and maximum values (5 random seeds)

4.3.2 Definition Understanding Tests

The Spearman's rank correlation coefficients between semantic similarity of the definitions and the results of the definition-based zero-shot classifiers are shared in Table 4.2. The strength of the correlations is fair, which supports the claim that GPT-3 is able to correctly interpret the definitions and apply them to conspiracy detection. Namely, higher similarity between EG and HW definitions leads to more accurate classifications, which suggest that the model can translate better definitions into better predictions. Additionally, higher similarity between two EG definitions correlates with higher agreement between their corresponding predictions, which suggest that the model translates similar definitions into similar predictions.

An interesting question that stems from the variation of the definitions is whether the performance increase is a result of the quality or the quantity of information in the definitions. To address this question, we correlated the length of the 45 EG definitions measured by the number of tokens with their classification performance measured by MCC. We found a lack of correlation – a very small ρ of 0.062. We take this as evidence supporting the claim that the performance depends on the quality, and not on the quantity, of information in a definition.

	MCC	F1
Similarity (EG, HW)	0.375	0.390
	Cohen's κ	
Similarity (EG, EG)	0.407	

Table 4.2: Results of the two definition understanding tests based on semantic similarity and classification results. Top row contains Spearman's correlations of similarity between EG and HW definitions, and performance of EG zero-shot classifiers. The bottom row contains correlations of similarity between pairs of EG definitions, and Cohen's kappa of their classification.

4.4 Additional experiments

In this section, we conduct additional experiments by changing the models to (i) generate the definition and (ii) annotate the data. We are also applying our method on a different task, the detection of persuasion techniques in text.

Use of open-access models The Zephyr-7B-β model [142] is based on the Mistral-7B [66] language model, trained using DPO [117]. It replaces humans-in-the-loop in the training with powerful LLMs for data generation and ranking of preferred responses. We use this model to perform the same experiments presented in Section 4.1, i.e. to annotate conspiracy theories in tweets, and to generate definitions.

We also use the Llama-2-13B model [139] to generate definitions and annotate data. This model released by Meta is a popular choice because of its rich performance in many domains while still being open-source.

Few-shot learning Since we need examples to generate definitions, it is fair to compare the results with "few-shot" methods. In this set of additional experiments, we use examples in the prompt to analyze differences in the results with example-generated definitions.

Propaganda technique classification Lastly, we experiment with another task, the detection of propaganda techniques in text. We use a slightly modified version of the Propaganda corpus defined in section 2.3.2. We only annotate propagandist sentences, and only keep the most used propaganda techniques[5]. This results in a smaller subset of 2,000 sentences.

[5]Name Calling/Labeling, Repetition, Slogans, Appeal to fear/prejudice, Doubt, Exaggeration/Minimisation, Flag-waving, Loaded Language, Oversimplification, Appeal to authority, Black-and-white Fallacy/Dictatorship

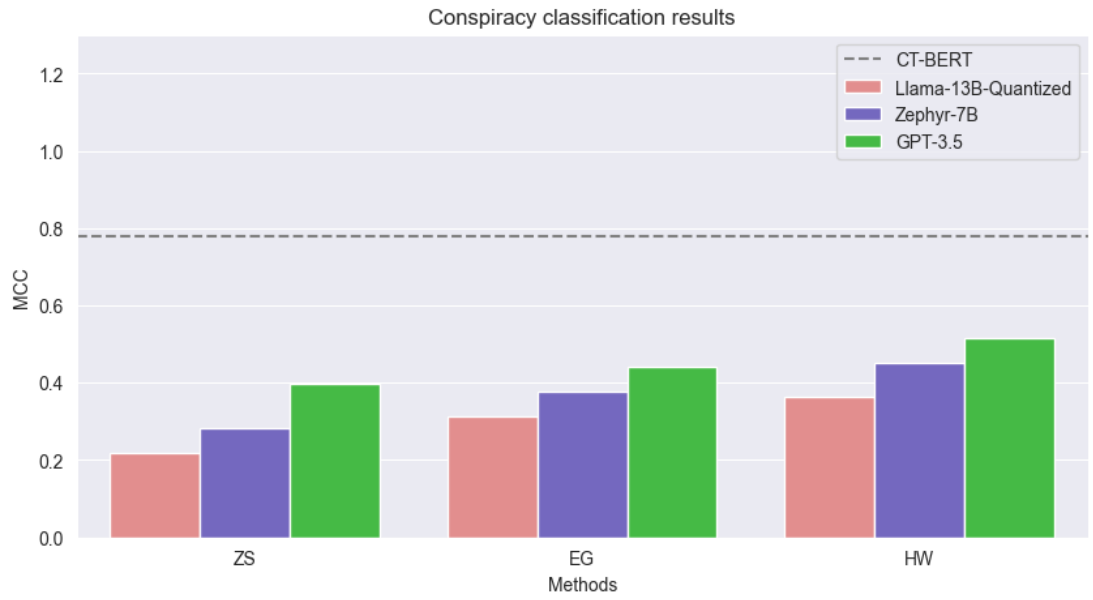


Figure 4.2: Average MCC for different LLMs on different definitions settings

4.4.1 Results

We can see in Figure 4.2 the average MCC for the Llama-2, Zephyr and GPT-3.5 models. As we can see, GPT performs best and Llama worse in all settings. Interestingly, Zephyr with high quality definitions such as the Human-written performs better than GPT-3.5 zero-shot. Also, for all the models, zero-shot classification performs worse than using example-generated definitions that in turn performs worse than human-written definitions. This further proves the importance of high-quality class definitions for classification using LLMs.

Figure 4.3 shares per-class results for multiple classification settings using the Zephyr-β model. We compare zero-shot, with multiple few-shot settings and multiple definition of the classes. Our first few-shot settings (FS1) consist of adding in the prompt 10 positive and 10 negatives examples, in order. The examples are the same for all the tweets to annotate. The second few-shot settings (FS2) are the same 20 examples, with their order randomized. Our last few-shot settings (FS3) consist of randomly selecting 20 example for each new sample. Figure 4.4 show the same results averaged on the conspiracies.

We see that few-shot experiments do not perform better than zero-shot, which explain the difficulty of experimenting with this technique. The number of examples, their order and the choice of the sample can all impact the performance of few-shot, and are not trivial to set. We see FS1 performing worse than all other techniques on average, this may be because the model learns the pattern of the examples (10 positive, followed by 10 negatives), which may introduce a bias. On the other hand, we see that adding a definition has on average a positive

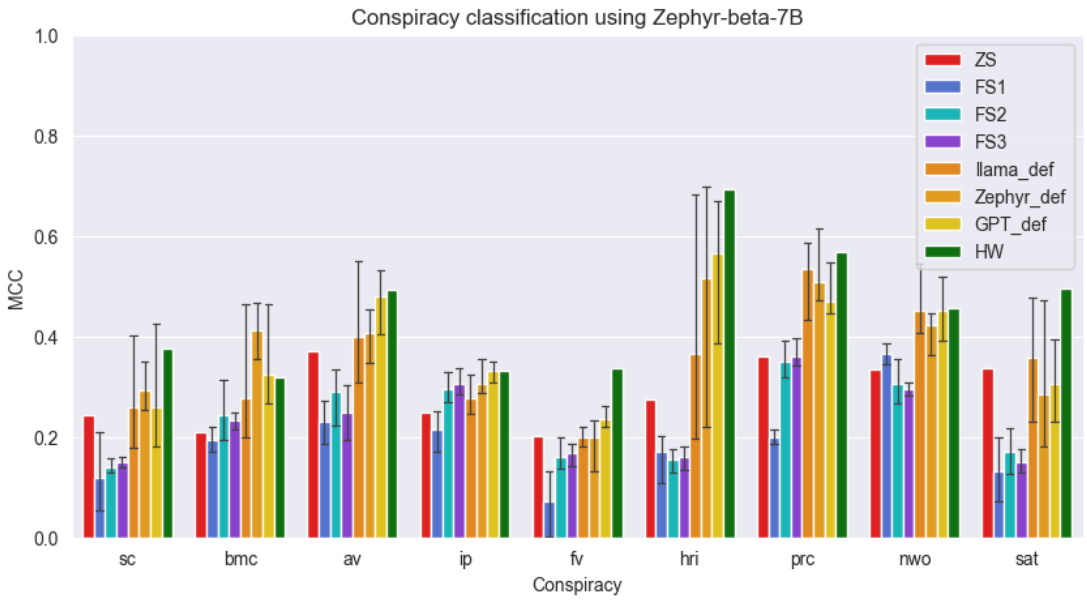


Figure 4.3: Classifications results of Zephyr-β using different definitions for each conspiracy theory

results on the MCC. While those written by experts generally perform better, we can also see that a better model will tend to generate better definitions, and those definitions could be used with a smaller model. Indeed, we can see that GPT-3.5 generates the best definitions, even if Zephyr is used for annotation. While calls to the OpenAi API can be expensive, the cost of generating definitions is fairly low (1 call per class), and the definitions can then be used with cheaper models to improve classification results with smaller budget.

In Figure 4.5, we can see the results of propaganda technique classification using GPT-3.5. Zero-shot performs almost as good as with definitions, which could be explained as the model already knowing the definitions. It is very possible that the model was trained on this exact dataset, since it was released before the training of GPT-3.5. If the model has been trained on this data, it already has seen the definitions, and 'memorized' them. Also, we can see that the results are very poor, with most classes below 0.2 F1 Score.

4.5 Conclusion

In this chapter, we analyze the impact of label definitions on the performance of GPT-3 zero-shot classification, on a challenging task of fine-grained conspiracy theory detection. This research helps understand how LLMs process information, as well as detect textual features, thus addressing RQ2.

We show that the use of better definitions leads to a significant gain in most classification

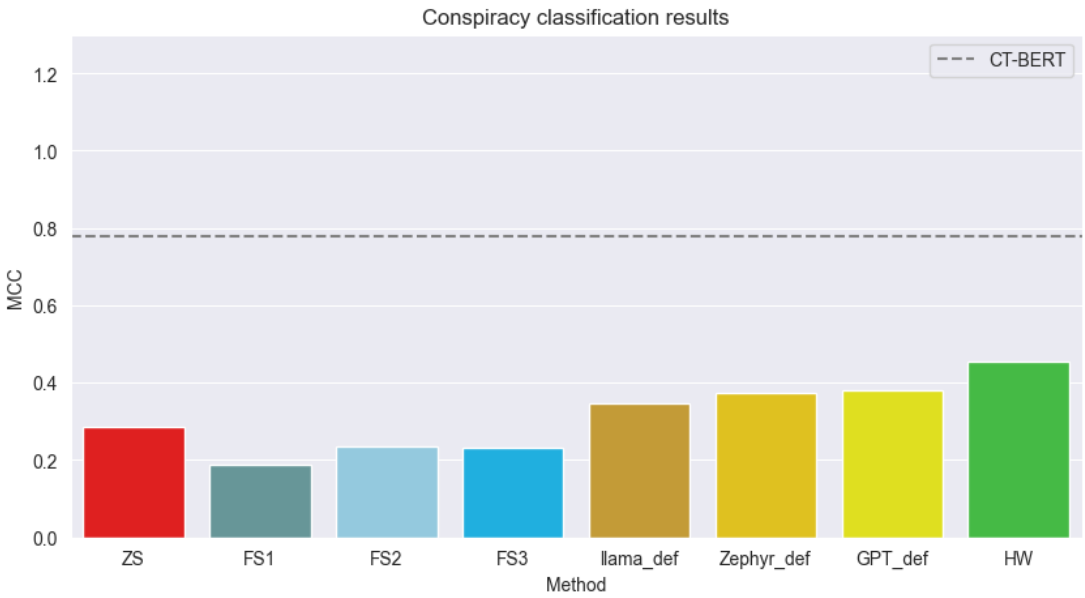


Figure 4.4: Average MCC for Zephyr-β using different classification settings

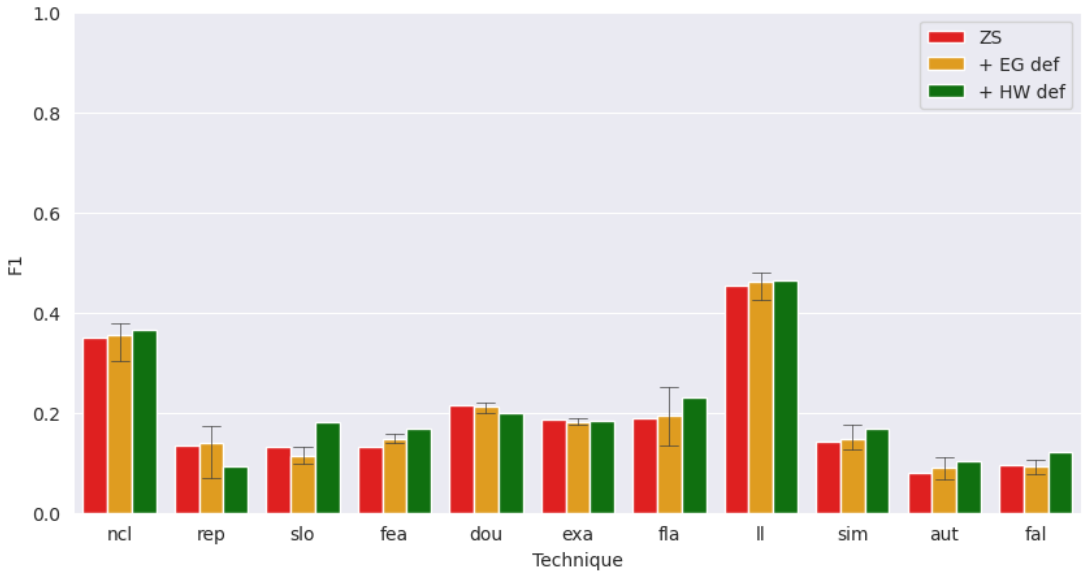


Figure 4.5: F1 score for propaganda classification using GPT-3.5-turbo

metrics (MCC, Precision, F1). We also demonstrate an approach of generating definitions from examples. Human-Written definitions still provide better results, while example-generated definitions show promising performance. Additionally, we successfully tested GPT-3's ability to understand and apply these definitions for classification. We also experiment with other open LLMs, namely Zephyr-β and Llama-2, to generate definitions and annotate tweets. We show that powerful models can be used to generate high quality definitions, and benefit lower-cost models for significant classification accuracy gains. Future work could focus on exploring more diverse tasks and models, as they can lead to better understanding LLMs.

Chapter 5

Automatic Detection of Factors in Social Media Posts

This chapter addresses the detection of textual features in social media posts, in particular with the use of transformer-based models. We refer to *factors* as textual features that allow better understanding of textual content. They represent dimensions that impact the way we understand online content, through emotions, sentiment, political-bias, persuasion techniques, conspiracy theories, tropes, etc. In the following sections, we describe our approach for creating models that detect such *factors* in Section 5.1. Additionally, we explore correlations between the aforementioned factors, revealing useful insights. In section 5.2, we propose a dataset annotating a novel *factor* representing easily recognizable devices used in narratives to convey a specific theme or idea, called Tropes. We release an annotated dataset and propose approaches to detect Tropes on social media. Finally, we compare tropes to other factors such as conspiracy theories and persuasion techniques. Both sections of this chapter have been published in [107] and [44] respectively, while the code is available on GitHub[1][2].

5.1 Detecting Emotion, Sentiment, Political Leaning

As the amount of information shared online increase[3], we are prone to face more misinformation on the web. Events such as the 2016 U.S. Presidential Elections [4] or the Brexit [58] are prime examples of strongly discussed topics with large amount of false information shared online. Social media websites can have strong influence on shaping the beliefs of one individual, and can have consequences on real life topics such as politics [4, 58], science [141], economics[4] or health [89]. Considering that 'fake-news' tends to spread faster and wider than the truth [146], researchers have started to help fact-checkers scale up their ability to verify information [97]. The need for such technology has been even more evident with the

[1] https://github.com/D2KLab/covid-twitter-discourse-analysis
[2] https://github.com/Tireswind/ADTIST24
[3] https://www.domo.com/learn/infographic/data-never-sleeps-5
[4] https://www.forbes.com/sites/kenrapoza/2017/02/26/can-fake-news-impact-the-stock-market/

recent COVID-19 pandemic, with misinformation shared profusely online, and the World Health Organization (WHO) describing it as an *infodemic*[5]. According to [18], the number of fact-check reports rose by more than 900% between January and March 2020, reflecting the large amount of misinformation shared about COVID-19.

While some research is focused on *detecting misinformation*, in this work, we focus on better understanding the online discourse around COVID-19 on dimensions that go beyond misinformation classification. We explore the relationships among emotion, sentiment, political bias, stance, veracity and conspiracy theories, by leveraging a dataset for each textual feature. We use three datasets for training models that detect sentiment, emotion and political bias, and we use those models on the other datasets to study in detail their interactions. We then compute the conditional distribution of the labels between those features to analyze and share some insights about their relationships. Notable results show that political bias plays a role in the stance toward COVID-19 regulations and conspiracy theories, or that emotion and sentiment are used by people who share potentially misleading content.

5.1.1 Related Work

Online misinformation during the COVID-19 pandemic has urged researchers to study its prevalence in social media websites, such as Twitter. Many datasets have been built around annotating textual features in tweets during the pandemic. In this work, we selected three different core datasets, each one allowing the training of a model for the detection of one textual feature: COVID LTSE Attributes (Emotion) [56], COVIDSenti (Sentiment) [99] and Russian Troll (Political Bias)[6] [79]. We also selected three additional external datasets, which will only be used for the evaluation of the correlation: COVID-19 Stance (Stance) [50], Birdwatch (Veracity) [123] and MediaEval-FND (Conspiracy theories) [111]. The three core datasets are also used for the evaluation of the correlation. All datasets are described in Section 2.3.2.

Classification Models

Transformer-based models [144] have largely contributed to progress in many Natural Language Processing (NLP) tasks, including machine translation [29, 131], question answering [85, 154] and text classification [41, 92]. Most notably, BERT [33] has outperformed other methods, such as TF-IDF or Recurrent Neural Networks, while providing a pre-trained model that can be fine-tuned for specific tasks [133]. For example, Covid-Twitter-BERT (CT-BERT) [95] has been trained on textual data from Twitter during the COVID-19 pandemic, which improves results on domain-specific datasets.

[5] https://www.who.int/health-topics/infodemic
[6] https://fivethirtyeight.com/features/why-were-sharing-3-million-russian-troll-tweets/

5.1.2 Methodology

In order to detect correlations, we build three text-classification models on the core datasets for sentiment, emotion and political bias. In this section, we will discuss the methodology to train the different models on their respective data, and explain how we analyze potential correlations between the textual features.

Model Training

Models are trained to perform text classification using supervised learning. As some datasets are very large, we sample 25,000 tweets from the COVID LTSE Attributes dataset and 42,000 from the Russian Troll dataset. We first apply some basic pre-processing on the text, by removing links and special characters. We then split datasets into a train set and a validation set with a 80/20 stratified split ratio. Models have pre-trained CT-BERT weights and a classification layer depending on the number of output classes. They are trained using Adam [72] optimizer, with a weight decay of 1.10^{-2} and a learning rate of 1.10^{-5} for 25 epochs. We use a Cross Entropy loss with weights proportional to the inverse of the class distribution. We monitor the performance of the model with the F1 score on the validation set, and save the best performing model. We use the PyTorch [102] and huggingface-transformers [152] libraries to implement our code. Results for the models during training are available in Section 5.1.3.

Correlation

In order to find possible correlations between the textual features, we used the different models to predict features on the other datasets. Core datasets are used for the training of the models and the prediction of other features, while external datasets are only used for prediction. For example, the model trained on COVIDSenti is used on the COVID LTSE Attributes dataset, the Russian Troll dataset, the COVDI-19-Stance dataset, the Birdwatch dataset and the MediaEval-FND dataset. This way, we can analyze the conditional distribution of the predicted labels given the ground truth labels. The results and analysis are presented in Section 5.1.3.

5.1.3 Results

In this section, we discuss the results of the models during the training (5.1.3), and the analysis of the possible correlations between the features (5.1.3). Some examples of tweets in the dataset are available in Table 5.2 and 5.3, highlighting a particular predicted label.

Table 5.1: F1-Score of the models on a validation set

Core Dataset	F1-score
COVID LTSE Attributes	0.622
COVIDSenti	0.769
Russian Troll	0.636

Model performance

As discussed in Section 5.1.2, we split all core datasets into training and validation sets with a 80/20 stratified split ratio. Table 5.1 shows the performance of the models on the validation set.

The model based on the COVIDSenti dataset obtains the best score on its own evaluation set. This might be expected, as sentiment detection is arguably the easiest task of the three. The overall performance of the models are fair, given the noise in the datasets, as COVID LTSE Attributes and COVIDSenti have been automatically annotated, and the Russian Troll dataset has been labelled at the user level, resulting in some generic tweets having annotations towards political bias.

Correlation Analysis

In order to detect some correlations between the studied textual features, we compute the matrix of frequency of the labels of two textual features in the data. The y-axis label[7] is the ground truth of the corresponding dataset, while the x-axis label represents the prediction of the model. Rows have been normalized to represent the conditional distribution of the predicted label given a ground-truth label.

Sentiment feature Figure 5.1 shows the correlation between the sentiment feature and the other features. We can see in Figure 5.1a 5.1b 5.1c 5.1d that people against the mentioned topics tend to share a more negative sentiment. This is especially true for the topic 'Face masks', and less apparent for the topic 'School closures'. People in favor of the 'Face masks' and 'School closures' topics tend to use more negative sentiment as well. However, people use more positive sentiment when supporting 'Stay at home' orders.

Figure 5.1e shows that all emotions except happiness tend to be more negative than positive, which is expected. It also shows that anger is the emotion where negative sentiment is the most prevalent. Figure 5.1f shows that tweets that have a political bias use more sentiment (positive and negative) than other tweets. However, the distribution of sentiment is the same

[7]The labels 'N', 'H', 'A', 'S', 'F' represent the following emotions: 'None', 'Happiness', 'Anger', 'Sadness', 'Fear'.

for both Left and Right bias.

Figure 5.1g shows that tweets that share potentially misleading content tend to use slightly more negative sentiment than the non-misleading tweets. Very few tweets share positive sentiment in this dataset overall, suggesting that people on Birdwatch are more interested in labeling negative tweets. However, conspiracy theories do not seem to be particularly correlated with sentiment, as shown in Figure 5.1h.

Emotion feature Figure 5.2 shows how the emotion feature is correlated to the other features. First, it seems clear in Figure 5.2a 5.2b 5.2c 5.2d that the four topics 'Fauci', 'Face masks', 'School closures' and 'Stay at home' are quite controversial on Twitter, with the anger emotion dominating almost all stances. The sadness emotion is the most used when discussing the topic of 'School closures', showing empathy for the teachers and the children. The 'Stay at home' topic sees more happiness in the tweets, with people enjoying working from home.

Figure 5.2e shows that a majority of tweets from the COVIDSenti dataset use the fear emotion, even in positive tweets. This seems counter-intuitive and may be due to having numerous tweets about wishing people to stay safe, in fear of covid. However, positive tweets also use happiness a lot, which is to be expected.

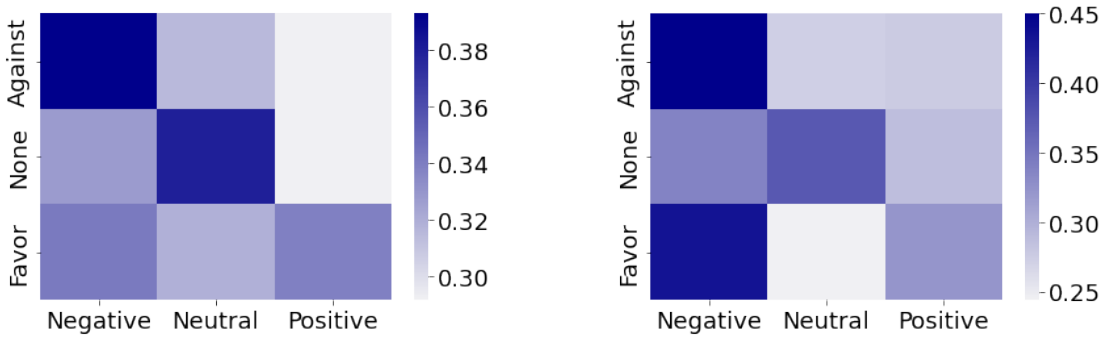
Political biased tweets are more likely to have an emotion than not, as shown in Figure 5.2f. Tweets from users tagged as having left bias tend to contain more happiness, while tweets from users having right bias tend to contain more anger.

Regarding veracity, in Figure 5.2g, anger is dominating the tweets sharing potentially misleading information, while emotions are slightly more even on not misleading tweets. In Figure 5.2h, we can see that emotion and conspiracy theories are not heavily correlated. We notice a slight decrease in anger in non-conspiracist tweets.

Political bias feature Lastly, we analyze correlation between political bias and other textual features, highlighted in Figure 5.3. We again notice that some topics are controversial, for example, 'Face masks' and 'Stay at home'. In those topics, we see that people against (face masks) have more right political bias and people in favor have more left political bias. This reflects the U.S. political landscape during the pandemic, as Democrats governors had generally more strict mandates towards wearing face masks than their Republican counterparts [48]. The topic of school closures and re-opening was also highly controversial, with Republicans leaning toward having more in-person classes and Democrats toward having more online-classes[8].

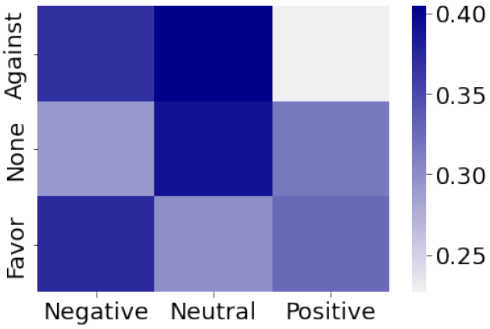
Figures 5.3e 5.3f show that specific sentiment and emotion are not strongly correlated to one

[8]https://www.pewresearch.org/fact-tank/2020/08/05/republicans-democrats-differ-over-factors-k-12-schools-should-consider-in

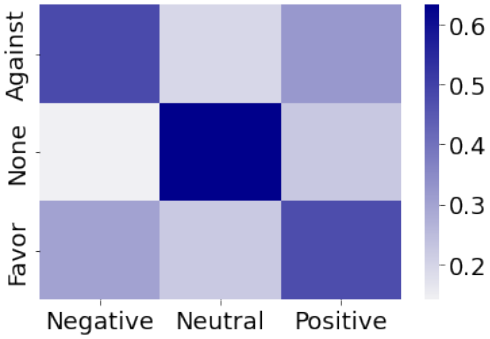


(a) Stance towards 'Fauci'

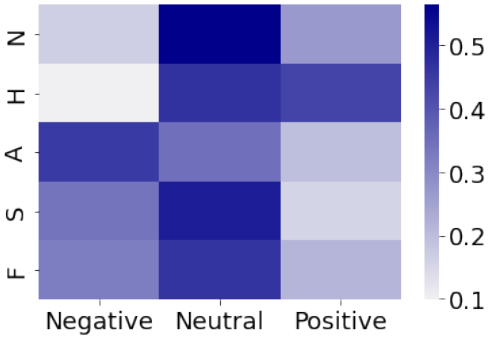
(b) Stance towards 'Face masks'



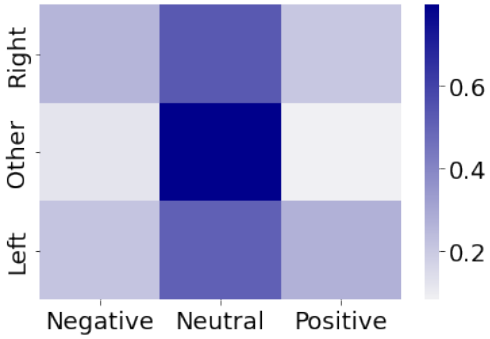
(c) Stance towards 'School closures'



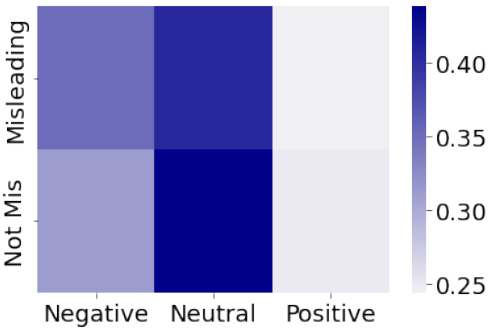
(d) Stance towards 'Stay at home'



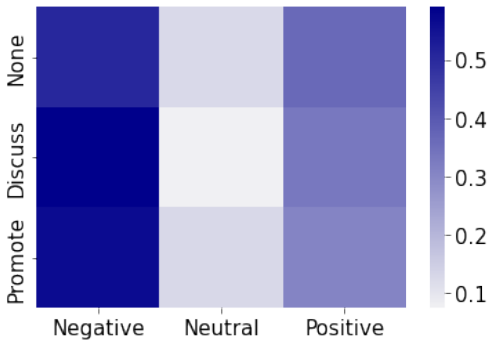
(e) Emotion



(f) Political bias

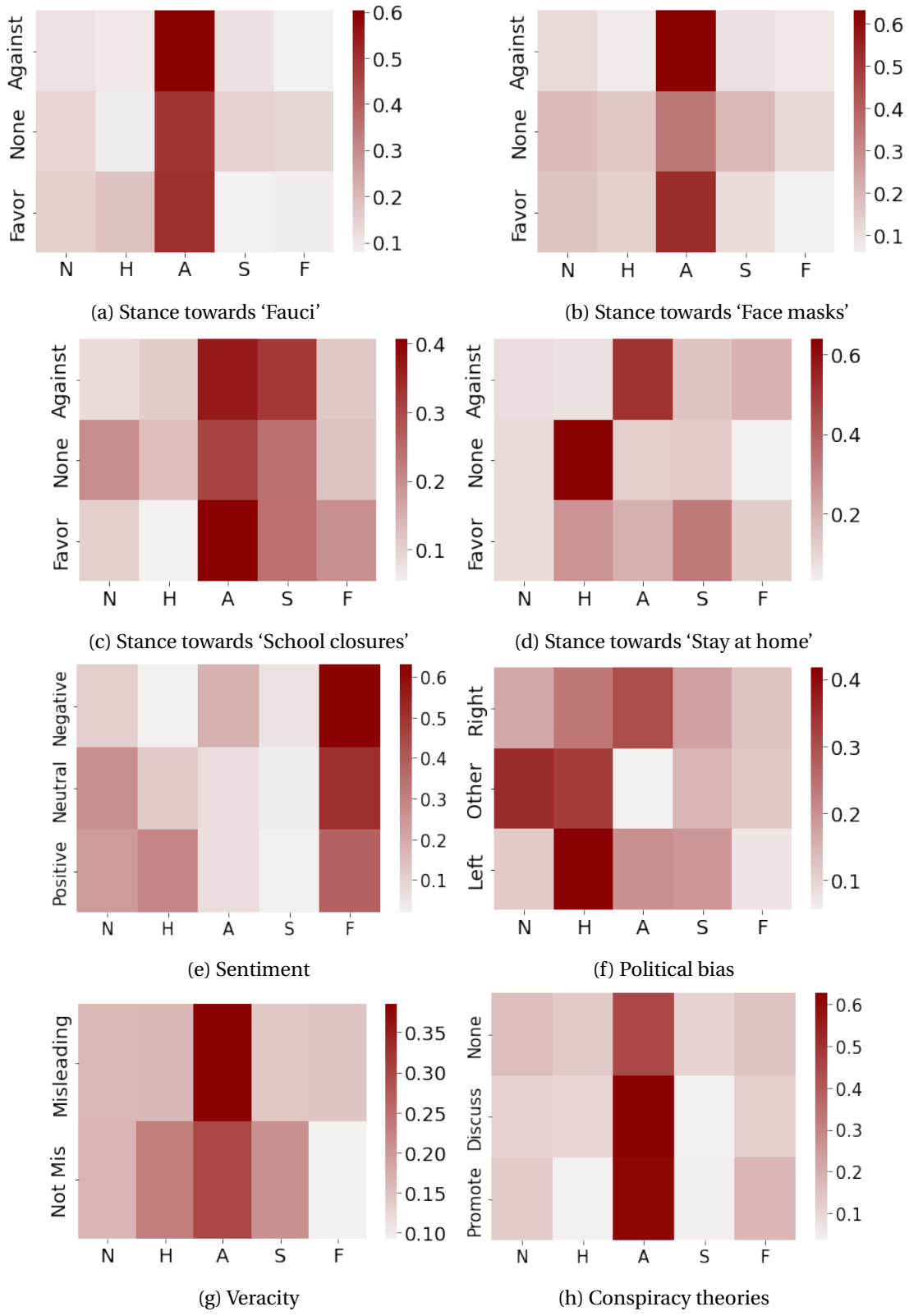


(g) Veracity



(h) Conspiracy theories

Figure 5.1: Distribution of the labels for the sentiment feature



(a) Stance towards 'Fauci'

(b) Stance towards 'Face masks'

(c) Stance towards 'School closures'

(d) Stance towards 'Stay at home'

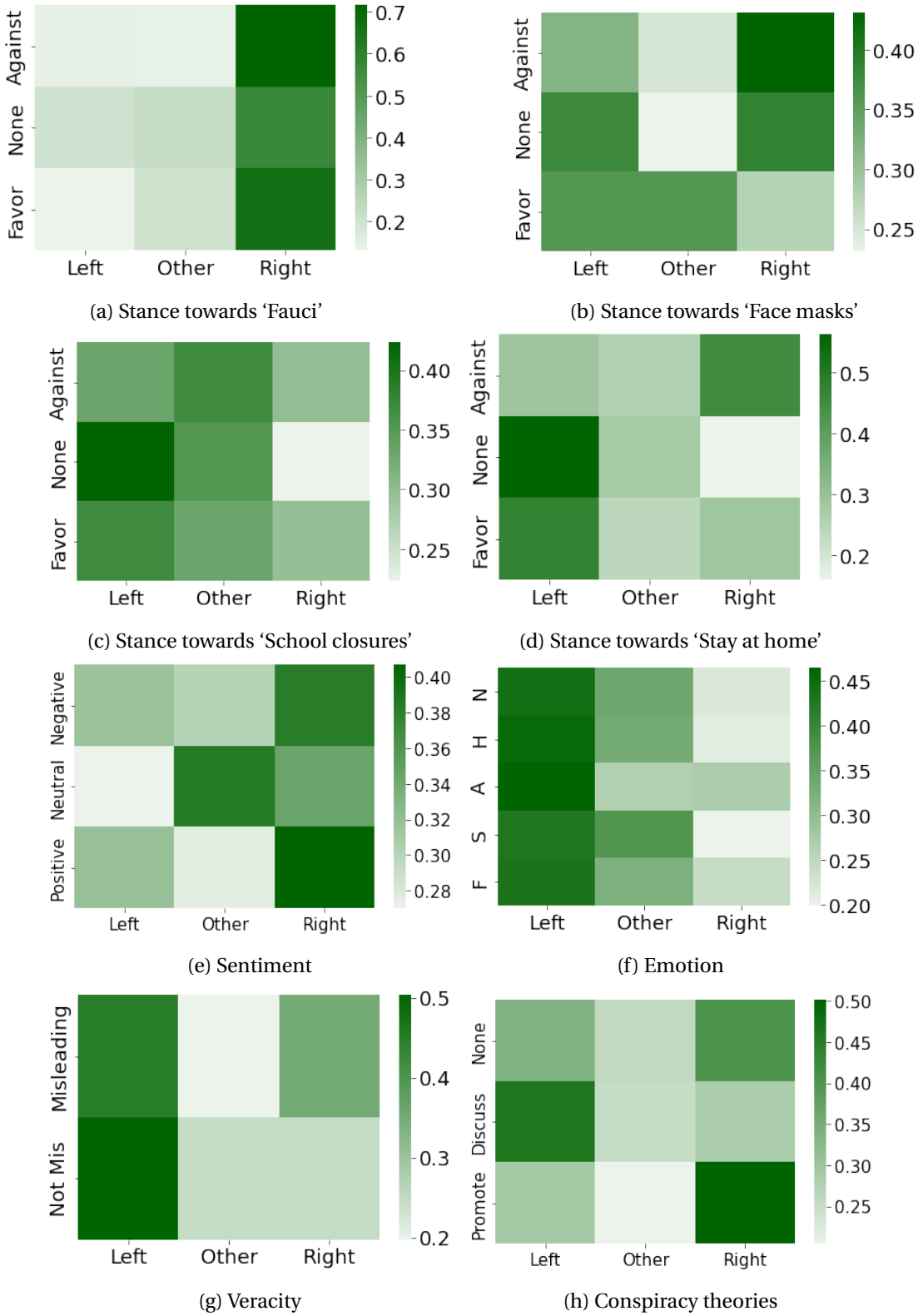
(e) Sentiment

(f) Political bias

(g) Veracity

(h) Conspiracy theories

Figure 5.2: Distribution of the labels for the emotion feature



(a) Stance towards 'Fauci'

(b) Stance towards 'Face masks'

(c) Stance towards 'School closures'

(d) Stance towards 'Stay at home'

(e) Sentiment

(f) Emotion

(g) Veracity

(h) Conspiracy theories

Figure 5.3: Distribution of the labels for the political bias feature

political bias or the other. However, Figure 5.3g shows that potentially misleading tweets are not tied to one political bias, while non-misleading tweets are more likely to be shared with left political bias. Moreover, Figure 5.3h shows that tweets discussing conspiracy theories are more likely to have left political bias, while it is the opposite for tweets promoting conspiracy theories. This supports the findings in [118], which states that conservatives tend to share more anti-science information than pro-science, thus being more inclined towards conspiracy theories.

5.2 Detecting Tropes

5.2.1 Introduction

A trope is an easily recognizable device used in narratives to convey a specific theme or idea [46]. This mechanism is widely used in the movie industry to generate effects and emotions in the audience, as it can be traced back to the familiar feeling a person may sense when knowing what is coming up next in a given scenario [132]. Examples of tropes in this context are “the girl next door”, “the love triangle”, and “the damsel in distress”. In fact, tropes are used today in almost any form of communication, given their ability to convey attitudes and beliefs. In particular, just as storytellers in media use tropes to make stories more understandable and relatable, online content producers use them to communicate news and opinions, exploiting tropes’ familiarity and preconceived notions.

It has been observed, however, that this mechanism used in movies and literature to impact the audience’s perception is often used online to manipulate and deceive audiences [36]. Notably, the pervasive use of tropes in online anti-vaccine discourse holds significant potential for dangerously shaping public opinion, as it can lead to individuals making uninformed vaccination decisions [61]. These tropes persist over time, recurring across various vaccines and contributing to ongoing anti-vaccine dialogues. For instance, in the 1800s, some people argued that natural methods were better than getting inoculated with cowpox-derived smallpox vaccines [68]. Fast-forward to today, we see similar claims that traditional cures are more effective than mRNA vaccines. Indeed, in spite the differences in details, most underlying tropes are consistently used across time and topics as well. For example, the narrative that “authorities cannot be trusted to make a decision that will benefit people” can be found both in the context of immigration (e.g., border control) and vaccine (e.g., vaccination policies) to invoke skepticism towards those authorities.

Therefore, the development of techniques for detecting and understanding these deceptive narrative elements is crucial to monitoring public discourse and promoting evidence-based communication. To this aim, we note that tropes are used not only in extended narratives but also in shorter forms of communication. For instance, a cinematic trope can be detected from

	Tweets	Ground Truth	Predicted Label
(a)	*Idc what you say, you're selfish if you refuse to wear a mask. This shouldn't be political. #MaskUp #MaskMoaners*	Favor 'Face masks'	Negative sentiment
(b)	*@kylegriffin1 Close the damn schools until there is a vaccine. #NotMyChild*	Favor 'School closures'	Negative sentiment
(c)	*If grocery stores can be open and people can risk their lives working there, then so can the schools and teachers. #OpenSchools*	Against 'School closures'	Neutral sentiment
(d)	*Corona virus Day 4 diary entry: I have now been social distancing for the past 26 years.*	Sadness	Neutral sentiment
(e)	*Italy Declares State of Emergency Over Wuhan Coronavirus*	Fear	Neutral sentiment
(f)	*Biden blames rise of COVID-19 cases on the unvaccinated: "This is a pandemic of the unvaccinated."*	Not Misleading	Neutral sentiment
(g)	*@saraecook Fauci is such a hypocrite! He knew back during the SARS outbreak most people who died was due largely to cytokine storm. Much the same with Coronavirus. He had no problem with Hydroxychloroquine being used then. #FauciFraud*	Against 'Fauci'	Anger emotion
(h)	*Trump and the White House are straight up publicly attacking the country's leading infectious disease expert during a #pandemic that has already killed nearly 140,000 Americans. Yup, that tracks. #COVID19 #DrFauci*	Favor 'Fauci'	Anger emotion
(i)	*The policymakers need to consider the fact that schools can't run without fees and teachers can't survive without salary. #SaveOurSchools*	Against 'School closures'	Sadness emotion
(j)	*@GrandadJohn5 Good news that County cricket is starting up but no news on recreational cricket the cut off point for our league is August 8th after that only friendlies or right off the season #StaySafeStayHome*	No stance towards 'Stay at home'	Happiness emotion
(k)	*Coronavirus: is it safe to travel and should children be kept home?*	Positive sentiment	Fear emotion

Table 5.2: Examples of tweets from all the datasets. Ground Truth indicates the label of the tweet in its original dataset, while Predicted Label is the output of one of the trained models

	Tweets	Ground Truth	Predicted Label
(l)	*I had such a good day with the students at @UNCG and @ncatsuaggies discussing activism, social justice, & organizing. They were incredible.*	Left political bias	Happiness emotion
(m)	*THE ATTACK ON FREEDOM OF SPEECH CONTINUES! #CrookedHillary will destroy the 1st Amendment Right of her Opposition!*	Right political bias	Anger emotion
(n)	*Take sports away and Social Interaction in schools. Your kids will have a great immune system! Way to teach your kids your saving them from the coronavirus. Bill gates and all Ted Talk technocrats have wanted online learning for years. Wake up! #OpenSchools*	Against 'School closures'	Right political bias
(o)	*Wearing a mask and social distancing doesn't mean you are "living in fear." It's like wearing a seat belt or using your headlights in the rain, it's for your safety and the safety of others. #WearADamnMask*	Favor 'Face masks'	Left political bias
(p)	*@Athens108 @realDonaldTrump It may have worked for an old Coronavirus that is a different virus from COVID19. All the studies coming out says chloroquine does not work for this virus. Dr. Fauci is always on the frontlines for all viruses. #FauciIsAHero*	Favor 'Fauci'	Right political bias
(q)	*87% of the deaths were caused by democrat leadership. Things like forcing nfected patents into nursing homes by executive order and banning HCQ. We now know that HCQ could have easily saved over 100000 lives over 20000 in NY alone. Trump was right. Democrats own the pandemic*	Promote conspiracy	Right political bias

Table 5.3: (Cont.) Examples of tweets from all the datasets. Ground Truth indicates the label of the tweet in its original dataset, while Predicted Label is the output of one of the trained models

a single scene, such as "love at first sight" in a fleeting glance exchanged by two characters. Similarly, the underlying message of a trope can be discerned from brief textual content – without any explicit mention of the trope itself, e.g., the “love triangle” in “Paul likes Anne, but his friend Harry met her first.”

Building on this observation, in this work, we strive to address the challenge of detecting *online tropes* in short text segments from social media. Automatically detecting online tropes from short text presents a challenging technical problem, as it requires not only the accurate identification of nuanced narrative elements, but also the ability to extract and interpret context-dependent patterns within limited textual information. To address this problem, we define the general task of automatic trope detection. We start by providing the definition for nine tropes after an iterative qualitative coding process of online social posts discussing vaccines and immigration. These tropes are general, as they are common in discourses on any matter, but we found out that they are often used in these specific domains. Given the trope definitions, we create the first corpus of labeled short texts. This dataset highlights the prevalence of this problem and its distinct nature compared to other text classification tasks. Leveraging supervised machine learning techniques for multi-label classification, we present methods that can identify tropes even with limited textual information.

Numerous works focus on enhancing online information quality through text content analysis, including computational fact-checking [55, 97], identification of conspiracy theories [129], and detection of propaganda/persuasion techniques deployment [27]. However, although trope identification is a powerful means to enhance our understanding of storytelling techniques, and effectively uncover implicit biases in many contexts, the task of trope detection has been ignored by the research community. In this work, we aim to bridge this gap.

Our contributions can be summarized as follows:

- We define the task of automatic trope detection and discuss its distinctions from prior research, focusing on the context of vaccine and immigration discussions on social media.
- We develop and provide a dataset of 3.3K vaccine and immigration related Twitter posts labeled with tropes.
- We demonstrate how supervised machine learning techniques for multi-label classification perform in this new task.
- We show that tropes are widely used online and analyze how these labels correlate with other popular tasks in text classification.

5.2.2 Task Definition

We start with a definition of online tropes, then list the tropes we identified, and finally present our problem formulation.

Definition. We use the term *trope* as defined in the movie industry: "a storytelling device or convention, a shortcut for describing situations the storyteller can reasonably assume the audience will recognize"[9].

By *online trope*, we mean a trope used in online discussions. These tropes are not used to refer to plots, but rather to human situations. Even if the general behavior, habit or issue is not stated explicitly, the reference is clear to the reader.

As examples of the trope "Natural is better", which is often used in discussions about a variety of topics, consider the following texts :

t_1: `"Not sure I will get the vaccine, natural immunity is the best immunity".`

t_2: `"GMO food is created by corporations to make profit, cannot be better than natural food".`

The writers of these messages are both advocating for natural solutions as the most healthy. Online tropes appeal to popular concepts, common experiences, or part of a culture that is known by the target audience.

Online Tropes. We outline the definitions for nine online tropes used in short texts that we have identified through our analysis of tweets related to two major topics: vaccine and immigration. We point out that we focus on tropes that can be found in general discussions, not necessarily involving the two topics at hand. To pinpoint these tropes, we employed a systematic and iterative qualitative coding process consisting of four phases: familiarization (reviewing the literature on tropes and examining thousands of topic-related tweets), open-coding (labeling tweets with potential trope codes), framework development (organizing codes into themes and higher-level categories), and finally verification (re-validating the established categories by applying them to the tweets examined during the open-coding phase).

We list below the online tropes, and we refer to Table 5.4 for the corresponding complete examples.

- **Skepticism Towards Authority (STA)**. Text appeals to skepticism towards scientific experts or political authorities, with statements such as "They should know/do better" and "They don't know what they are doing". An example message is "authorities have

[9] https://tvtropes.org/

failed now and before".

- **Defend The Weak (DTW)**. Text emphasizes the negative effects of something (e.g., vaccine, immigration policy) on vulnerable populations, with statements like "it is especially harmful to children". Example messages: "we must protect the weak", "they put the weak ones in danger".
- **Hidden Motives (HM)**. Text alludes to underlying agendas, suggesting that something (e.g., vaccines, illegal immigrants) is promoted by individuals with malicious intentions (such as hypocrites and tyrants) and concealed motives ("There is clearly an untold story behind it"). Examples of messages are "we must stop this scam" and "they are lying for their interest".
- **Liberty, freedom (LF)**. Text emphasizes personal autonomy and rights, using statements such as "my body, my choice", "not anti-something but pro-choice", and "people were stripped of their rights, jobs, freedom and forced against their will." Examples of messages are "I should be able to do what I want" and "They are forcing on me something I don't want".
- **Natural Is Better (NIB)**. Text promotes the idea that natural or traditional approaches are superior, with assertions like "natural immunity is the best immunity", "traditional solutions are more effective and secure", and "nature had a solution for this". Examples of messages are "I trust tradition more than innovation" and "They want to force non-natural solutions".
- **Time Proves Me Right (TPMR)**. Text appeals to the eventual validation of one's argument over time ("time will prove me right") and asserting foresight ("I told you this would happen"). Examples of messages are "I knew it / I know what is gonna happen" and "They don't see the problem coming"
- **Too Fast (TF)**. Text implies that something (such as vaccines) is unsafe or unreliable because it is experimental, untested, developed too quickly ("haste makes waste"), or not yet fully approved by authorities. Example of messages is "They rushed the decision".
- **Scapegoat (SC)**. Text that attributes blame for a (possibly under-specified) problem to a person or entity not directly involved, such as "They claim it's A or B's fault, but it's really X's fault", or assigning responsibility for an issue to a popular entity, such as Bill Gates. Example of message is "It is all their fault!".
- **Wicked Fairness (WF)**. Text compares to how two entities are being treated, highlighting application of different principles for similar situations (i.e., double standard). Some examples use questions, "Why can't X have access to Z while Y can?", if/then statements "If X can be punished for that, then Y should be punished as well", or the claim "It's not fair".
- **None**. Texts that do not fit clearly into any trope category. A portion of these tweets contains misinformation and conspiracy theories related to vaccination or immigration without involving tropes. For instance, content suggesting that vaccines cause autism.

Problem Definition. Given a short text, our goal is to assign one or more labels corresponding to the online tropes used in expressing the message, if any. Identifying the trope category can be a complex task, posing challenges for automated methods.

Notice that tropes can be seen as a tool used in persuasion techniques to achieve their goals [27]. For example, tropes such as "Defend The Weak" can be used to implement the "Appeal to fear" persuasion technique. Similarly, the "Antivax" conspiracy theory could use a "Hidden Motives" trope.

5.2.3 Dataset

We opted to use supervised learning to detect tropes automatically. Thus, we created a ground truth for the model to learn from, focusing on topics that have been strongly debated in recent years and in which they can oversimplify complex matters and deteriorate public discourse. Specifically, we built a dataset comprised of short texts retrieved from Twitter (now X) by using the keywords "vaccine" for one topic, and "migration", "migrant" and "asylum" for the other. The retrieval did not take into account the specific user when scraping for texts, but it was keyword centered. We keep only posts written in English. We point out that we did not check the presence of misinformation in these posts, we simply collected tweets in which the keywords occurred at least once.

Annotation process

The annotation activity was guided by the following general criteria:

- A trope is a storytelling device, which exploits a shortcut for describing situations the storyteller can *reasonably assume* the audience will recognize. For this reason, if the presence of a trope in a text is likely but not evident, the text has to be annotated with the label "none".
- A short text can involve more than one trope. Hence, the labelling has to include all relevant tropes, not just the one that appears to be the "strongest".

To start, four co-authors[10] reviewed independently about 200 tweets and annotated them according to the nine tropes mentioned in the previous section. Next, they compared and discussed the tweets with disagreement in the labels to refine the labeling process. The Cohen's kappa coefficient agreement of annotation of the sample before the refinement was 0.62.

We realized that we encountered difficulties in labelling the texts with certain features, such as

[10]The pool consisted of three males and one female. Their ages spanned between mid 20s and mid 50s. Annotators span two nationalities.

the use of sarcasm (which is very difficult to detect without context), references to different cultural aspects, and generally mixed-up topics brought into the argumentation. Moreover, we realized that some posts involved tropes we had not defined with precision: thus, we refined and redefined the labels each time this happened.

This initial activity was followed by another round of labeling, by four independent annotators, of 3.1K new posts with the refined labels. A subsequent consolidation meeting with all authors on all the posts resulted therefore in a set of around 3,300 annotated tweets with unanimous agreement. 2,074 tweets (63%) are about Vaccine and 1,230 (37%) are about Immigration.

During the annotation process, we made sure that the data did not contain any information that identifies individual people.

Data Analysis

Table 5.4 shows the distribution for each label. Despite the sampling of the scraped dataset being totally random, the tweets resulted to be fairly balanced after assigning the labels, in terms of texts with tropes and texts without them.

Interestingly, the result aligns with previous findings on key elements narratives, where studies [120] found that the most frequent conversation about vaccines on social platforms involved a concept they labeled *liberty and freedom*. Conversely, the least numerous labels are Time Proves Me Right, Natural Is Better, Scapegoat, and Wicked Fairness. These tropes probably require a deeper dialectic, as the speaker tries to bring forth a kind of reasoning, making them more sporadic throughout the dataset compared to other, more direct arguments that characterize other tropes.

We investigated the correlation among the tropes we defined, to show that they would not overlap. As shown in Figure 5.4, no significant signal was detected. It is possible, however, given the nature of the problem and the way to express opinions, that some tropes are used together more often, like, for instance, the feeling of distrust towards science (Skepticism Towards Authority) that developed a solution too quickly (Too Fast), or pointing out a double standard (Wicked Fairness) while referring to a vulnerable target (Defend The Weak).

Vaccine vs Immigration

Topics such as Vaccine and Immigration inherently trigger different discourse on social media. Even though most Tropes are found on both topics, there are some significant differences between the two sets of tweets. We notice that tropes appear twice more often on the Vaccine topic than on the Immigration topic. We also found that the tropes STA, TF, NIB, LF and HM are shared more in Vaccination topics, the tropes DTW and WF in Immigration topics,

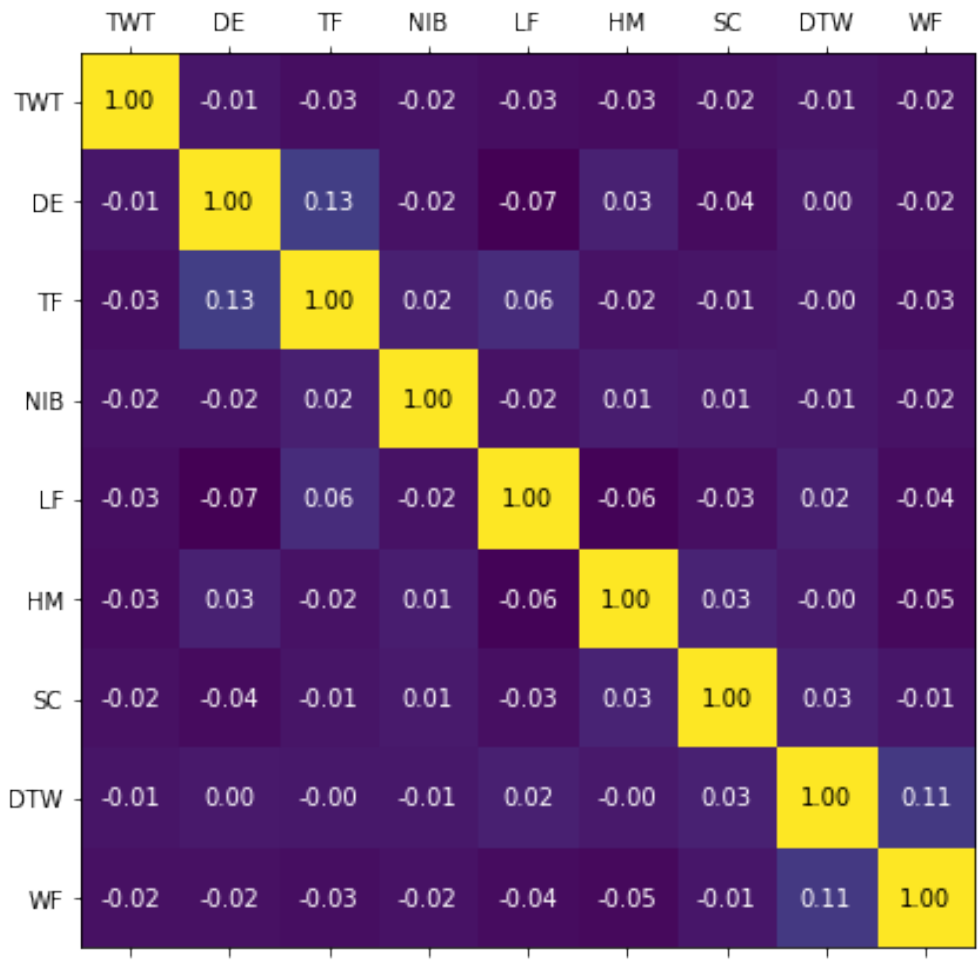


Figure 5.4: Correlations between tropes using the Pearson coefficient.

and the tropes TPMR and SC are shared equally in both subsets. In fact, WF is only found in Immigration tweets and TF is only found in Vaccine tweets.

5.2.4 Models

We model the problem of trope detection as a multi-label classification task, which focuses on categorizing instances into several non-exclusive classes, with each associated class of an instance referred to as a label. We describe below the four models used in our study. Fine-tuned models are trained on 80% of the examples in the annotated dataset. All models are tested on the remaining 20%. All reproducibility settings of our experiments (hyper-parameters, prompts, etc.) are shared in Appendix A.3.

Bert-FT. To predict one or more tropes for a given tweet, we fine-tune a BERT-large-uncased pre-trained language model using HuggingFace (345M parameters). We save the model with the best average F1-score on the validation set out of 20 epochs.

CovidBert-FT. Given that we analyze tweets and most of them discuss covid-related topics, such as vaccine hesitancy, we also fine-tune a second language model, COVID-TwitterBERT (CovidBert), which is a BERT-large model pre-trained on COVID-related tweets [95] (336M parameters). We follow the same fine-tuning process used for Bert-FT described above.

ChatGPT-ZeroShot. We model the trope detection task with the ChatGPT-3.5 turbo[11] engine (175B parameters). We use the OpenAI APIs to request ChatGPT to label all the texts from our dataset with the tropes we have identified. We use a Zero Shot approach by prompting, to obtain the labels, using only the tweet at hand and the trope definitions. The definitions of the labels prompted to ChatGPT are the ones reported in Section 5.2.2. The prompt itself is in the Appendix section.

Llama-3-ZeroShot. We also use an open weight LLM to perform the Trope classification task. We chose the 'Meta-Llama-3-8B-Instruct' model from Huggingface[12] (8B parameters) using the same prompt used with ChatGPT-3.5, baring a few adjustments to fit the Llama prompting syntax.

5.2.5 Experiments

We first report results for the trope detection task over our annotated dataset (**Tropes**). We then show how tropes might be correlated to other textual features, namely conspiracy theories and persuasion techniques. Finally, we discuss the results.

[11] gpt-3.5-turbo-0125
[12] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

Models for Trope Classification

In the first experiment, we evaluate and compare the four alternative trope detection models over our annotated validation dataset. We compute, for each trope, the F1-score, as well as for the 'None' category. We also report the weighted average of the F1-score across the dataset.

Table 5.5 reports the F1-scores and overall results of our models. It shows that CovidBert-FT has the best performance with a weighted average F1-score of **0.65**. Both supervised models perform better than LLMs with zero-shot prompting.

Results also show that some tropes are easier to detect than others. Indeed, both LF and TF obtain high F1-scores. However, models struggle to detect the TPMR trope. One explanation of this difference can be found in the most frequent bi-grams of each trope, where clear messages exist in LF ('body, choice', 'experimental, vaccine', or 'vaccine, mandates') and TF ('clinical, trials', 'trials, future' or 'emergency, use') while no clear insights appear for TPMR ('covid, vaccines', 'long, term' or 'wait, til'). Another reason as to why some classes are harder to detect than other, is because of the low number of samples in the training set. Some tropes, such as TPMR, have a low number of examples (around 2.3% in our dataset). This makes it more challenging for supervised models to properly learn how to detect them. Conversely, tropes with a high number of samples tend to be easier to detect, such as LF (10.4%).

Overall, models perform well in the binary detection of the 'None' class.

Additional results are available in the Appendix, giving further insights on the difference between training supervised models only on Vaccine or only on Immigration data. Finally, in the Appendix, we also report an in-depth error analysis, giving false positive examples for every class. The main takeaway is that false positives come from model over-fitting on certain keywords.

Persuasion Techniques and Conspiracy Theories

This section is devoted to studying the relationship between tropes and two detection tasks in misinformation analysis: (i) the use of persuasion techniques and (ii) the presence of text discussing and promoting conspiracy theories about COVID-19.

For this study, we select two datasets for which we have a ground truth, specifically:

- **Persuasion techniques**: a set of 7k texts extracted from online memes annotated with human-provided labels indicating the use of persuasive techniques[13] [34].
- **Conspiracy**: a set of 2k tweets about Covid manually annotated with labels for nine conspiracy theories [75].

[13] Even though the data contains memes, only the textual content was used for the annotation.

Given that the CovidBert-FT shows the best results for trope detection, we use it in the rest of the experiments. To detect the use of persuasion techniques and conspiracy theories in our **Tropes** dataset, we rely on state-of-the-art models from the literature, specifically PERSUASION TECHNIQUE DETECTION [108] (see Section 3.2) and CONSPIRACY DETECTION [106] (see Section 3.1).

Tables 5.6 and 5.7 show the results from the execution of the models for the **Conspiracy** and **Persuasion Technique** datasets, respectively. In both tables, we report the human-labelled results (ground truth) in italic. The other results are obtained by running the detection models and can therefore be noisy. We remark that our model is trained only for trope detection: any text containing conspiracy theories or persuasion techniques, but without tropes, is labelled as "None". In this experiment, we compare all tasks at a binary level, i.e. the use of Tropes, Persuasion Technique, or Conspiracy Theories in text.

Comparison with Conspiracy Theories. First, we can see in Table 5.6 that the proportion of tweets that contain conspiracy theories is constant across both datasets (around 50%). This holds for the proportion of tweets containing Tropes. This shows that datasets are not biased towards the textual features they annotate. We also analyze how Tropes and Conspiracies appear together on those datasets. In both datasets, more than 60% of tweets contain at least a Conspiracy or a Trope, showing the prevalence of such features in social media posts. We also analyze the correlations between tropes and conspiracy theories using Matthews correlation coefficient. The only positive correlation found is with the 'Hidden Motives", even though the coefficient is low (Matthews correlation coefficient is 0.19). This confirms that Tropes and Conspiracy Theories are orthogonal concepts.

In order to evaluate that our Tropes model can be applied to Conspiracy data, we perform manual validation by checking 40 tweets positively labeled by our model with high confidence. We obtain a binary F1-score of 0.943, highlighting that our model can be used outside its training distribution.

Comparison with persuasion techniques. Table 5.7 shows the proportions of both Tropes and Persuasion techniques in both datasets. We notice that Persuasion Techniques are used a lot more than any other textual features, however, tropes seem to be used less in online memes. We also note that their appearance together is not consistent across both datasets. This may be due to the fact that both data are coming from different sources (social media textual posts for Tropes and memes for persuasion techniques) and about different topics. Indeed, the persuasion dataset contains a significant amount of memes heavily biased towards US politics, most of them being offensive to certain groups of people. We have found no positive correlations between tropes and persuasion techniques (Matthews correlation coefficients are less than 0.07).

We also evaluate the performance of our Tropes model on persuasion technique data by manually labeling the textual content of 60 memes positively labeled with high-confidence by our Tropes detection model. We found a binary F1-score of 0.843, showing that our model can safely be used on this other kind of content.

Discussion

Results from these experiments highlight some interesting insights. First, we see that Tropes exist independently of Persuasion Techniques and Conspiracy Theories in the online discourse about Vaccines and Immigration. They therefore provide new information that can be used to understand written language better, in addition to existing textual features. Indeed, we show that tropes are orthogonal to conspiracy theories and persuasion techniques. As much as mentioning a conspiracy or using a certain persuasion technique does not necessarily imply spreading misinformation, we believe that tropes are yet another dimension of analysis that should be studied.

One more important aspect separates tropes from conspiracy theories and persuasion techniques. Tropes can be used to polarize opinions either way, in a more neutral manner: in this context, most of the time, they are used to belittle the efforts of experts but it is not the only way. Consider for instance the following sentence:

t_3: `"Great point - collectively, we failed to get the vaccine to hundreds of millions of people who needed it because Canada, the USA, the UK, and others supported windfall profits of drug companies over people's health."`

Here, the "Hidden Motive" that led to negative consequences was used to support the argument for the failure of vaccine availability.

Results also show that LLMs struggle to detect Tropes, but supervised models reach convincing performance. However, not all Tropes are detected with the same precision, giving us insights about the difficulty of the task. For example, the trope TPMR shows poor performance from the models.

Lastly, we manually annotated conspiracy tweets and persuasive memes on a high-confidence threshold, as we believe that precision is a more important metric than recall in an out-of-distribution setting. This way, we can reliably detect documents with tropes, which provide useful information for our study. The classes detected the most out-of-distribution are **Defend The Weak**, **Hidden Motives** and **Liberty, Freedom**.

5.2.6 Related Work

Several works study the impact of false information and misleading narratives on online social media platforms. For identifying and addressing misleading information in online text, current techniques focus on detecting (i) veracity (fact-checking) [55, 97], (ii) the use of propaganda or persuasion techniques [27], and (iii) support for conspiracy theories [129]. Some of these works specifically focus on vaccine-related content [38, 65], but to our knowledge, there is no work yet on vaccine or immigration tropes detection. We remark that persuasion techniques are methods employed to manipulate public opinion and promote a specific agenda, while tropes are communication devices that are not inherently tied to misinformation. Examples of persuasion techniques are "reductio ad hitlerum", to discredit an idea that is popular in groups hated by the audience, and "bandwagon", to appeal to the popularity of an argument [28]. A trope is also different from a text that supports a conspiracy theory. The latter is focused on content, i.e., on the entities and arguments for the topic at hand, while the former is rather a tool for achieving a communication goal. Indeed, a given text that refers to a conspiracy theory (the "what") can use different techniques to convince the audience, including tropes and propaganda techniques (the "how"). Similarly, tropes are different from themes [64, 101]: while themes are the central messages conveyed by a narrative, tropes provide familiar and concise elements that can be used to implement multiple complex themes, e.g., the same tropes are found in both the Vaccine and Immigration posts.

TV tropes have been widely studied, given their persuasive use to simplify narratives and improve communication [46, 132] and the problem of trope detection has been studied on a TVTropes dataset of 5.6k movie synopses and 95 tropes [24].

There are several studies that focus on analyzing the public discourse surrounding vaccines and vaccine hesitancy, as well as the use of tropes and misinformation in this discussion [36,61]. It has been observed that a multitude of narratives, including tropes, converge to create an environment of extreme uncertainty in the vaccine information ecosystem [68, 120]. Studies have also been done on the problems with the immigration discourse on social media [40, 86]. None of them, however, propose methods for the automatic detection of tropes in this context.

In this work, we focus on supervised ML algorithms for detecting tropes in short texts. We model the problem as a multi-label classification task and report results for state-of-the-art methods using pre-trained language models, such as BERT [33] and GPT [19], using fine-tuning and zero-shot learning.

5.3 Conclusion

In this chapter, we introduce approaches to detect multiple textual features in tweets. These approaches help answer RQ2, by extracting these factors and analyzing their use on social media.

We analyze the correlations between the emotion, sentiment, political bias, stance, veracity and conspiracy theories factors. We leverage relevant datasets to train three models to predict emotion, sentiment and political bias on COVID-19 related tweets. These models allowed us to analyze the conditional distribution of the different labels to better understand the online discourse. Main findings include that COVID-19-related regulations topics, such as 'Face masks', 'School closures' or 'Stay at home orders' are highly controversial, generating a lot of negative sentiment and anger emotion in the Twitter discourse. The users' political bias on those topics also outlined the stance of US politicians in the debate. Similarly, conspiracy theories are usually promoted with negative sentiment and right political bias, which might reflect the inclination of conservatives towards anti-science information.

In addition, we define the task of trope detection and demonstrate its distinct nature compared to other text classification tasks. We create and share a unique ground-truth dataset of 3,300 vaccine (63%) and immigration (37%) related Twitter posts labeled with common tropes, which can be used to further advance this area of research. Results show that supervised approaches for multi-label classification achieve significant success in detecting tropes. Our work contributes to a better understanding of public opinions and biases through the lens of tropes.

This chapter answers RQ2 by introducing models to detect factors, and using them to analyze social media dynamics. Future work could increase the number of factors, or extend the scope of our tropes dataset by incorporating additional topics to better understand tropes' usage in different domains.

Posts	Tropes	Vaccine	Immig.	Total
Our government should stop the boats from coming, not help them to shore.Mark my words this issue is not going away	Time Proves Me Right (TPMR)	43 / 2.1%	33 / 2.7%	76 / 2.3%
The FDA's 'Future Framework' for COVID Vaccines Is Reckless Plan The dangerous mandates should be ruled unconstitutional!	Skepticism Towards Authority (STA)	194 / 9.4%	30 / 2.4%	224 / 6.8%
As Trudeau still goes on pushing this untested experimental vaccine using mRNA that has never been used successfully before on people!	Too Fast (TF)	142 / 6.8%	0 / 0%	142 / 4.3%
These vaccines are a negative cost/benefit for most people, particularly those with natural immunity.	Natural is Better (NIB)	63 / 3.0%	3 / 0.2%	65 / 2.87%
My body my choice no vaccine for me, but that woman over there? I decide her medical operations' - Everyone okay with the SCOTUS decisions.	Liberty, Freedom (LF)	325 / 15.7%	19 / 1.5%	344 / 10.4%
Well here is the exclusive footage of migrants throwing their phones into the Channel. Why would legitimate refugees with nothing to hide throw their mobile phones into the sea?	Hidden Motives (HM)	244 / 11.8%	58 / 4.7%	302 / 9.1%
Well, it HAS TO BE either Climate Change or Putin! It can't possible be anything related to the mRNA vaccines, right?!	Scapegoat (SC)	58 / 2.8%	19 / 1.5%	77 / 2.3%
Publix Declines to Offer Coronavirus Vaccine to Children Under 5 PUBLIX IS PROTECTING OUR BABIES FROM THE POISON IN THE VACCINE	Defend the Weak (DTW)	99 / 4.8%	78 / 6.3%	177 / 5.4%
We can't find homes for the 6000+ homeless veterans yet we can find them for thousands of illegal immigrants crossing the channel	Wicked Fairness (WF)	0 / 0%	68 / 5.5%	68 / 2.1%
these vaccines becoming like those goddamn app updates.	None	1100 / 53	968 / 78.7%	2068 / 62.6%

Table 5.4: Examples of tropes occurring in tweets and frequency of their presence in our dataset.

Model	STA	DTW	HM	LF	NIB	TPMR	TF	SC	WF	None	*Avg*
Bert-FT	0.54	0.57	0.42	0.78	0.50	0.33	0.75	0.48	0.55	0.83	0.58
CovidBert-FT	0.60	0.68	0.59	0.80	0.55	0.27	0.77	0.64	0.57	0.87	**0.65**
ChatGPT-3.5-ZeroShot	0.19	0.36	0.27	0.66	0.27	0.00	0.31	0.20	0.44	0.55	0.32
LLAMA3-8B-ZeroShot	0.15	0.29	0.20	0.38	0.27	0.12	0.16	0.10	0.10	0.24	0.23

Table 5.5: F1-score results for our models for each trope, the 'None' class, and weighted average across the dataset.

Dataset	Consp.	Trope	Both	Trope Only	Consp. Only	None
Tropes	49.9%	*37.4%*	24.1%	13.3%	25.8%	36.8%
Consp.	*51.9%*	30.4%	19.9%	10.5%	32.0%	37.6%

Table 5.6: Proportions of conspiracy and tropes in respective datasets. Ground truth in italics.

Dataset	Pers.	Trope	Both	Trope Only	Pers. Only	None
Tropes	91.7%	*37.4%*	35.9%	1.5%	55.8%	6.8%
Persuas.	*81.9%*	11.4%	10.6%	0.7%	71.3%	17.3%

Table 5.7: Proportions of persuasion techniques and tropes in respective datasets. Ground truth in italics.

Chapter 6

CimpleKG: a Knowledge Graph For Explaining Misinformation

This chapter presents Cimple KG, a knowledge graph of misinformation-related documents. Section 6.1 explains the creation of the graph from the ground-up and Section 6.2 showcases a usage of the graph through an exploratory search engine. This work has been published in [22], the code is shared on GitHub[1][2], the KG is accessible at a SPARQL endpoint [3] and the exploratory search engine at https://explorer.cimple.eu/.

6.1 The CIMPLE KG

We introduce a continuously updated public knowledge graph (KG) called CimpleKG[4] that can be used for supporting misinformation research. CimpleKG links various previously published static misinformation datasets with daily updated claims verification from vetted fact-checking organizations and augments them with additional information such as named entities and contextual factors (e.g., emotions, sentiment, political leanings, conspiracy theories, propaganda techniques).

Although our KG is not the first attempt at gathering and representing fact-checking data [136], CimpleKG is much larger than previous works in terms of time coverage, topics, country, language, quantity and freshness. It is also novel as it includes so-called factors extracted from the text to explain misinformation; it normalizes the rating schemes used by fact-checking organizations and also resolves shortened URLs to their unshortened version. This is useful as fact-checkers tend to use archiving URL services when referring to misinforming URLs. Finally, contrary to previous work, CimpleKG is continuously updated, making research more representative of the current misinformation landscape and near real-time integration into applications possible.

[1] https://github.com/CIMPLE-project/knowledge-base
[2] https://github.com/MartinoMensio/claimreview-data
[3] https://data.cimple.eu/sparql
[4] CimpleKG SPARQL Endpoint, https://data.cimple.eu/sparql

At the time of writing,[5] CimpleKG contains over 203k ClaimReview[6] spanning 26 languages, issued by 77 fact-checkers from over 36 countries. CimpleKG is updated daily as new claims are collected from fact-checkers. The KG has over 15m triples and also includes 217k documents from static datasets (news and well-known misinformation datasets of claims and tweets), 263k+ distinct entities and 1m+ textual features. Besides the aforementioned SPARQL endpoint, the daily collected fact-checks are also freely accessible as graph and non-graph serialized databases snapshots.

6.1.1 Connecting Misinformation, Reviews, Factors and Entities

The ability to assign credibility ratings to a piece of information or claim is key for the development of research and tools that try to better understand or address the proliferation of misinformation. In this context, since the 2000s, fact-checking organizations have been created to identify and verify claims that may be misleading, incorrect or harmful [51]. The types of fact-checked content can vary from political claims to health-related claims, and often involve the creation of an article that discusses identified claims and assigns them a rating or label that typically goes from completely *misinforming* to *credible*. Although these ratings or labels are not always the same between fact-checkers, the way they are structured has been standardized in the Schema.org vocabulary as ClaimReview (Section 2.2.3). In this paper, we use ClaimReview as the base of our KG and extend it with additional features such as textual *Factors* and named *Entities*. These features make it easier to discover how particular claims relate.

The textual content of a Claim associated with a ClaimReview typically involves some textual features or *Factors* such as emotion, sentiment, political leaning, propaganda techniques, and the mention of conspiracy theories that affect how specific claims are perceived. These factors can be extracted, to some extent, for a better understanding of how such features are associated with particular credibility labels. We extract these aforementioned factors automatically using the models developed in [106, 107] presented in Chapters 3 and 5. These models reported on average an $F1$ score of 0.71 ($\pm$0.09).

Claims typically mention named entities, such as specific individuals or locations. Identifying such entities makes it possible to formulate more advanced questions about claims. For instance, we can search all the claims that mention *Ukraine* or *Donald Trump*. In this paper, we extract and disambiguate entities from the claims using DBpedia spotlight[7] [87] because of its simplicity and computational performance. It also identifies broader non-named entities (e.g. “vaccine”), and supports many languages.

[5] These statistics are based on the 11th of April 2024 snapshot.
[6] ClaimReview, https://schema.org/ClaimReview.
[7] DBPedia Spotlight, https://www.dbpedia-spotlight.org.

Misinformation-related knowledge is not always completely captured by fact-checking organizations, and some of such information may be available in manually annotated research datasets [96, 111] or through specific social media verification programs [123]. These data sources may provide additional contextual information not directly found in fact-checks, such as social media mentions or conspiracy theory annotations. In this work, we integrate and link many of these static datasets to the ClaimReview data as they provide additional layers of information (Section 6.1.4).

6.1.2 The Cimple KG Data Model

As mentioned in the previous section, CimpleKG reuses the Schema.org ontology (denoted with the `sc` prefix in the rest of this document). An instance of a `sc:ClaimReview` is connected to a `sc:Claim` through `sc:itemReview`. It is also connected to the organization that fact-checked the claim through `sc:author`, as well as the issued rating through `sc:reviewRating`. We have created `co:normalizedReviewRating`[8] to provide a normalised rating which is a controlled vocabulary represented in the Simple Knowledge Organization System (SKOS) [91]. An instance of *Rating* has a name (`sc:name`) and a rating value (`sc:ratingValue`). If it is an original rating, it is also connected to the organization that used it through `sc:author` and is connected to the corresponding normalised rating through `sc:sameAs`. *SocialMediaPostings* are linked with *Claims* with `co:related` (based on some ground-truth from some datasets). We also provide the appearance of a *Claim* with `sc:appearance`. We use `sc:mentions` to link entities with any textual document (*ClaimReview*, *Review*, *Claim*, *SocialMediaPostings*, *NewsArticle*). Lastly, we extract textual features on the textual content and represent this information with the predicates `co:hasEmotion`, `co:hasSentiment`, `co:hasPoliticalLeaning`, `co:mentionsConspiracy`, `co:promotesConspiracy` and `co:usesPropagandatechnique`. An illustration of the data model is shown in Figure 6.1 and additional details about how to query the KG can be found on KG code and data repository[9].

The CimpleKG data can be accessed through a SPARQL endpoint and as RDF dump files.[10] All URIs are dereferenceable following the linked data principles. A RESTful API has also been deployed to access the KG.

6.1.3 Collecting and Integrating Newly Published Fact-Checks

The CimpleKG is generated using ClaimReview data collected from fact-checking organizations and various static datasets. Data from fact-checking organizations is continuously

[8] We prefix the newly defined properties and types in CimpleKG with the `co` prefix.

[9] CimpleKG repository, https://github.com/CIMPLE-project/knowledge-base.

[10] The RDF dumps and their automation are available as releases in https://github.com/CIMPLE-project/knowledge-base.

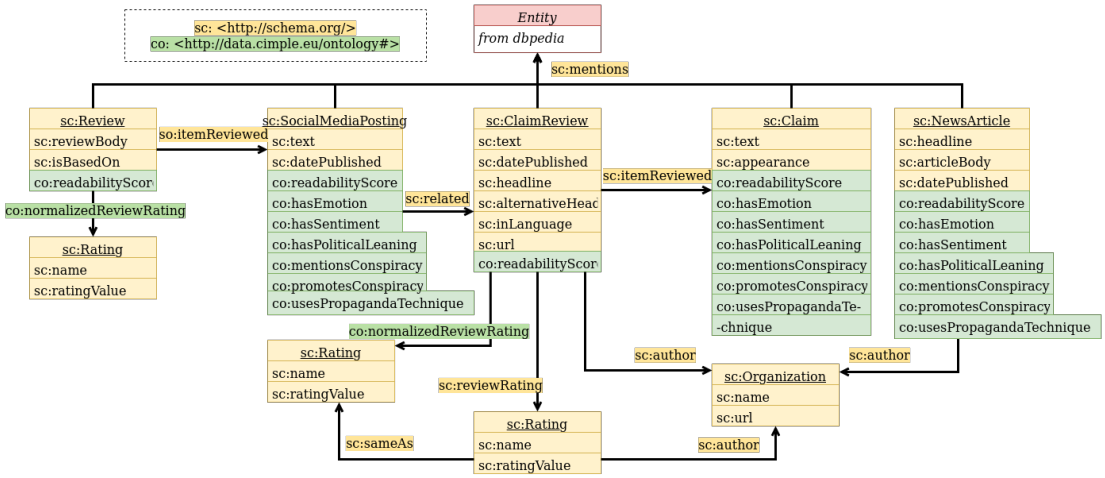


Figure 6.1: Illustration of the CimpleKG data model.

integrated, whereas static datasets from static sources are added as relevant datasets once identified and published. New data is collected at 10 am UTC daily and takes 3 hours and 20 minutes to process on average.

To integrate newly published fact-checks into CimpleKG, we rely on a two-step process where: 1) data is continuously collected from fact-checking sources and, then; 2) the collected data is mapped to the CimpleKG graph structure presented in Section 6.1.2. During this step, both related entities and additional textual features are extracted to complement the KG with additional relevant knowledge. The various steps required for collecting and processing the data are displayed in Figure 6.2.

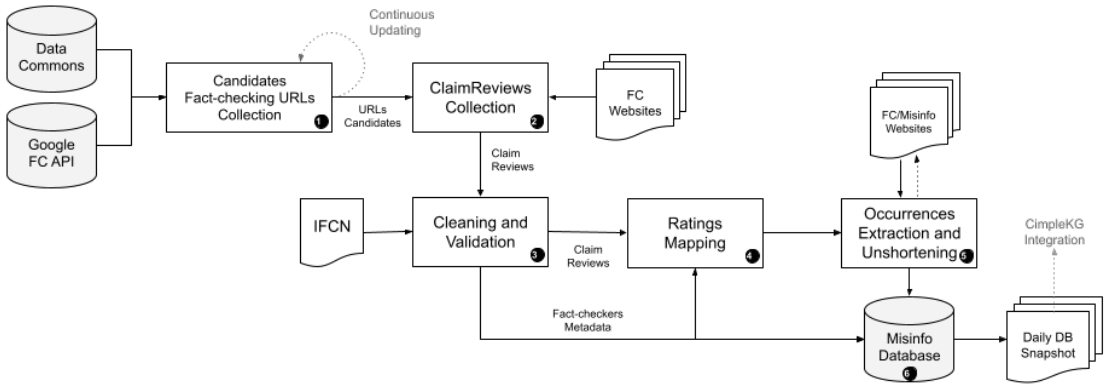


Figure 6.2: Data collection and processing pipeline for gathering ClaimReviews.

The two-step process generates two different versions of the misinformation data. First, the semi-structured data created as part of the data collection step is made available daily as a set of files[11]. Second, the KG version of the data is integrated into CimpleKG and made available.

[11] ClaimReview data, https://github.com/MartinoMensio/claimreview-data/releases.

The data collection and processing steps of the various fact-checks that are integrated into CimpleKG are shown in Figure 6.2 and can be divided into 6 primary steps.

1. *Collection of ClaimReviews URLs Candidates:* The first step required for collecting the fact-checks is to identify the URLs that contain them. We collect this data from DataCommons[12] irrespective of their publication language using their public data feed, and we use the Google Fact-checking API for obtaining additional URLs. We use these two aggregators because they contain the largest quantity of fact-checks, and they are updated very frequently. Going manually to all the IFCN signatories would require additional custom collection logic, while these aggregators can already provide the data together. For both data sources, we collect the URLs of the reviews. The other fields, especially from Google Fact-checking API, tend to be incomplete in the `appearance` and `firstAppearance` attributes. Instead, when scraping from the Google Fact-checking search interface background data, we are able to retrieve URLs of appearance, but they are frequently mixed with URLs whose stance does not support the claim. Since these fields are critical for understanding *where* misinformation happens, we find it best to recollect them directly from the fact-checkers.

2. *Collection of ClaimReview from fact-checkers:* The second step involves the retrieval of the ClaimReview data associated with the previously identified URLs directly from the fact-checkers' websites. This step is needed because the data collected from the previous step may be incomplete (the issue of missing `appearance`). For each URL collected during the first step, we obtain the page content where the corresponding ClaimReview appears. For some fact-checkers, the ClaimReview is not embedded in the source of the page, because the submission to Google may be performed on a private channel. As we see in Table 6.1, for most of the fact-checkers, we can collect the data with complete attributes, while with some fact-checkers the recollection fails (total recollection percentage: 71.07%, average recollection percentage: 50.87%).

3. *Validation and Cleaning:* The third step is designed for cleaning and validating the data collected in the previous step, as some of the data may be wrong or incomplete. To make the collected data usable, we try to fix and normalize it with several processes (e.g. dirty-json[13] to fix common JSON errors with strings or use multiple parsers to allow parsing JSON-LD transformed with different specifications). We discard items that are not easily fixable and, for the remaining ClaimReview, we only keep the ones that are from International Fact-Checking Network (IFCN) signatories[14] in order to ensure that the collected data is trustworthy (we discard 63,955 ClaimReview that cannot be

[12] DataCommons, https://www.datacommons.org/factcheck/download#fcmt-data.
[13] Dirty-JSON, https://github.com/RyanMarcus/dirty-json.
[14] IFCN, https://ifcncodeofprinciples.poynter.org/signatories.

verified). The list of IFCN signatories is updated every time new data is collected, and this data is used for adding information about fact-checking organizations such as their country of origin and language.

4. *Ratings Mapping:* Since each fact-checker uses a different type of rating, we need to map them to a common value (step 4 in figure 6.2). Similar to our previous work [88], we first try to use the numerical ratings provided by fact-checkers. We use the following mappings: *credible* when the rating is greater than 0.8, *mostly_credible* when the rating is between 0.6 and 0.8, *uncertain* when the rating is between 0.4 and 0.6, *not_credible* when the rating is less than 0.4, and *not_verifiable* when the numerical value is missing. For the textual labels, mappings are created for each fact-checker based on their textual labels so they map to the 5 aforementioned labels.

5. *Occurrences Extraction and Unshortening:* The next step (step 5 in the figure) is focused on extracting the `appearance` and `firstAppearance` fields from the collected ClaimReview that have them. The extracted URLs are then unshortened since many fact-checkers use URL shorteners or archiving websites to capture snapshots of the page for the content that then gets deleted. URL unshortening allow us to know the real URL where it appeared, so it can be used for tracking their appearance online rather than the more rarely used shortened version of the URLs.

6. *Misinformation Database and Snapshot:* The final step is to store the data in a database and export it in a format that can be easily processed for integration in CimpleKG. A snapshot is created daily based on the collected data and made available publicly. The data comprises both statistical information about the collected data and various subsets of the data.[15]

The integration of the daily collected data into CimpleKG follows also six primary steps. The code used for converting the daily snapshots is available at https://github.com/CIMPLE-project/knowledge-base.

1. *Claim Review text scrapping:* First, we extract the textual data of the new ClaimReview documents. We use the `trafilatura` python package [10] to retrieve the body of the Claim Review from the specified URL.

2. *Entity extraction:* We use DBpedia spotlight [87] to extract relevant entities in the text of the claims, and ClaimReview. This results in 192,183 distinct entities extracted. We also experimented with the latest spaCy models leveraging on LLMs that also extract non-named entities.

[15]The details of the daily snapshot and the description of each exported file can be found at https://github.com/MartinoMensio/claimreview-data.

Table 6.1: Recollected percentages from the top 30 fact-checkers. Total recollection percentage: 71.07%, average recollection percentage: 50.87%

Web Domain	Recollected	Total	Web Domain	Recollected	Total
afp.com	86.87%	33,727	youturn.in	0.00%	4,835
snopes.com	99.98%	16,321	dpa-factchecking.com	0.00%	4,822
vishvasnews.com	99.99%	13,417	indiatoday.in	99.71%	4,498
politifact.com	51.29%	12,718	newtral.es	0.00%	4,249
newschecker.in	99.95%	11,694	newsmeter.in	99.98%	4,238
boomlive.in	99.97%	10,270	fullfact.org	100.00%	4,118
factly.in	0.10%	8,394	thequint.com	99.97%	3,969
checkyourfact.com	99.91%	8,093	usatoday.com	0.00%	3,787
leadstories.com	100.00%	7,719	aosfatos.org	99.97%	3,559
altnews.in	99.96%	7,270	maldita.es	0.00%	3,440
factcrescendo.com	0.06%	6,992	dogrulukpayi.com	99.91%	3,422
uol.com.br	35.86%	6,926	correctiv.org	100.00%	3,331
demagog.org.pl	92.08%	6,088	factcheck.org	41.44%	2,985
sapo.pt	100.00%	6,020	observador.pt	100.00%	2,908
teyit.org	93.01%	4,953	tfc-taiwan.org.tw	0.00%	2,871

3. *Factors extraction:* We also extract *factors* from the textual content of the claim (Section 6.1.1). This results in 497,182 *factors* extracted.

4. *Conversion of objects to RDF triples:* Then, each `Claim`, `ClaimReview`, `Organization` and `Rating`[16] are converted to RDF triples. They are associated with their respective types and properties (e.g. name, datePublished, URL, etc.). For each resource, we generate a unique URI identifier using the SHA224 cryptographic hash function over a unique string identifier[17]. This way, ClaimReviews fact-checking the same claim will point to the same document in the KG.

5. *Connection of the objects:* We connect resources through the following Schema.org properties: author, mentions, reviewRating, itemReviewed and appearance. We also define our own set of properties for the tracking of *factors*. This results in a graph totaling 8,454,322 RDF triples.

6. *Mapping of the KG and serialisation:* Lastly, to map the collected data to the CimpleKG model, we use the RDFLib python library, and serialize it using the TTL file format. The data is then integrated into CimpleKG.

[16] Both original and normalized ratings are accessible

[17] The CimpleKG URI patterns are specified at: https://github.com/CIMPLE-project/converter/blob/main/URI-patterns.md.

Table 6.2: Statistics of the static datasets integrated into CimpleKG.

Dataset	Document Types	Nb. of Documents
AFP	News Article.	193,933 news articles.
Birdwatch	Social Media Posts, Reviews.	6,563 tweets, 1,983 reviews, 1,112 links to ClaimReview.
CLEF CheckThat!	Social Media Posts, Claim Reviews.	1,196 tweets, 1,198 links to ClaimReview.
MediaEval 2022	Social Media Posts.	2,702 tweets.
Propaganda Corpus	Claims.	1,908 claims.

6.1.4 Integrating Static Datasets with the Fact-checks

Integrating previously published misinformation datasets into the KG makes it possible to link existing fact-checked claims with related data such as social media posts (`sc:SocialMediaPost`) and news articles (`sc:NewsArticle`). Table 6.2 shows the statistics of these static datasets. In this work, we have specifically integrated datasets of tweets and claims labelled as misinformation related to COVID-19. As with the ClaimReview data integrated into CimpleKG (Section 6.1.3), we extract the entities and textual factors from the text of these documents. We use the Community Notes Matching, CLEF CheckThat! 2022, COCO (MediaEval 2022), AFP and Propaganda Corpus datasets presented in Section 2.3.2.

Extraction of factors and entities is also performed on the static datasets[18], and then all objects are converted to RDF triples and integrated into the CimpleKG, along with the ClaimReview data. The static datasets represent 6,782,846 triples, totaling around 45% of Cimple KG, and include 624,402 textual factors.

6.1.5 CIMPLE KG Statistics

This section provides statistics about the misinformation data integrated into CimpleKG. These statistics are based on the 11th April 2024 database snapshot.

6.1.6 Fact-checkers and Language Statistics

The current fact-checked data integrated into CimpleKG contains ClaimReview from 77 different fact-checking agencies based in 36 different countries and publishing fact-checks in 26 different languages. As shown in Table 6.3, most fact-checks are published as English (37.7%), followed equally by French and Portuguese (respectively representing 9.1% and 7.8% of the

[18] For news articles, factors are only computed on headline and first paragraph, as those sections contain the most important information per journalistic practice

languages found in the data). However, as displayed in Table 6.4, the country with the most IFCN-registered fact-checking organizations is India (18.2%) followed by France (10.4%) and the USA (9.1%).

Table 6.3: Distribution of ClaimReview languages for the fact-checkers found in continuously updated fact-checkers data.

Language	Amount	Proportion	Language	Amount	Proportion
English	29	37.7%	Croatian	1	1.3%
French	7	9.1%	Danish	1	1.3%
Portuguese	6	7.8%	Dutch	1	1.3%
Spanish	6	7.8%	Filipino	1	1.3%
Hindi	3	3.9%	German	1	1.3%
Italian	3	3.9%	Greek	1	1.3%
Polish	3	3.9%	Indonesian	1	1.3%
Turkish	2	2.6%	Nepali	1	1.3%
Albanian	1	1.3%	Norwegian	1	1.3%
Arabic	1	1.3%	Russian	1	1.3%
Bangla	1	1.3%	Serbian	1	1.3%
Bulgarian	1	1.3%	Serbo-Croatian	1	1.3%
Catalan	1	1.3%	Telugu	1	1.3%

Currently, the fact-checked data integrated into CimpleKG contains 203,209 fact-checks, with most of the reviewed claims identified as *Not Credible* (69.8%) or *Not Verifiable* (15.6%). The remaining claims are identified as *Credible* (6%), *Uncertain* (7.2%) or *Mostly Credible* (1.4%). As displayed in Table 6.5 and Figure 6.3, most of the fact-checks are produced by India (28.6%), followed by the USA (19.9%) and France (15.6%) with AFP fact checking from France producing the most fact-checks (14.4%) followed by Snopes.com from the USA (8%).

6.1.7 CIMPLE KG Use Case and Usage

The CimpleKG dataset has been used in multiple research studies and is integrated into multiple applications:

- *Misinfome Bot* (https://twitter.com/MisinfomeB) is a social media bot that automatically corrects misinformation spreaders by posting fact-checks to known misinformation sharers. The bot uses CimpleKG to identify recent misinformation and fact-checks URLs (Listing 6.1). It was used for understanding the impact of automated misinformation corrections in social media [23].
- *Fact-Checking Observatory* (FCO, https://fcobservatory.org/): The FCO monitored the spread of misinformation and corresponding fact-checks during the COVID-19 pandemic, taking into account their topics, language, and geographic location of fact-

Table 6.4: Top 10 countries with the most fact-checkers.

Country	Amount	Proportion
India	14	18.2%
France	8	10.4%
USA	7	9.1%
Brazil	4	5.2%
Italy	4	5.2%
Poland	3	3.9%
Turkey	3	3.9%
United Kingdom	3	3.9%
Australia	2	2.6%
Portugal	2	2.6%

Table 6.5: Top 10 countries with the most fact-checks.

Country	Amount	Proportion
India	58,49	28.6%
USA	40,468	19.9%
France	31,605	15.6%
Brazil	9,302	4.6%
Portugal	8,928	4.4%
Turkey	8,767	4.3%
Poland	7,244	3.6%
United Kingdom	5,878	2.9%
Italy	3,840	1.9%
Germany	3,342	1.6%

Table 6.6: Top 15 fact-checking organizations with the most fact-checks.

organization	Country	Amount	Proportion
AFP fact checking	France	29,300	14.4%
Snopes.com	United States of America	16,318	8.0%
MMI Online Limited	India	13,416	6.6%
Newschecker	India	11,746	5.8%
BOOM	India	10,267	5.1%
Check Your Fact	United States of America	8,086	4.0%
Lead Stories	United States of America	7,719	3.8%
Pravda Media Foundation	India	7,268	3.6%
Demagog Association	Poland	6,718	3.3%
PolitiFact	United States of America	6,523	3.2%
Polígrafo	Portugal	6,020	3.0%
Full Fact	United Kingdom	5,656	2.8%
Teyit	Turkey	4,607	2.3%
TV Today Network Limited	India	4,485	2.2%
Newsmeter	India	4,237	2.1%

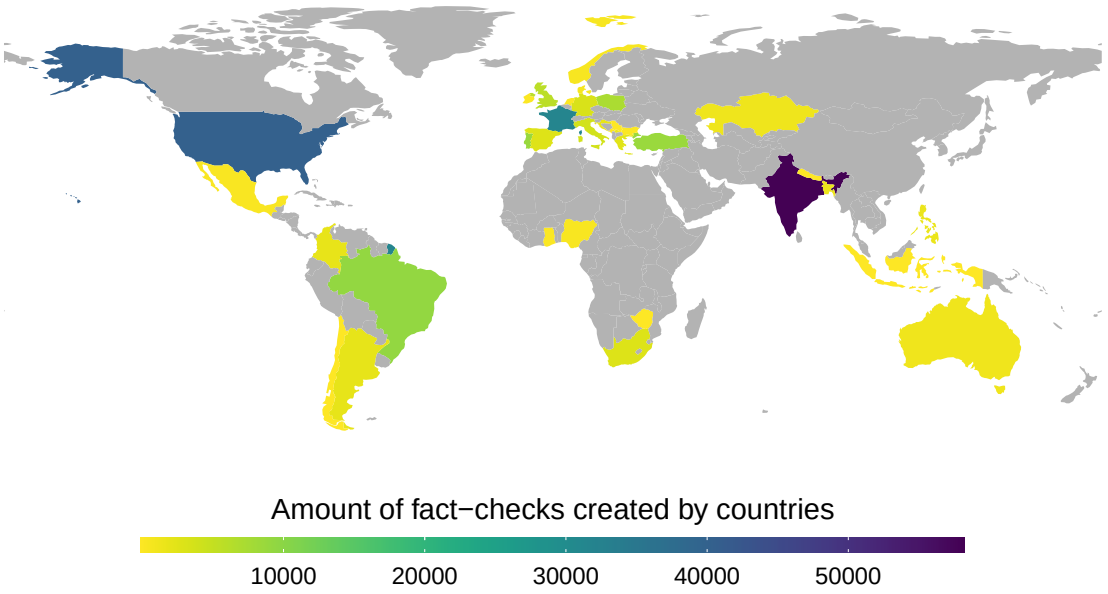


Figure 6.3: Amount of fact-checks created for each country.

checkers. FCO used the pairs of misinformation links and their fact-checks to track their spread on Twitter/X. The FCO data was used for studying the co-spreading relationships between misinformation and fact-checks during the COVID-19 pandemic [20, 21].

- *Iffy Index* (https://iffy.news/index/) is an external website that collects source-credibility assessments from multiple sources, including our Misinformation dataset. Iffy has been used in 24 research papers and several tools.
- CimpleKG was used by a large-scale study that compared fact-checking by experts (ClaimReview) against those done by the crowd (Twitter BirdWatch/Community Notes) [122, 123]. This study relied on the data contained in CimpleKG, and discovered that, in some settings, crowdsourced fact-checks are comparable to those performed by expert fact-checking organizations.
- CimpleKG data was used by the *Linked Credibility Reviews* system [30] and for performing explainable misinformation detection [31, 32]. The authors used CimpleKG data to run their experiments and evaluations.

The KG can be queried using the ontologies described in Section 6.1.2. For example, using the DBPedia ontology and resources, we can obtain the individuals (`dbo:Person`) that are the most associated with Donald Trump (`dbr:Donald_Trump`). We can also easily obtain information about how the original fact-checker ratings are mapped to normalized ratings using the `schema:Rating` type and `schema:sameAs` property. Finding recent misinformation and fact-checks URLs pairs can be performed using the SPARQL query in Listing 6.1. Such a query is used by the *Misinfome Bot* when looking for misinformation spreaders. Additional query examples can be found on the KG data repository.

```
PREFIX sc: <http://schema.org/>
PREFIX co: <http://data.cimple.eu/ontology#>
SELECT DISTINCT ?fc_url ?misinfo_url
WHERE {
    ?rev a sc:ClaimReview ;
         sc:url ?fc_url ;
         sc:datePublished ?date_published ;
         co:normalizedReviewRating ?rating ;
         sc:itemReviewed ?claim .
    ?claim a sc:Claim ;
           sc:appearance ?misinfo_url .
    ?rating sc:ratingValue "not_credible" .
    FILTER (?date_published >= xsd:date("2024-03-11")) .
}
ORDER BY DESC(?date_published)
LIMIT 10
```

Listing 6.1: SPARQL Query used by the Misinfome Bot for retrieving the 10 most recent *not_credible* fact-checks and misinformation URL pairs published since the 11th of March 2024.

6.2 Cimple KG Explorer

Another use-case of the Cimple KG is the exploratory Search Engine[19]. This tool allows querying the graph with entities, factors, dates etc. Figure 6.4 shows the web interface, where we can either directly browse for keywords, or open the detailed view. The detailed view (Figure 6.5) propose ways to filter the data: type, textual search, entities, veracity, language, fact-checking organizations, date, conspiracy theories, persuasion techniques, political leaning, sentiment, emotion, and sub-graph. The right side of the page presents the remaining documents after the filtering and changes dynamically depending on the selection. The user can interact with the documents by clicking to obtain more details about the specific item. On Figure 6.6, we can see a screenshot of the document view. This view showcases the text, the factors detected (emotion, sentiment, conspiracy theories, etc.) and the entities mentioned in the text. The user can also interact with these items to search for documents related to the item.

[19] https://explorer.cimple.eu/

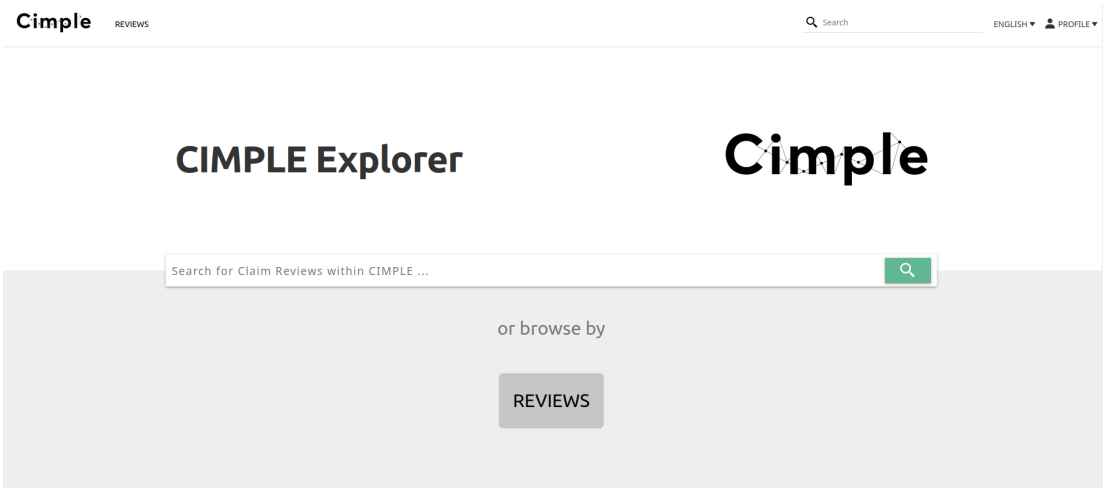


Figure 6.4: A screenshot of explorer.cimple.eu, an exploratory search engine to browse Cimple KG

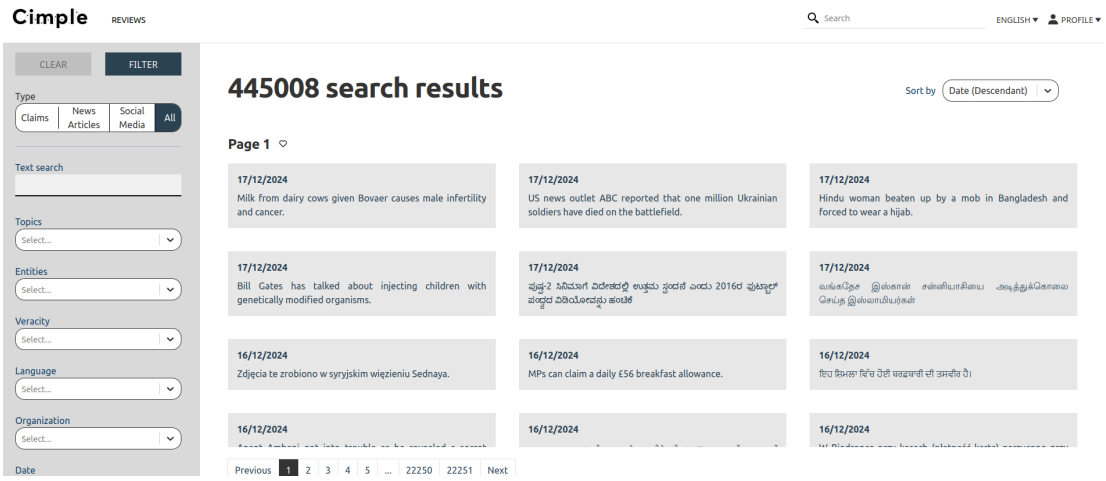


Figure 6.5: A screenshot of the detailed view of the Cimple KG explorer

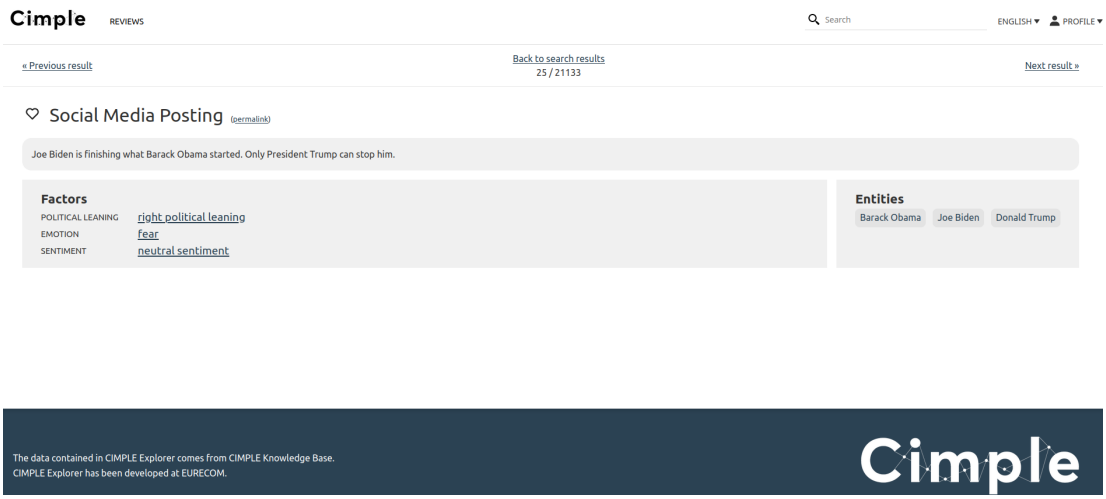


Figure 6.6: A screenshot of the document view in the Cimple KG explorer

6.3 Conclusion

In this chapter, we present Cimple KG, a continuously updated knowledge graph of misinformation-related documents. This resource helps piece together the fragmented documents around the fact-checking ecosystem, directly addressing RQ1. It also contains the extracted factors and entities from these documents, enriching the graph for different applications.

Cimple KG contains over 15 million RDF triples that describe more than 220k fact-checked claims and 210k documents from static misinformation datasets. It also contains more than 250k distinct entities and 1M textual features to further describe the ingested documents and claims. Cimple KG is freely available and has already been used by numerous studies and tools and is continuously updated daily.

This chapter answers RQ1 by introducing Cimple KG, a powerful open resource for any misinformation-related research. The modular aspect of the KG allows for easily deployable upgrades, such as new factors, or ingest new static datasets, which could be tackled in future works.

Chapter 7

Novel Textual Similarity Concepts

In this chapter, we explore novel textual similarity concepts, in the context of misinformation research. First, we introduce a notion of similarity in narratives (Section 7.1) and a notion of similarity for entities (Section 7.2). We then present and compare automatic approaches to retrieve similar documents in Section 7.3. Lastly, we introduce a notion of similarity for documents with different granularity (Section 7.4).

7.1 Fact-Checking Similarity - Narrative vs Implication

In this section, we study how similarity measures could be used in the context of automatically retrieving documents to fact-check social media posts. The retrieval of claims that can fact-check posts could help fact-checking organization save time. We propose a matching dataset of pairs of claims/tweets on the notion of fact-checking similarity. We discuss in Section 7.3 automatic approaches that use this dataset.

We share an example to better understand the need of similarity measures for retrieving documents based on narrative, or fact-checking implications. If we consider the following document:

Tweet: Totally unacceptable. We need a full forensic audit of the 2020 election in Georgia & Brad Raffensperger's resignation. NOW!

We might want to retrieve documents that help fact-checking the claim that the 2020 United States presidential election results have been arranged, such as:

Fact-checked Claim: A signature audit of absentee ballots in Georgia must be conducted and could overturn the 2020 presidential vote results in that state.

We might also want to retrieve documents that share the narrative that elections can be manipulated by third parties, such as:

`News Headline: The extent of fraud in Russia's presidential election begins to emerge. Vladimir Putin's record-breaking 87% of the vote is suspected to have been massively falsified.`

This example showcases the need for similarity measures tailored for different applications. Next, we describe our experiments defining the 'Implication' and 'Narrative' similarities.

First, we define the notion of 'Implication match' between a claim and a tweet as: "Assuming the information used to fact-check the claim is available, this information would be enough to fact-check the tweet, or one of the claim it makes". This notion covers the highest notion of similarity in terms of fact-checking. However, we also define 'Narrative match' as a softer similarity between the documents: "Both documents share a similar overall narrative". This allows for annotating documents not similar enough to be considered 'fact-checking' match, but still better than unrelated. The notion of 'narrative match' is implied if the notion of 'fact-checking match' exist between documents. Factors defined in the previous chapters could enrich the similarity measures, for example some narratives can use some tropes or share conspiracy theories. Table 7.1 shows examples of matches.

Three annotators have labeled a dataset of 200 pairs of tweets and claims. The annotation process was split into two stages: first the annotators would label the same 40 examples to refine the process and discuss the hard examples, then they would split the 160 remaining pairs for parallel annotations. The initial agreement between annotators was a Cohen's kappa of 0.59. The annotators have found 26 'fact-checking' matches, and 86 'Narrative' matches.

7.2 Entity-Based Similarity - Entity vs Concept

Here, we describe a similarity measure that focus on entities and concepts in textual documents. Entities are used everywhere on social media and news article, and carry an import role in the way we process information. They represent real world objects, such as people, locations or organizations. We define concepts as broader parent classes of the entities, as described by Wikidata hierarchies. For example, both entities "Joe Biden" and "Emmanuel Macron" can be represented with the concept "President". The goal of this similarity measure is to match documents if they mention similar entities, or concept, even if they appear in different contexts. We also describe in Section 7.3 automatic approaches that use this dataset.

We present the motivation of such approach by showcasing an example. Let us first consider the three following documents from news media, with the headline in bold followed by the first sentence of the article:

Tweet	Claim	Label
President Biden says, “Fight like hell.” He gets praised. President Trump says, "Fight like hell.” He gets impeached.	Donald Trump wrote letter to President Biden stating "Joe, You know I won."	Entity match
Did you know: The CDC recommends pregnant women "do not touch or change dirty cat litter" So according to our government—if you’re pregnant, changing your litter box is too dangerous but taking an experimental vaccine is now fully recommended & endorsed? Got it.	The vaccine is not safe for pregnant women or women planning on becoming pregnant within a few months of taking the vaccine... We are the lab rats."	Implication match, Entity match
Tucker with the shot heard around the world: Yes, there was meaningful voter fraud in Georgia. Here is the hard, verified evidence:	CNN and ABC showed proof of Georgia election fraud on TV.	Implication match, Concept match
Trump says "no reason" for officer to shoot Capitol rioter, pushing conspiracy theory	CNN reported investigators believe riot at Capitol was not inspired by President Donald Trump and was a pre-planned event, plus Congress voted without any information from investigators.	Narrative match, Entity match

Table 7.1: Examples of pairs using the labels defined in 7.1 and 7.2

- D1: **Macron makes ’end of summer’ vaccine pledge to France.** Emmanuel Macron said on Tuesday that all of his countrymen who wanted a vaccine would be offered one "by the end of the summer".
- D2: **Too early to say if summer vacations possible, Macron tells French.** Emmanuel Macron said Tuesday it was too early to say if vacations will be possible this summer, even as the country prepares a gradual lifting of a two-month coronavirus lockdown.
- D3: **Trump expects enough Covid-19 vaccine for every American by April.** Donald Trump said Friday he expects enough Covid-19 vaccines "for every American" to be produced by next April, and that the first doses will be distributed immediately after approval later this year.

In this scenario, a typical use-case would be to find the most similar document from D1. Traditional semantic textual approaches result in a similar score for both documents D2 and D3[1]. However, entities play a key role in the similarity of the documents. Figure 7.1 shows

[1] Sentence-BERT model gives similarity scores around 0.65

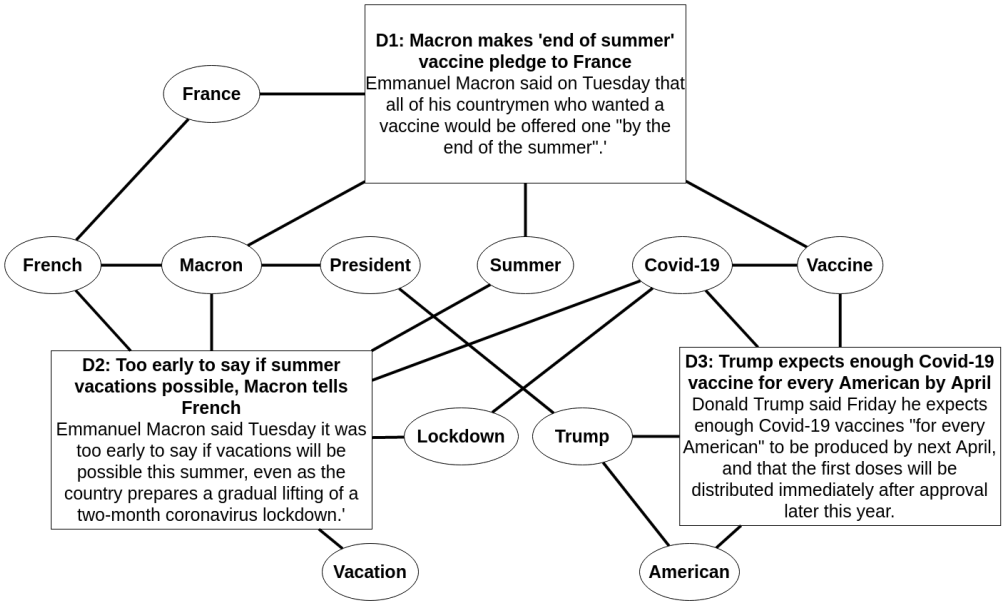


Figure 7.1: Example of a use-case for entity and concept similarity

the graph of entities and concepts extracted from the three documents. In this graph, nodes represent documents, entities and concepts. We create an edge between a document and an entity or a concept if the document mention them. Also, we link entities and concepts together if they are related with a property, such as belonging to a parent class. The example showcases the ambiguity of textual similarity. If the entity 'Macron' is useful for the use-case, then D2 should be more relevant, however D3 is a better match if 'Vaccine' is considered. In this toy example, we show the need for alternative similarity measures in the context of textual document matching.

We refer to entities as they are defined in the NLP community, for tasks such as named-entity recognition or entity linking. We define the notion of 'entity match' if both tweets and claims refer to the same entities. We also defined the notion of 'concept match' for cases where entities are different but their 'type' or 'parent class' is the same. For example, both 'Joe Biden' and 'Donald Trump' would belong to a parent class 'President of the United States'.

Similarly to the previous similarity measure, three annotators have labeled a dataset of 200 pairs of tweets and claims, using the same two stages annotation process. The initial agreement between annotators was a Cohen's kappa of 0.75. The annotators have found 32 entity matches and 62 concept matches. Table 7.1 also contains examples of matches.

The task of annotating similarity for fact-checking and entities proved to be challenging. Consider the following example:

```
Tweet:  When I was growing up, someone putting an American flag on their
```

```
property was a sign of actual patriotism.  Now, it might as well be a
confederate flag.  Racist, bigoted ideology is NOT what being an American
is about.

Claim:  McDonald"s removes their American flags in support of Antifa &
BLM nationwide
```

In this example, the task is not trivial. While it seems clear that tweet cannot be fact-checked with information from the claim (no Implication match), the narrative of both documents is subtle. We argue that both texts use a narrative of using flags as activism tools, to push political ideas (Narrative match). Considering entities and concepts, they are hard to extract in the tweet. We propose the following entities: American flag, confederate flag, America, and the following concepts: patriotism, racism, as well as the ones implied by the entities. In this example, the entities and concepts in the tweet are not enough to match the ones in the claim, which mentions more content such as McDonald's, Antifa or BLM. This example shows that the actual extraction of the entities, and the abstraction needed for the concepts are challenging tasks.

7.3 Comparing automatic approaches for document matching

In this section, we detail and compare automatic approaches to retrieve similar fact-checked claims from social media posts. We will compare two different approaches: Sentence-BERT and Graph-based.

7.3.1 Using Sentence-BERT

A very common baseline when retrieving previously fact-checked claims is using Sentence-BERT to embed the documents, then compute the cosine similarity between them [128]. We use the 'all-mpnet-base-v2'[2] model from the SentenceTransformers python library[3] [119] as it performs the best on average on both sentence embeddings and semantic search tasks.

7.3.2 Using Graphs

Similarly to Cimple KG, we extract the entities and concepts present in all the textual documents. We use an open-source model from the spaCy library [4]. We then build a graph by creating nodes for each document (claims and tweets) and each entity and link the document to the entities it mentions, similarly to Figure 7.1. For these experiments, we do not re-use the

[2] https://huggingface.co/sentence-transformers/all-mpnet-base-v2
[3] https://www.sbert.net/
[4] https://github.com/egerber/spaCy-entity-linker

Method	MRR	Acc@1	Acc@5	Acc@10	Acc@50	Acc@100
Sentence-BERT	0.844	0.808	0.885	0.923	0.962	1
Graph-approach	0.440	0.346	0.538	0.615	0.692	0.808

Table 7.2: Results for retrieving Fact-checking matches

Method	MRR	Acc@1	Acc@5	Acc@10	Acc@50	Acc@100
Sentence-BERT	0.783	0.744	0.814	0.884	0.953	0.977
Graph-approach	0.401	0.349	0.442	0.488	0.628	0.721

Table 7.3: Results for retrieving Narrative matches

Cimple KG, but rather focus on a small subset of data, as a proof of concept.

After creating the graph, we use the node2vec algorithm [53] to create embeddings for each node in the graph. The first step is to simulate multiple random walks from every node in the graph. Starting from a node, the walk hops to a connected node with a uniform probability. In the next iterations, the walk has different probabilities for the adjacent nodes depending on two parameters p and q. p controls the probability of returning back to the previous node and q controls the probability of visiting a node that was not in the neighborhood of the previous node (exploration). Random are composed of N nodes, and are repeated D times for every node. These walks are then used to train a word2vec model [90] (skip-gram), where nodes replace the words. Once trained, this model generates an embedding for every node. The parameters p and q will have an impact on the random walks, changing the training of the model and thus the embeddings.

7.3.3 Comparison of the approaches

Both Sentence-BERT and the node2vec algorithm generate an embedding for each document in the dataset. We are comparing the approaches by using them to rank all claims from every posts and comparing the average ranks of the ground truth.

We can see on the Tables 7.2, 7.3, 7.4 and 7.5 the matching results using both Sentence-BERT and the Graph-based approach. These results show that Sentence-BERT is much better at retrieving the correct document. The graph-based approach retrieves entity match the best, this is due to the presence of entities extracted from textual documents in the graph.

Method	MRR	Acc@1	Acc@5	Acc@10	Acc@50	Acc@100
Sentence-BERT	0.732	0.688	0.750	0.781	0.938	0.969
Graph-approach	0.578	0.469	0.656	0.781	0.844	0.906

Table 7.4: Results for retrieving entity matches

Method	MRR	Acc@1	Acc@5	Acc@10	Acc@50	Acc@100
Sentence-BERT	0.765	0.726	0.790	0.839	0.935	0.968
Graph-based	0.465	0.403	0.500	0.565	0.645	0.758

Table 7.5: Results for retrieving concept matches

Another advantage of using Sentence-BERT is the ability to embed new documents. Indeed, the model can embed any textual documents as long as it fits the 512 input token limit, and can then be compared to other documents using cosine similarity. For the graph-based approach, since it relies on node2vec, which uses random walks over the graph, adding new document change the embedding of all neighbor documents. This means having to re-compute the embeddings for each new document added to the graph, which can be resource intensive and time-consuming. However, this graph-approach does not have a limit of tokens in input, as the entity extraction tools work on documents of arbitrary length.

7.4 Comparing Documents With Different Granularity - Long vs Short

A common problem one might encounter while using transformer-based models is the limit of the input length. Indeed, most models (BERT, RoBERTa, etc) have a maximum length of input tokens of 512. This limit is easily reached if the input text is long, such as for news articles. While most of the useful information of a news article is stored in the headline or the first paragraph (the “Lede” or “lead paragraph”), the full article still contains more information inaccessible to a standard BERT model. Also, comparing two documents with different granularity is difficult because the notion of similarity is not defined for such case. In our experiments, we will consider the comparison between a long document (e.g. a news article) with a short document (e.g. a tweet, or a claim). For example, a news article about school closures during COVID-19 in the New York state might mention that America’s coronavirus epicenter is New York, that the state has more than 1,500 public schools, that the number of COVID deaths in the state is high or that the governor doesn’t know how long schools will be closed for. All those topics can be relevant based on the interests of the reader. In a claim-verification scenario, retrieving relevant documents can include documents fact-checking the number of COVID cases in New York, the number of schools, or the speech made by the governor. This example showcase the dense nature of news articles and the difficulty in retrieving relevant similar documents.

7.4.1 Data

We consider two datasets for our experiments: AFP and Birdwatch defined in Section 2.3.2. We only consider claims from the Birdwatch dataset, representing 'short' documents (mostly one-sentence length), while news articles in the AFP dataset represent 'long' documents. We will consider around 1.2k claims and 20 news articles. The articles are first split into shorter chunks[5] in order to be compared to short claims. The articles were picked at random over a selection of more than 200k articles.

7.4.2 Method

We create a ground truth dataset by handcrafting claims that are similar to the selected news articles. For each article, we write three types of claims: i) generic, ii) specific and iii) hybrid. Generic claims are similar to the entire article, and leverage the global information of the article. Specific and hybrid claims are about a selected chunk of the article. Specific claims must only use information contained in the chunk, without focusing on information contained outside the chunk, while hybrid claims focus on the chunk but may also introduce information from the rest of the article. Those claims are then merged to the Birdwatch claim dataset. The goal is to rank all claims by 'similarity' for each article and retrieve consistently the corresponding claims.

Our method relies on semantic textual similarity scores and extends the concept to longer documents. First, we compute the embeddings of all claims and all news article chunks using the Sentence-BERT model [119]. Then we use Equation 7.1 to compute the embedding of the article. This equation showcase a coefficient x, which emphasize local information when close to 0 and global information when close to 1. Lastly, we compare the embeddings of the articles with the embeddings of the claims using cosine similarity, computing a ranking for each claim. The pipeline can be seen on Figure 7.2.

$$Emb(Article, x, i) = \frac{1}{n} \cdot \left(x \cdot \sum_{j=0}^{n} Emb(chunk_j) + n \cdot (1-x) \cdot Emb(chunk_i) \right) \tag{7.1}$$

[5] In our experiments, the chunk size is equal to a sentence

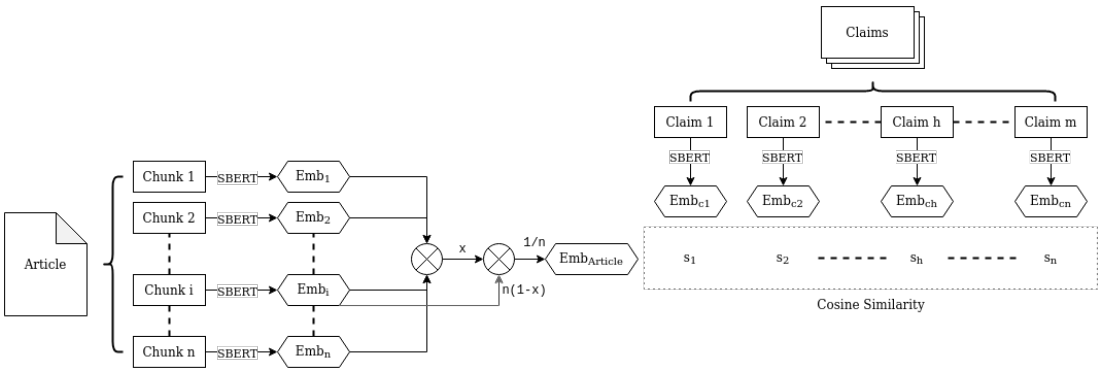


Figure 7.2: Pipeline for computing similarity between short and long documents, using local-global coefficient x and chunk i.

With:

$$n = \text{the total number of chunks in the article, } n \in \mathbb{N}$$
$$i = \text{the index of the chunk in the article, } i \in [0, n]$$
$$x = \text{the local-global coefficient, } 0 \leq x \leq 1, x \in \mathbb{R}$$
$$Article = \{chunk_i | i \in [0, n]\}, \text{ the news article composed of chunks}$$

7.4.3 Results

We compute the results on our dataset, by creating different embeddings for each article, depending on the local-global coefficient x and the selected chunk in the ground truth. We then rank all the claims for every article embeddings and report the average rank of the three types of claims (generic, specific, hybrid) handcrafted for the article. We plot the results in Figure 7.3.

We can see that the local-global coefficient has a major impact on the retrieved claims. As we can see, when x is low, the average rank of the specific claim is lower than the one of the generic claim. However, this tendency is reversed as x increases, as the embedding of the article is computed using more and more global information. We can also see the hybrid claim has the best average rank across all values of x, which means that both local and global context are helpful.

7.5 Conclusion

In this chapter, we explored new approaches to measure textual similarity within the context of fact-checking applications, a central focus of our efforts in addressing RQ3. By introducing distinct similarity measures—fact-checking/narratives, entity/concept, and long/short

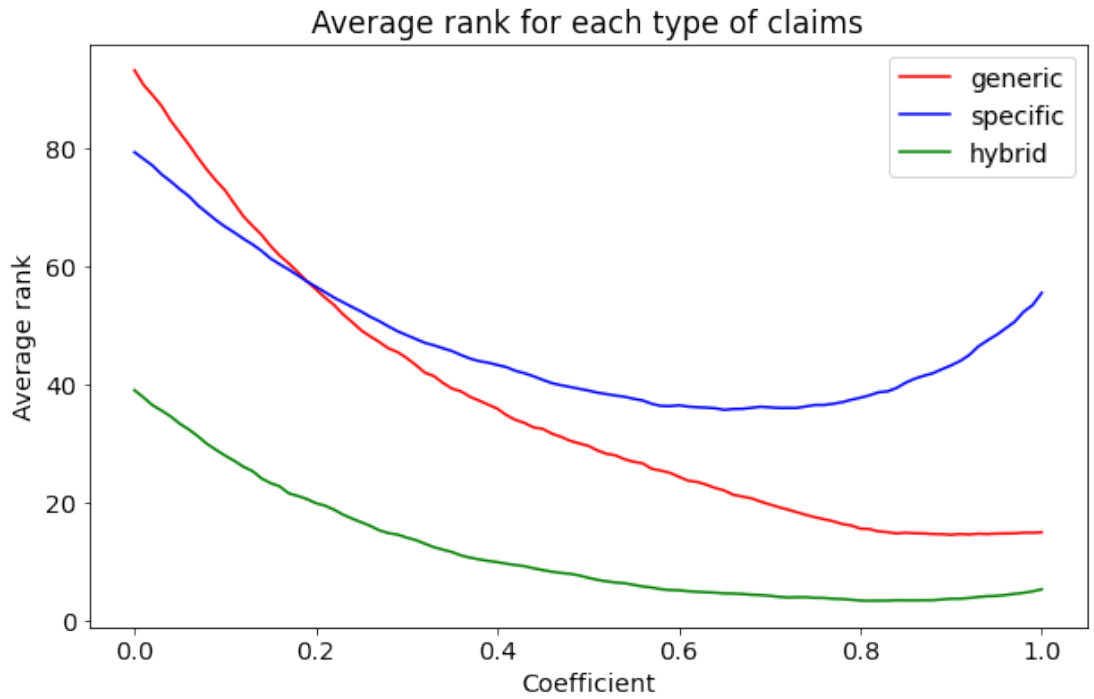


Figure 7.3: Average rank of the different types of claims depending on the coefficient x

document comparisons, we have systematically examined the role of textual similarity in enhancing the efficiency and effectiveness of fact-checking processes.

Our findings highlight that traditional semantic textual similarity measures often fall short in the nuanced landscape of misinformation. However, by developing specialized measures tailored to the demands of fact-checking, we could improve the retrieval of previously fact-checked claims. The datasets we created, which annotate the various levels of similarity, provide a valuable resource for new research. The ability of Sentence-BERT to handle both narrative and entity-based similarity demonstrates how leveraging advanced NLP techniques can directly contribute to automation in combatting misinformation. Furthermore, our exploration into comparing documents with varying granularity show that both local and global contexts must be integrated for more accurate and contextualized claim retrieval.

Ultimately, this chapter not only answers RQ3 through the creation and validation of novel similarity measures but also open directions for future research. Subsequent work could explore refining these approaches further, for example by incorporating adaptive algorithms that could dynamically respond to the evolving nature of misinformation.

Chapter 8

Conclusion and Perspectives

As misinformation becomes more prominent in our means of information, it becomes necessary to counter it. While manual fact-checking is absolutely essential in reducing the spread of misinformation, it has trouble scaling-up to the need. This thesis introduces tools and resources to help fact-checkers in their fight against misinformation. We share automatic approaches, useful resources and methods to better understand online textual documents, and what role they play in misinformation spread. In this chapter, we conclude the thesis by summarizing the contributions, and discuss how they answer the research questions introduced earlier in the Thesis. We also present the Future Works that could improve our work further.

8.1 Conclusion

In this work, we introduce tools to better understand, detect and explain the spread of misinformation on the web. We use language models and knowledge graphs to extract textual features and represent data. Our contributions span topics such as automatic textual features detection, knowledge graph data model and curation and textual similarity measures. In the following section, we cover the answers to the research questions defined in Chapter 1.

We tackle the first research question "**RQ1**: How to model relationships between all the different types of data used for fact-checking" in Chapter 6, where we introduce Cimple KG, a knowledge graph of misinformation-related documents. This KG is composed of news articles, social media posts, claims, memes, claim reviews, etc. Documents are linked through their mention of entities, their use of *factors*, or other metadata. This creates a rich and dense knowledge graph that can be used for many applications related to fact-checking. Cimple KG is the largest up-to-date misinformation-focused semantic resource, containing data from 77 fact-checking organizations and data from several static datasets. Currently, CimpleKG contains over 15 million RDF triples that describe 203,209 fact-checked claims and 217,616

documents from static misinformation datasets. It also contains 263,243 distinct entities and 1,121,584 textual features to further describe the ingested documents and claims. CimpleKG is freely available, has already been used by numerous studies and tools and is continuously updated daily. This resource not only model the relationships between all the different types of data used for fact-checking, but also integrates the many features of interest of the documents.

Considering "**RQ2**: How to better understand textual documents with the use of automatic approaches.", we use language models in Chapter 3 and 5 to detect textual features in documents. We train multiple CT-BERT models and create an ensembling model to detect conspiracy theories in tweets, reaching-state-of-the-art performance in MediaEval Fake News Detection challenges. During the SemEval challenge, we also trained multiple BERT-based models to detect persuasion techniques in memes. Additionally, making use of the hierarchical structure of persuasion techniques by creating a custom loss function and allowing the model to output parent classes improved the performance the most. While most of our best performing approaches at detecting *factors* rely on BERT-based models, we also experiment with LLMs. We use GPT-3.5, LLama-2 and Zephyr-β to annotate conspiracy theories and propaganda techniques. We find that the quality of the definition of the class in the prompt will have a significant impact on the classification results. In doing so, we propose a method to generate definitions based on examples. Furthermore, we also train BERT-based models to extract emotion, sentiment and political-bias in social media posts. These models are then used to analyze the online social discourse on Twitter regarding COVID-19. Insights show that some topics are controversial and that online political bias aligns with the stance of US politicians on those topics. We also define a novel feature, "*Tropes*". Tropes are easily recognizable devices used in narratives to convey specific themes or ideas. We define 9 online tropes and annotate them in Tweets on vaccination and immigration topics. We then show baseline models to automatically detect them. We also show that tropes are different from conspiracy theories or persuasion techniques, and they represent another dimension to consider when analyzing misinformation discourse.

In Chapter 7, we tackle different novel similarity measures, relying on entities, fact-checking scenarios and comparing document with different lengths. This answers "**RQ3**: How to extend the notion of textual similarity for fact-checking applications", as it helps retrieve previously fact-checked claims in different contexts. We first create datasets to annotate fact-checking, narrative, entity and concept matches, and propose two methods to automatically retrieve them. We find that Sentence-BERT performs better at retrieving all the kinds of matches. Then, we propose methods to compare documents with different length. Precisely, we compare one-sentence claims with news articles. As we target chunks of the news article, we experiment with different local vs global information trade-offs. We annotate data to validate our methods, and show that we can reliably retrieve the correct claims in the correct settings. These studies on similarity measures show notion of textual similarity have room for additional features for

practical uses.

8.2 Perspectives

The list of textual *factors* that can be used to explain misinformation spread could be increased. For example, online hate speech is closely related to misinformation [71], especially during the COVID-19 pandemic, when some misinformation campaign towards the Chinese government in the origin of the COVID-19 virus lead to anti-Asian speech. The detection of such feature would add insights to the analysis of the spread of misinformation. Also, the analysis of the correlations between such feature and the one already analyzed would add insights on how misinformation spreads. Additionally, most of our models are trained on tweets, and perform well on this type of data. However, they are used at inference on out-of-domain data, such as news articles, impacting the performance. Training on more diverse data would make the models more robust to changes in the domain.

Our work on Cimple KG is modular, which makes it easier to add new resources and factor detection. For example, we can add to our pipeline new models to detect additional features (*Tropes* for example), or new static datasets coming from the literature (climate change misinformation related for example). Also, a significant amount of data collected from the Google Fact-checking API contains malformed ClaimReviews. While we already try to fix most errors automatically, we drop around 30% of the DataCommons fact-checks due to the quality assurance issues. Improving our methods to resolve malformed ClaimReviews when possible would increase the number of documents in the graph.

Regarding experiments generating definitions from LLMs, we discuss multiple limitations and future works. While classification of conspiracy theories using EG definitions is done in a zero-shot fashion, the generation of the definitions still relies on annotated examples. This is different from standard in-context few-shot classification, as these examples do not need to be part of the classification prompt. Indeed, it can be seen as a way to compress the information from few-shot examples into a shorter descriptive context that can be appended to the zero-shot prompt. Further experiments could explore this approach. The correlation tests of definition understanding in Section 4.3.2 support the claim that GPT-3 can indeed interpret and apply the definitions correctly. This is complemented by the results in Section 4.3.1 which show that better definitions lead to better results. However, further testing should be done on more LLMs and with other corpora. As the field of LLMs is changing rapidly, with the release of novel models and prompting techniques, our study could be updated with the most recent tools available. Moreover, we struggled to improve the zero-shot results for the propaganda technique classification experiments. This shows the limitation of our method, as human-written definitions do not provide significant performance boost. Our understanding is that the model is already aware of the task, as it was published long before the training of the

model. However, this reveal interesting insights about LLMs and could be studied further on other tasks, on Tropes detection for example. Such experiments are an interesting direction for future work, with the potential to shed light on the semantic capabilities of LLMs. Several practical recommendations and potential applications can be found in Appendix A.2.3.

The work on similarity shows some limitations, as the graph approach performs poorly compared to Sentence-BERT to retrieve the correct documents. One improvement could be to use a more dense graph, with more textual features extracted and not only entities. Also, we could explore hybrid approaches that leverage both the graph and sentence-BERT. As of now, Sentence-BERT retrieves document better, but the graph-approach could be used for additional explanation, as similarity-scores are not explainable. Considering the comparison of documents with different length, we show promising results on a smaller dataset of 20 news articles. However, this dataset needs to be extended to make the results more robust. Having more samples in the data would also allow error-analysis to understand where our methods perform well, and where they struggle.

Chapter A

Appendix

A.1 Appendix of Chapter 3

For reproducibility, we share the exact prompt used to generate new examples using GPT-4-Turbo (as of January 2024):

[system] Your task is to generate short sentences that contains the <current_propaganda_technique> propaganda technique.
The definition of the <current_propaganda_technique> propaganda technique is the following: <current_propaganda_technique_definition>

Here are some examples:
– <Random example x5>

([user] Please generate a short sentence that contains the <current_propaganda_technique> propaganda technique similar to the examples, on similar topics.
[assistant] <Random example>) x5
[user] Please generate a short sentence that contains the <current_propaganda_technique> propaganda technique similar to the examples, on similar topics.

A.2 Appendix of Chapter 4

A.2.1 Examples of Definitions

Definitions of Suppressed Cures Conspiracy Category

Example Generated (Seed 0) The definition of the concept is a conspiracy theory that suggests the existence of a deep state that is orchestrating the COVID-19 pandemic and blocking the release of Hydroxychloroquine, a cure for the virus. This theory also involves the belief that the pandemic is being used to push liberal agendas, create economic recession, help China's economy, and stop Trump rallies. It is often associated with the QAnon movement and involves the idea that Dr. Fauci is a Deep State Killer.

Example Generated (Seed 1) The definition of the concept is the use of hydroxychloroquine (HCQ) as a possible treatment for COVID-19.

Human Written Suppressed cures: Narratives which propose that effective medications for COVID-19 were available, but whose existence or effectiveness has been denied by authorities, either for financial gain by the vaccine producers or some other harmful intent.

Definitions of Harmful Radiation Conspiracy Category

Example Generated (Seed 4) The definition of the concept is a conspiracy theory that claims that the COVID-19 pandemic is not caused by a virus, but rather by the implementation of 5G technology and that the pandemic is being used as a distraction to implement a digital currency and implantable microchips for population control. This theory has been widely discredited by scientific evidence and health organizations.

Example Generated (Seed 1) The definition of the concept is a conspiracy theory that claims that the COVID-19 pandemic is caused by the deployment of 5G wireless networks. This theory suggests that the radiation from 5G towers weakens the immune system, making people more susceptible to the virus, or that the virus is somehow being transmitted through the 5G network. Despite being debunked by scientists and health experts, this theory has gained traction among some individuals and has led to the destruction of 5G towers in some areas.

Human Written Harmful radiation: Narratives that connect COVID-19 to wireless transmissions, especially from 5 G equipment, claiming for example that 5 G is deadly and that

COVID-19 is a coverup, or that 5 G allows mind control via microchips injected in the bloodstream.

Definitions of Depopulation Conspiracy Category

Example Generated (Seed 2) The definition of the concept is depopulation, which refers to the deliberate reduction of the world's population through various means such as disease, vaccines, and other methods. The tweets in the first set suggest that there is a group of people, including Bill Gates, who are actively working towards depopulation as part of their agenda. The tweets in the second set, on the other hand, do not mention depopulation and focus on other topics related to COVID-19 such as conspiracy theories, mental health, and political issues.

Example Generated (Seed 3) The definition of the concept is depopulation, which refers to the deliberate reduction of the human population through various means such as disease, sterilization, or other forms of population control. The tweets in the first set suggest that some individuals believe that the COVID-19 pandemic and the development of vaccines are part of a larger depopulation agenda orchestrated by individuals such as Bill Gates. These beliefs are not present in the second set of tweets, which focus on other topics related to COVID-19 such as its impact on mental health, the effectiveness of vaccines, and political responses to the pandemic.

Human Written Depopulation: Conspiracy theories on population reduction or population growth control suggest that either COVID-19 or the vaccines are being used to reduce population size, either by killing people or by rendering them infertile. In some cases, this is directed against specific ethnic groups.

Definitions of Satanism Conspiracy Category

Example Generated (Seed 0) The definition of the concept is the belief in a conspiracy theory that the COVID-19 pandemic is part of a larger plan orchestrated by a group of powerful individuals or organizations, such as the Illuminati, the New World Order, or the Deep State, to exert more control over the population and ultimately lead to the implementation of the "mark of the beast," which could be a microchip or a vaccine that would be mandatory and used to monitor and control people's actions and movements. This belief often involves the idea that the pandemic is a hoax or exaggerated, and that the real agenda behind it is to push for a global government and depopulation.

Example Generated (Seed 3) The definition of the concept is the belief that the COVID-19 vaccine or any other form of mandatory vaccination is the "Mark of the Beast" as described in the Book of Revelation. This belief is often associated with conspiracy theories involving the government, deep state, and Luciferian Freemasons who are seen as trying to control and enslave the population through the use of tracking chips and microchipped vaccines. The concept is rooted in religious and apocalyptic beliefs and is often used to justify opposition to vaccination and other public health measures.

Human Written Satanism: Narratives in which the perpetrators are alleged to be some kind of satanists, perform objectionable rituals, or make use of occult ideas or symbols. May involve harm or sexual abuse of children, such as the idea that global elites harvest adrenochrome from children.

A.2.2 Prompt Description

Example Prompt for EG Definitions

```
SYSTEM = "You will be given two sets of tweets. The first set of tweets
    contains examples of texts that mention the same concept. The second
     set of tweets contains examples of texts that mention other
    concepts, but not the same concept that tweets from the first set.
    Your task is to provide the definition of the concept present in the
     first set"

USER = "First set of tweets:
[25x Tweets containing the conspiracy]

Second set of tweets:
[25x Tweets not containing the conspiracy]

Given those two sets of tweets, what is the definition of the concept
    present in the first set that is not present in the second set of
    tweets? Start your answer with: 'The definition of the concept is' "
```

Example Prompt for annotating a Tweet with regard to a conspiracy theory

```
SYSTEM = "Your task is to label tweets regarding the '[CONSPIRACY]'
    COVID-19 conspiracy theory. The available labels are: 1) mentions
    the conspiracy, 2) does not mention the conspiracy.
```

```
The definition of the '[CONSPIRACY]' conspiracy theory is the following:
[CONSPIRACY definition]"

USER = "[TWEET]

Does the tweet: 1) mention the '[CONSPIRACY]' conspiracy, 2) do not
    mention the '[CONSPIRACY]' conspiracy? Please include the
    corresponding number in your answer."
```

A.2.3 Recommendations For Practical Use

In this section, we elaborate on some recommendations for applications of definition-based zero-shot classifiers. These recommendations are mainly motivated by the classification results from Section 4.3.1.

Fixing the class imbalance for labeling Recall of the definition-based zero-shot classifiers is high and comparable to the recall of the fine-tuned model. Therefore, a possible application of such classifiers is the selection of text data for labeling, with the goal of fixing the class imbalance, i.e., increasing the expected proportion of positive examples. This approach could help mitigate the rarity of positive examples in many text classification use-cases, such as various misinformation detection scenarios.

Correcting annotation errors Another potential application of the definition-based zero-shot classifiers is detecting and correcting annotation errors. The approach we propose is to perform error analysis of the classifiers based on human definitions, which are commonly used for text annotation. As suggested by low precision scores (see Table 4.1), the number of false positives is high – on average 145.89 texts per category for the test set of 830 texts. However, the number of false negatives is lower and more tractable (on average 27.11 texts per category). Additionally, high recall implies that the texts tend to be correctly detected as non-conspiracies, so the false negatives also seem more likely to identify examples wrongly annotated as conspiracies.

Our preliminary analysis indicates that this is indeed the case. We randomly selected 5 false negative texts per category and checked the annotations using the category definitions from [75]. We found, on average, 3.8 labeling errors per category (76% of inspected texts).

Mitigating the low precision The classification results in Table 4.1 show that the definition-based zero-shot classifiers suffer from low precision. This means that there is a high occurrence

of false positives – texts belonging to other related categories being recognized as adhering to the definition of the category being classified. A possible remedy for this could be to upgrade the category definitions with text explicitly excluding similar categories.

Example-generated definitions use cases for few-shot learning An interesting use-case of EG definitions is the fact that they serve as a way to encode a lot of information into a shorter paragraph. Indeed, the LLMs can provide a descriptive definition of the task from a set of examples. This way, rather than providing all the examples each time we want to annotate a sample, we can provide a much shorter context, allowing to reduce the prompt size, and thus the cost, significantly.

Also, the quality of the definition matters, meaning we can actually use a more powerful model (such as GPT-4) to generate the definition, but still use a cheaper model to run the annotation (such as GPT-3.5-turbo). This allows to annotate large amount of data with a higher-quality definition without increasing the cost by much.

A.3 Appendix of Chapter 5

A.3.1 Reproducibility

To encourage reproducibility of our experiments, we share our code at: https://anonymous.4open.science/r/ADTIST24-768D.

For the training of our Bert-FT and CovidBert-FT models, we used one Tesla K80 GPU. Training time is around two hours. We used the following hyper-parameters: batch size of 12, learning rate of $2 \cdot e^{-5}$, 20 epochs, AdamW optimizer, weight decay of 0.01. We use a Cross-Entropy Loss weighted with the inverse frequency of the class sample. We split the dataset into 80% training and 20% validation using a stratified split (according to the nine tropes).

We used the API provided by OpenAi to prompt the gpt-3.5-turbo-0125 model used in our experiments. We used the Llama-3-8B-Instruct model locally on a NVIDIA GeForce RTX 3090 GPU, with an inference time of around 4 seconds per annotation. For both GPT and Llama experiments, we use the following prompt:

```
The task is to label some texts according to these definitions:

Skepticism Towards Authority (STA): The text appeals to skepticism
   towards science and scientific experts or towards political
   authorities, featuring narratives such as 'authorities have failed
   now and before', 'this political party does not know what they are
```

doing' (I know better than experts; They should know better; They don 't know what they are doing).

Defend The Weak (DTW): The text emphasizes the negative effects of something on vulnerable populations, e.g. children (it is especially harmful to the weak; I must protect the weak; They are putting the weak in danger).

Hidden Motives (HM): The text alludes to underlying agendas, suggesting that something is secretly promoted by individuals with malicious intentions (such as hypocrites and tyrants) and concealed motives (There is clearly an untold story behind it; I am being lied to; They are trying to hide their real motives).

Liberty, freedom (LF): The text emphasizes personal autonomy and rights (my body, my choice; I should be able to do what I want; They are forcing on me something I don't want; people were stripped of their rights, jobs, freedom and forced against their will).

Natural Is Better (NIB): The text promotes the idea that natural or traditional approaches are superior, with assertions like 'natural immunity is the best immunity' and 'natural/traditional solutions are more effective and secure' (I think natural solutions are more effective; The other solutions put us in danger).

Time Will Tell (TWT): The text appeals to the eventual validation of one's argument over time and asserting foresight (I know what is gonna happen; I knew it was gonna happen; They don't see the problem coming).

Too Fast (TF): The text implies that something is unsafe or unreliable because it is experimental, untested, developed too quickly ('haste makes waste'), or not yet fully approved by authorities (I currently don't feel safe without more evidence; They rushed the decision, it's dangerous).

Scapegoat (SC): Text that attributes blame or responsibility for a problem to a person or entity not directly involved, such as 'They claim it's A, B, or C's fault, but it's really X's fault' or assigning responsibility for an issue to a famous or popular entity, such as Bill Gates (I think this group of people/entity is to be held responsible; They are the biggest/only problem).

Wicked Fairness (WF): Text that hints to the fact that someone is receiving something they do not deserve, pointing to the unfairness of the situation (something feels unfair about one group of people/ entity; They should receive the same treatment as someone else).

Model	TPMR			STA			TF			NIB		
	V+I	V	I	V+I	V	I	V+I	V	I	V+I	V	I
Bert-V+I	0.33	0.20	**0.43**	0.54	0.58	0.0	0.75	0.75	0.0	0.50	0.52	0.0
Bert-V	**0.35**	0.31	0.40	0.58	0.60	0.0	0.76	0.76	0.0	0.52	0.55	0.0
Bert-I	0.17	0.14	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CovidBert-V+I	0.27	0.18	0.36	**0.60**	**0.63**	0.0	0.77	0.77	0.0	**0.55**	**0.57**	0.0
CovidBert-V	0.30	**0.36**	0.22	**0.60**	0.62	0.0	**0.78**	**0.78**	0.0	0.46	0.48	0.0
CovidBert-I	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
# support	16	8	8	45	42	3	28	28	0	13	12	1

Table A.1: F1-score per class on subsets of the test data. Models are trained on different subset of the train set. V stands for Vaccine and I for Immigration.

```
None: Texts that do not fit clearly into any trope category.

No other labels are allowed if you think the text should be labelled as
   a None. Labels are not mutually exclusive, there can be up to three
   but not necessarily.
```

A.3.2 Data Collection

In constructing our dataset, we have focused on Twitter posts, which are publicly available data. We have removed personally identifiable information from the dataset, and the content of the post has been stripped of links; however, no profanity filter has been applied. The Twint Python library has been used for data collection. The library scraped, just by querying the keyword “vaccine”, “migrant”, “migration” and “asylum” filtering for results in the English language, resulting in about 50k tweets. We collected tweets posted throughout the 26th and 27th of June 2022 for the “vaccine” keyword and late November 2023 for the immigration domain. However, for the immigration topic, we realized soon enough that posts were too similar one another: thus, to have a less biased dataset as possible and to avoid a consequent bias in the training process, we went back in time to retrieve data since late 2019 up to 2022, and then chose about 300 tweets from each year. The vaccine domain did not encounter the same issue.

A.3.3 Additional Experimental Results

We share additional results, about the difference between Vaccine and Immigration topics on the training of supervised models in Table A.1, A.2 and A.3, as well as correlations between Tropes on Vaccine and Immigration subsets in Figures A.1 and A.2. We train Bert and CovidBert models on subset of the training set: full dataset (V+I), Vaccine only data (V) and Immigration

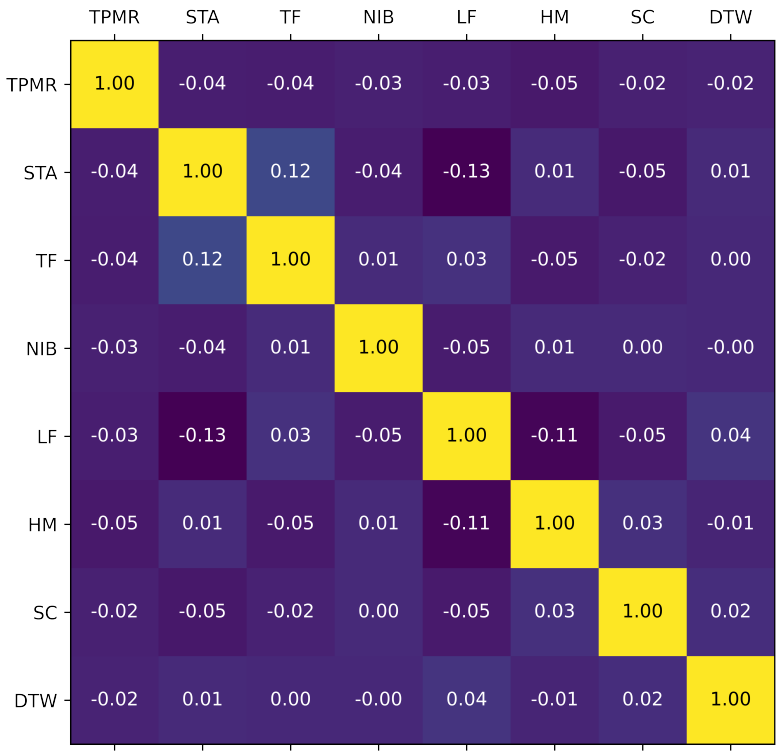


Figure A.1: Correlations between tropes using the Pearson coefficient on the Vaccine subset.

Model	**LF**			**HM**			**SC**			**DTW**		
	V+I	V	I	V+I	V	I	V+I	V	I	V+I	V	I
Bert-V+I	0.78	0.80	**0.29**	0.42	0.46	0.24	0.48	0.56	0.29	0.57	0.56	0.59
Bert-V	0.61	0.78	0.093	0.43	0.52	0.22	0.35	0.50	0.0	0.47	0.61	0.19
Bert-I	0.03	0.0	**0.29**	0.13	0.1	0.25	0.0	0.0	0.0	0.49	0.17	0.73
CovidBert-V+I	**0.80**	**0.83**	0.0	**0.59**	**0.64**	0.40	**0.64**	**0.70**	**0.50**	**0.68**	**0.58**	**0.82**
CovidBert-V	0.74	0.81	0.14	0.49	0.61	0.16	0.41	0.53	0.20	0.41	0.55	0.11
CovidBert-I	0.24	0.25	0.0	0.15	0.03	**0.48**	0.07	0.0	0.29	0.47	0.08	0.77
# support	68	64	4	61	51	10	16	12	4	36	20	16

Table A.2: F1-score per class on subsets of the test data. Models are trained on different subset of the train set. V stands for Vaccine and I for Immigration.

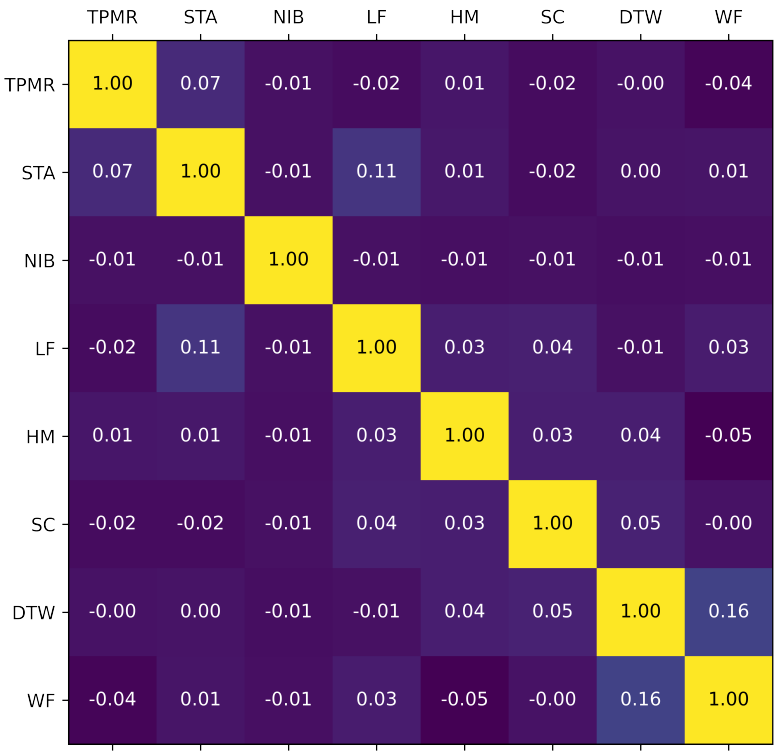


Figure A.2: Correlations between tropes using the Pearson coefficient on the Immigration subset.

Model	**WF**			**Weighted AVG**			**None**		
	V+I	V	I	V+I	V	I	V+I	V	I
Bert-V+I	0.55	0.0	0.55	0.58	0.62	0.42	0.83	0.78	0.88
Bert-V	0.0	0.0	0.0	0.50	0.63	0.15	0.79	0.79	0.80
Bert-I	0.41	0.0	**0.59**	0.12	0.04	0.42	0.78	0.71	0.90
CovidBert-V+I	**0.57**	0.0	0.57	**0.65**	**0.68**	**0.50**	**0.87**	**0.84**	**0.91**
CovidBert-V	0.0	0.0	0.0	0.54	0.66	0.11	0.83	0.81	0.85
CovidBert-I	0.20	0.0	0.58	0.16	0.08	0.44	0.80	0.72	**0.91**
# support	14	0	14	100%	63%	37%	411	218	193

Table A.3: (continued) F1-score per class on subsets of the test data. Models are trained on different subset of the train set. V stands for Vaccine and I for Immigration.

only data (I). We report results on subsets of the test set: full dataset (V+I), Vaccine only data (V) and Immigration only data (I). Lines Bert-V+I and CovidBert-V+I correspond to the Bert-FT and CovidBert-FT in Table 5.5.

We can see that models trained on full data tend to obtain best results, even outperforming models trained and tested on specific subsets. For example, a CovidBert model trained only on Vaccine data performs worse on Vaccine tweets than a model trained on both Vaccine and Immigration data. This shows that Tropes can be generalized and can be transferred from one topic to another, as the information of Immigration tweets help the classification of Vaccine tweets. However, we see that models tend to over-fit: models trained on Immigration data perform poorly on Vaccine data, and inversely.

Another takeaway is that Tropes on the Immigration subset are more difficult to detect. Indeed, the average F1-score for Vaccine data is consistently higher than the average on Immigration data. This can be due to the number of Vaccine tweets in the training set being twice the number of Immigration tweets.

A.3.4 Error Analysis

In this section, we analyze some false positive examples for every class and try to identify the cause. Results are obtained using our best detection model (CovidBert-FT). We focus on false positive rather than false negative because we think precision is a more important metric than recall in this use case. Obviously, a similar study could be done for false negatives.

Time Proves Me Right The model tends to classify time-related predictions ('it will soon be', '100 years ago was very similar to today') as Time Proves Me Right, even though it's not a sufficient condition, as it lacks the actual prediction of what "is to come".

$text_0$: `It's not too late but we must act quickly to reduce immigration by a lot, or it soon will be.`

$text_1$: `A global pandemic 100 years ago was very similar to today I thought the alleged Spanish Flu started as an experimental vaccine gone wrong in a US military hospital where all the vaccinated soldiers went down with bronchial pneumonia.`

Skepticism Towards Authorities In this case, even though authorities are mentioned in the text, it's not clearly implied that the user wants to promote skepticism or suspicion.

$text_0$: `THE FDA IS ATTEMPTING TO 'FLU SHOT' FUTURE COVID VACCINES. NO TRIAL`

FOR THE VACCINES OF THE FUTURE

$text_1$: Last year, Home Secretary @sajidjavid set out plans for a new skills-based immigration system that would mark the end of free movement. Find out more: #Brexit.

Too Fast Here, the model understood that something declared as "emergency use" is a fast and temporary solution, thus developed too quickly. Also, the term "experimental" could have misled the model, but calling a product arbitrarily experimental does not represent a trope.

$text_0$: Where is the long term safety data for monkeypox vaccines? And why mass produce a vaccine for such a rare illness if not created out of a laboratory?

$text_1$: They sure pushed the fear....some fell for the scam... EMERGENCY USE ONLY VACCINE DID NOT STOP TRANSMISSION OR INFECTION....AND TRIPLE JABBED ARE GETTING INFECTED REPEATEDLY WITH COVID ????????

$text_2$: The hypocrisy is stunning. How can you not say vaccine mandates of an experimental product with NO liability and poor safety and efficacy is not 100% about bodily autonomy and free choice.

Natural is Better The model seems to trigger positively on the word 'immunity', which is surely strongly correlated with the trope Natural is Better.

$text_0$: Nonsense, many of those did not need the jab and would have recovered, fact. What we now know is that the vaccine has killed more than it has saved, fact. It has also undermined the natural immune system because it NEVER was a vaccine! FACT! So naff off Mr Village idiot!

$text_1$: Maybe the survival rate of 99.98% has something to do with people not being obsessed with it. And the fact that most have immunity now, through infection and vaccines

Liberty, Freedom The model activation seems to correlate with the word 'forced', which may not always be a cause of Liberty, Freedom trope.

$text_0$: How is depopulation possible through Forced Vaccines. I read the article, and yes it was said, and it is an agenda discussed as well as followed by everyone participating in Davos.

$text_1$: so the Pentagon can just ignore federal laws, but individuals in the

armed forces can't ignore vaccine mandates?

$text_2$: And your Forced VACCINE prevents nothing! Only in your head! It has not stopped the SPREAD anywhere! And people like you never talk about the side effects nor natural immunity! BUT ABORTION is not Reproductive Rights...it is MURDER plain and simple!

Hidden Motives The word "expose" is quite often used to talk about something that is revealed through investigative reports: the model has learned this, and used it to wrongly label as Hidden Motive texts that had it in them (as shown for $text_3$). We also see a trend of mentioning organizations ('the Tories', 'Big Pharma', 'Bill gates') in false positive tweets, hinting that the model may have over-fit on the training data since these may be behind a Hidden Motive narratives, but not always.

$text_0$: Up to date? The old polio vaccine worked fine for 40 years until Bill Gates Corp created a new one for Africa which is a failed vaccine

$text_1$: Just a reminder that the Tories' betrayal over post-#Brexit #immigration is only part of the Establishment's treachery. 'All the same, all to blame'.

$text_2$: Here is why Big Pharma wants their vaccine in your kids. They know. Please retweet.

$text_3$: EXCLUSIVE: Citizen journalist 'who exposed Migrant Crisis' in bid to become MP @UKIP @Steve_Laws_ via @PoliticaliteUK

Scapegoat It is not sufficient to mention a famous person for it to be a Scapegoat, especially if those are just mentioned on the fly or to attributed conspiracies.

$text_0$: And if you were as smart as you think you are, you would know that birth rates are plunging throughout the vaccinated World. So, guess what? Babies are going to be rare and precious. Gates is achieving his dream of depopulation through vaccine.

$text_1$: Look at the spoilt immigrant rich brat advocating not just drag queen indoctrination for our children, but also acid attacks on our history (not Nelson Mandela's statue though).Looking at her, there's a song in Cabaret comes to mind.....

Defend the Weak These examples are incorrectly classified as "Defend the Weak". It seems like the model puts too much emphasis on the mention of "kids" and "children" when classifying this label.

$text_0$: `Look at the spoilt immigrant rich brat advocating not just drag queen indoctrination for our children, but also acid attacks on our history (not Nelson Mandela's statue though).Looking at her, there's a song in Cabaret comes to mind.....`

$text_1$: `NY Post: Children, who ordinarily love shots, recoil in pain and horror from vaccine mandate forced on them by parents.`

$text_2$: `The truth is, this 'demographic' would have been better off not injecting their kids with 3-5 'vaccines' every single month which then ended up directly causing autism. That's a fact.`

$text_3$: `CDC Caught Using False Data To Recommend Kids' COVID Vaccine CDC showcased highly misleading data about the risk of COVID-19 to kids when its expert vaccine advisers voted to recommend vaccines for children under five years old.`

Wicked Fairness This set of examples highlights tweets that mention comparison between two entities. However, they do not stress the unfair treatment they may have received, thus not qualifying as positive Wicked Fairness examples.

$text_0$: `At last count, two thirds of 'child refugees' entering the UK were adults lying about their age in order to cheat their way into #BenefitsBritain. A reliable dental check to confirm their ages was suggested but was deemed 'racist' by the Woke mob.`

$text_1$: `RIDDLE ME THIS!HOW DO IMMIGRANTS STRENGTHEN OUR COUNTRY BUT NOT THEIR OWN ?!?`

$text_2$: `France is a wealthy country perfectly capable of affording refuge to those on its territory who are in need. Migrants there should either be considered for asylum in France or be returned to their own countries as economic migrants.`

Publications list

The research carried out during this PhD thesis has lead to the publication of the following scientific papers:

Conference

1. Peskine, Y., Alfarano, G., Harrando, I., Papotti, P., Troncy, R. **Detecting COVID-19-related conspiracy theories in tweets**. In *MediaEval Benchmarking Initiative for Multimedia Evaluation Workshop*, 2021. [104]

2. Peskine, Y., Papotti, P., Troncy, R. **Detection of COVID-19-Related Conspiracy Theories in Tweets using Transformer-Based Models and Node Embedding Techniques**. In *MediaEval Benchmarking Initiative for Multimedia Evaluation Workshop*, 2022. [106]

3. Peskine, Y., Troncy, R., Papotti, P. **Analyzing COVID-Related Social Discourse on Twitter using Emotion, Sentiment, Political Bias, Stance, Veracity and Conspiracy Theories.** In *the 3^{rd} International Workshop on Knowledge Graphs for Online Discourse Analysis (BeyondFacts), Companion Proceedings of the ACM Web Conference*, 2023. [107]

4. Peskine, Y., Korenčic, D., Grubisic, I., Papotti, P., Troncy, R., Rosso, P. **Definitions Matter: Guiding GPT for Multi-label Classification**. In *In Findings of the Association for Computational Linguistics (EMNLP) (pp. 4054–4063). Association for Computational Linguistics*, 2023. [105]

5. Peskine, Y., Troncy, R., Papotti, P. **EURECOM at SemEval-2024 Task 4: Hierarchical Loss and Model Ensembling in Detecting Persuasion Techniques**. In *18^{th} International Workshop on Semantic Evaluation (SEMEVAL), co-located with NAACL*, 2024. [108]

6. Burel, G., Mensio, M., Peskine, Y., Troncy, R., Papotti, P., Alani, H. **CimpleKG: A Continuously Updated Knowledge Graph on Misinformation, Factors and Fact-Checks**. In *23^{rd} International Semantic Web Conference (ISWC)*, 2024. [22]

7. Flaccavento, A., Peskine, Y., Papotti, P., Torlone, R., Troncy, R. **Automated Detection of**

Tropes In Short Texts. In the *23rd International Conference on Computational Linguistics (COLING)*, 2025. [44]

8. "CimpleKG: a Continuously Updated Knowledge Graph of Fact-Checks and Related Misinformation" was presented during *Journées infox sur seine*, 2024.

Résumé en français

1.1 Introduction

La montée rapide de la désinformation, en particulier à travers les plateformes en ligne, est devenue un problème mondial ayant un impact sur la politique, la nature, la santé et la société. Des événements comme les élections américaines, le Brexit, la pandémie de COVID-19 et des crises environnementales comme les feux de brousse en Australie ont été marqués par la diffusion massive de contenus faux ou trompeurs. L'Organisation mondiale de la santé (OMS) a même créé le terme "infodémie" pour décrire le flux accablant de désinformation lié au Coronavirus. En 2019, la désinformation était considérée comme le cinquième plus gros problème aux États-Unis, devançant des sujets comme la criminalité, le changement climatique ou le racisme.

Cette explosion des informations en ligne, en particulier via des plateformes comme X (anciennement Twitter), qui génère plus de 300 millions de publications par jour, a rendu la tâche d'identifier et de lutter contre la désinformation plus complexe. Le contenu partagé en ligne peut prendre différentes formes, telles que du texte, des images, des vidéos ou des métadonnées, ce qui rend la vérification des faits une tâche ardue.

Les organisations de vérification des faits et les plateformes jouent un rôle essentiel dans la lutte contre la désinformation. Ces groupes travaillent sans relâche pour démystifier les fausses informations virales en se référant à des sources fiables. Le Réseau international de vérification des faits (IFCN) est un acteur clé dans ce domaine, soutenant des organisations comme Snopes, Politifact et d'autres. Cependant, la vérification des faits est un processus intrinsèquement long. Alors que la désinformation se propage rapidement, les vérificateurs de faits se concentrent souvent uniquement sur les affirmations les plus virales. Selon une enquête de 2020, plus de 44 % des vérificateurs de faits ont exprimé le besoin d'outils pour identifier les affirmations déjà vérifiées, car les fausses informations virales ont tendance à réapparaître avec le temps. Des plateformes sociales comme Meta (Facebook, Instagram, Threads), Alphabet (Google et YouTube) et X ont commencé à collaborer avec des organisations de vérification des faits pour limiter la propagation de la désinformation. Par exemple, Meta a

travaillé auparavant avec l'IFCN pour étiqueter les contenus trompeurs, mais ce programme a été abandonné aux États-Unis et remplacé par une nouvelle initiative appelée Community Notes.

L'intelligence artificielle (IA) et le traitement du langage naturel (TALN) jouent un rôle crucial dans la lutte contre les défis liés à la désinformation. Le TALN est un domaine qui utilise des méthodes informatiques pour comprendre, générer et traiter le langage humain. Les techniques de TALN sont utilisées pour détecter et analyser la désinformation en étudiant la propagation des fausses nouvelles et en détectant les contenus nuisibles. Récemment, des modèles de langage de grande taille (LLM), tels que GPT-3 et GPT-4, ont fait des progrès significatifs, offrant des moyens plus efficaces et plus précis d'analyser des textes à grande échelle. Ces modèles sont entraînés sur de vastes ensembles de données et ont démontré leur capacité à surpasser les annotateurs humains dans certaines tâches, telles que l'annotation de textes et l'analyse de sentiments.

Le TALN s'est révélé particulièrement utile pour combattre la désinformation, car il permet d'analyser d'énormes ensembles de données, y compris des publications sur les réseaux sociaux, des articles de presse et d'autres contenus en ligne. Les chercheurs ont utilisé le TALN pour suivre la manière dont la désinformation se propage au sein des réseaux, identifiant des modèles qui contribuent à l'amplification des fausses nouvelles. Cependant, malgré ces progrès, plusieurs défis persistent. Par exemple, bien que les LLM aient montré de bonnes performances dans certaines tâches, leur nature "boîte noire" rend difficile l'explication de la manière dont ils parviennent à leurs conclusions. Ce manque de transparence constitue un défi majeur pour assurer la fiabilité et la responsabilité de ces modèles.

La thèse aborde les question de recherche suivantes:

- ***RQ1:*** Comment modéliser les relations entre les différents types de données utilisées pour la vérification des faits ?
- ***RQ2:*** Comment mieux comprendre les documents textuels grâce à des approches automatiques ?
- ***RQ3:*** Comment étendre les notions de similarité textuelle pour les applications de vérification des faits ?

La thèse aborde ces défis en proposant plusieurs contributions visant à améliorer la compréhension et la vérification de la désinformation. Tout d'abord, elle introduit Cimple KG, un graphe de connaissances mis à jour en continu qui organise les données liées à la désinformation, en rassemblant diverses sources de contenus mensongers. Ce graphe de connaissances inclut des données provenant de publications sur les réseaux sociaux, d'articles de presse et de

sites web de vérification des faits, et intègre diverses caractéristiques de ces documents. Cela répond à la première question de recherche en structurant et en modélisant les relations entre les différentes formes de données liées à la désinformation. Le travail propose également des méthodes d'analyse des caractéristiques textuelles telles que les émotions, les sentiments, les tendances politiques, les théories du complot, les techniques de persuasion et les tropes, ce qui contribue à mieux comprendre la désinformation et à répondre à la deuxième question de recherche.

De plus, la thèse étend la notion de similarité textuelle pour les applications de vérification des faits. Les techniques traditionnelles de similarité sémantique ne sont pas toujours suffisantes pour cette tâche, c'est pourquoi l'auteur propose de nouvelles mesures de similarité adaptées à la vérification des faits. Ces mesures se concentrent non seulement sur les entités et les concepts textuels, mais aussi sur la capacité d'un document à servir de source de preuve pour la vérification des faits. La thèse compare l'efficacité de ces nouvelles mesures de similarité à différents niveaux de granularité, y compris les courtes affirmations et les longs articles de presse. Cette approche vise à améliorer la précision des systèmes automatiques de vérification des faits, les rendant mieux adaptés aux applications du monde réel.

Grâce à ces contributions, la thèse améliore considérablement les outils disponibles pour comprendre et lutter contre la désinformation. Elle s'appuie sur les recherches existantes en proposant des méthodes innovantes d'analyse de texte et de vérification des faits, garantissant que les systèmes automatiques sont non seulement plus précis, mais aussi plus explicables. En développant Cimple KG et en étendant les mesures de similarité, la thèse fournit des ressources pratiques pour les chercheurs et les organisations travaillant à identifier, suivre et combattre la désinformation dans un paysage numérique de plus en plus complexe.

1.2 Travail Connexe

Dans cette section, nous abordons les concepts fondamentaux utiles pour comprendre le reste de la thèse. Nous présentons les récentes avancées de la recherche en Traitement Automatique du Langage Naturel (TALN), Graphes de Connaissances (KG) et Détection de Désinformation.

1.2.1 Traitement Automatique du Langage Naturel (TALN)

Le TALN est une discipline qui cherche à comprendre le langage textuel en utilisant des ressources informatiques. Ces dernières années, avec l'émergence de puissants modèles de langage comme ChatGPT, l'intérêt pour ce domaine a considérablement augmenté.

Les modèles de langage (LM) sont des modèles probabilistes qui traitent diverses tâches textuelles, telles que la traduction ou la classification des émotions. Initialement basés sur

les n-grammes, les modèles ont évolué vers des réseaux de neurones récurrents (RNN) pour mieux saisir le contexte séquentiel des textes. Le mécanisme d'attention, introduit par le modèle Transformer, permet de modéliser les relations entre les mots sans tenir compte de leur distance. Le modèle Transformer a obtenu des résultats de pointe dans plusieurs tâches et permet un apprentissage plus rapide. BERT est un modèle basé sur l'architecture Transformer qui apprend les représentations de manière bidirectionnelle, ce qui améliore la robustesse et les performances. De nombreuses architectures ont été développées sur cette base, comme RoBERTa et AlBERT, avec des objectifs tels que l'amélioration des performances ou la réduction des besoins en mémoire. Les modèles GPT, développés par OpenAI, ont marqué une avancée dans la génération de texte. GPT-3 et ses successeurs, comme GPT-4, sont capables de traiter une large gamme de tâches sans supervision explicite. Ces modèles ont permis de démontrer des capacités d'apprentissage en "zéro-shot" et "few-shot" (avec peu d'exemples).

1.2.2 Graphes de Connaissances (KG)

Les graphes de connaissances sont des structures de données qui représentent des relations entre des points de données sous forme de graphes. Ils sont utilisés dans des domaines variés comme le web sémantique, les moteurs de recherche, les systèmes de recommandation et les réseaux sociaux. Le RDF est un cadre utilisé pour représenter les graphes de connaissances sous forme de triples (sujet, prédicat, objet). Schema.org fournit un modèle standard pour représenter ces informations, facilitant l'interconnexion des données entre différentes entités. ClaimReview est un format de données structuré utilisé par les organisations de vérification des faits pour publier des informations sur les déclarations examinées. Ce format est intégré à des outils comme l'API de vérification des faits de Google et permet de lier les articles de vérification des faits aux revendications qu'ils vérifient.

Les graphes de connaissances ont été utilisés pour stocker des données de vérification des faits, en reliant les affirmations à leur contexte, comme le temps, les articles de vérification ou les entités nommées.

ClaimsKG est l'une des premières bases de données de graphes de connaissances liée à la vérification des faits, mais sa couverture reste limitée. The Database of Known Fakes (DBKF) est une initiative plus récente, qui permet aux utilisateurs de naviguer dans des documents fact-checkés et offre une interface web pour effectuer des recherches.

1.3 Détecter les caractéristiques de désinformation

1.3.1 Détection de théorie du complot dans les tweets

Pour la première tâche, une approche de classification multi-étiquettes et multi-classes a été appliquée pour classer les théories du complot dans le texte des tweets. Nous avons utilisé des modèles basés sur des Transformers, en particulier CT-BERT, pré-entrainé sur de grands corpus de données Twitter, comme modèle principal pour la classification textuelle. La deuxième tâche s'est concentrée sur la classification des nœuds (utilisateurs Twitter) dans un graphique d'interactions des utilisateurs, en utilisant des techniques d'intégration de nœuds (telles que node2vec) et des réseaux de neurones de graphes (GNN). Pour la troisième tâche, une combinaison de caractéristiques textuelles et de graphe a été utilisée pour la classification de la désinformation, avec à la fois le contenu textuel des tweets et les caractéristiques du graphe comme entrée dans le modèle de classification.

Nous avons abordé les tâches de classification de texte avec un pipeline de prétraitement détaillé, incluant le remplacement des emojis et la suppression des hashtags. Les modèles ont été affinés avec des fonctions de perte personnalisées et une approche de validation croisée à cinq plis. Pour la classification des nœuds, node2vec a été utilisé pour générer des intégrations de nœuds, suivies de classificateurs d'apprentissage automatique tels que Random Forest et Multi-Layer Perceptron. Dans la troisième tâche, les caractéristiques textuelles et de graphe ont été concaténées et introduites dans un modèle de classification.

Les résultats ont indiqué que le modèle CT-BERT a bien performé pour la classification textuelle, et que les stratégies d'assemblage (comme le vote majoritaire) ont amélioré les résultats pour les tâches textuelles. Cependant, les tâches basées sur les graphes ont été plus difficiles, la classification des nœuds ayant donné des résultats moins bons. Le modèle de fusion combinant les caractéristiques textuelles et de graphe n'a pas surpassé le modèle uniquement textuel, suggérant qu'une amélioration supplémentaire est nécessaire. La meilleure performance du concours a été obtenue grâce à un modèle combinant l'assemblage des modèles CT-BERT pour la classification textuelle, obtenant un score MCC de 0.719 sur les données de test.

1.3.2 Détection de techniques de persuasion dans les memes

Dans cette section, nous présentons le travail réalisé lors de notre participation à la tâche SemEval-Task4 : Détection multilingue des techniques de persuasion dans les memes. La tâche comprend trois sous-tâches : la sous-tâche 1, qui se concentre sur la détection des techniques de persuasion dans le contenu textuel des memes ; la sous-tâche 2a, qui utilise à la fois l'image et le texte pour la détection ; et la sous-tâche 2b, qui traite de la détection binaire.

Nous nous concentrons principalement sur la sous-tâche 1, où l'objectif est d'identifier 20 techniques de persuasion différentes dans le texte.

Notre approche repose sur un ensemble des trois meilleurs modèles pour chaque technique de persuasion, où nous avons constaté que l'exploitation de la structure hiérarchique des données et l'utilisation d'une fonction de perte hiérarchique ont donné les meilleurs résultats.

Nous avons expérimenté plusieurs modèles basés sur des transformateurs, dont BERT, RoBERTa, AlBERT, DistilBERT et DeBERTa, pour détecter les techniques de persuasion. De plus, nous avons intégré plusieurs ensembles de données d'entraînement, tels que SemEval-2021 et le corpus PTC, ce qui nous a permis de construire un modèle plus robuste. Nous avons également exploré l'utilisation d'une fonction de perte hiérarchique pour tenir compte des relations entre les techniques de persuasion, et nous avons testé différentes fonctions de perte telles que la Binary Cross Entropy (BCE), la Cross Entropy (CE), la Focal Loss (FL) et une fonction de perte hiérarchique personnalisée (HL).

L'augmentation de données a été utilisée pour résoudre le problème des données d'entraînement limitées pour certaines classes, en utilisant les traductions aller-retour et la génération GPT-4-Turbo, bien que cette dernière n'ait apporté qu'un léger gain. Notre processus d'entraînement a suivi une approche systématique avec l'optimiseur AdamW, la planification du taux d'apprentissage et l'arrêt précoce basé sur la performance F1H sur l'ensemble de validation.

Dans notre soumission finale, nous avons combiné les meilleurs modèles en fonction de la performance F1-score pour chaque technique de persuasion. Les résultats des ensembles de validation et de test dans différentes langues montrent que notre méthode d'assemblage a conduit à de bonnes performances, surtout en anglais, bien qu'il y ait eu des difficultés avec les langues non anglaises, comme l'arabe, en raison de problèmes de traduction.

À travers nos expériences, nous avons observé que les modèles entraînés avec une perte hiérarchique et ceux détectant les classes parentes ont mieux performé, confirmant l'importance de la nature hiérarchique de la tâche. Les techniques comme « Appeal to Authority » et « Ethos » étaient plus faciles à détecter, tandis que d'autres comme « Obfuscation » et « Causal Oversimplification » étaient plus difficiles. Globalement, notre approche a montré son efficacité pour détecter les techniques de persuasion dans les memes, bien que des défis subsistent pour améliorer la performance pour toutes les techniques et langues.

1.4 Les définitions comptent

Dans cette section, nous analysons l'impact des définitions sur la performance des modèles de langage, en particulier GPT-3, dans le cadre de la détection des théories du complot dans des tweets. Nous montrons que l'utilisation de définitions de classes peut améliorer

considérablement les résultats de classification par rapport à une approche zéro-shot de base, suggérant que GPT-3 peut effectivement exploiter les connaissances issues de ces définitions pour effectuer des tâches de classification complexes. Pour évaluer l'impact des définitions, nous avons exploré deux types de définitions : celles rédigées par des humains, fournies dans les lignes directrices du jeu de données, et celles générées par exemple à partir d'un ensemble d'exemples de tweets. Bien que les définitions humaines offrent les meilleures performances, les définitions générées par exemple ont également montré des améliorations significatives par rapport à la ligne de base zéro-shot.

Nous avons ensuite approfondi l'exploration de la manière dont GPT-3 "comprend" et "applique" ces définitions pour effectuer des classifications. Nous avons utilisé des mesures de similarité sémantique pour évaluer la cohérence entre les définitions humaines et générées par exemple, et pour corréler ces similarités avec les performances des modèles de classification. Les résultats ont révélé que plus la similarité entre les définitions générées par exemple et celles rédigées par des humains était élevée, plus les performances du modèle étaient bonnes. En outre, nous avons constaté que lorsque deux définitions générées par exemple étaient similaires, les prédictions associées avaient également tendance à être cohérentes, ce qui démontre que GPT-3 est capable d'appliquer correctement les définitions à des tâches de classification.

Une analyse plus poussée a révélé que la longueur des définitions générées par exemple n'était pas corrélée avec la performance de classification, ce qui suggère que c'est la qualité des informations contenues dans la définition, et non la quantité, qui impacte réellement les résultats. Ces tests de compréhension des définitions ont renforcé l'idée que GPT-3 peut interpréter et appliquer des définitions de manière efficace, ce qui ouvre des possibilités intéressantes pour l'utilisation de définitions générées par exemple dans des situations où les définitions humaines ne sont pas disponibles. Néanmoins, bien que l'approche utilisant GPT-3 et les définitions générées par exemple offrent des résultats prometteurs, elle reste inférieure aux méthodes de fine-tuning comme CT-BERT, qui surpassent largement les performances de classification.

1.5 Détection automatique de caractéristiques textuelles dans les publications sur les réseaux sociaux

1.5.1 Émotion, Sentiment et Biais politique

Cette section se concentre sur la détection des caractéristiques textuelles dans les publications sur les réseaux sociaux, en particulier à l'aide de modèles basés sur les transformateurs. Les facteurs incluent des éléments tels que les émotions, les sentiments, les biais politiques, les techniques de persuasion, les théories du complot et les tropes, qui influencent la manière

dont nous comprenons le contenu en ligne. Nous décrivons notre méthode pour créer des modèles permettant de détecter ces facteurs et explorons leurs interactions pour en tirer des insights utiles.

En se concentrant sur le COVID-19, nous analysons les discours en ligne au-delà de la désinformation. L'étude des émotions, du sentiment et du biais politique dans les tweets sur le COVID-19 révèle que ces facteurs sont fortement influencés par des événements comme les élections américaines de 2016 et la pandémie de COVID-19, où la désinformation a pris une ampleur considérable. Nous avons utilisé des modèles de classification, tels que BERT, pour détecter ces facteurs dans les tweets, en nous appuyant sur des ensembles de données spécifiques. Les résultats montrent que des éléments comme le sentiment et les émotions sont souvent associés à des opinions politiques et à des théories du complot.

Nos modèles ont permis d'analyser les corrélations entre ces facteurs, en montrant que des sujets comme les masques faciaux et les fermetures d'écoles sont des sujets très controversés, générant des émotions négatives et un biais politique marqué. En conclusion, cette étude souligne l'importance des émotions, du sentiment et des biais dans la compréhension du discours en ligne, en particulier dans un contexte de désinformation et de théories du complot liées au COVID-19.

1.5.2 Tropes

Dans cette section, nous abordons les tropes, des dispositifs narratifs récurrents utilisés pour transmettre des idées ou des thèmes spécifiques. Les tropes sont largement employés dans les médias et en ligne pour influencer les émotions et les perceptions du public. Cependant, elles sont aussi utilisées pour manipuler et tromper, notamment dans les discours anti-vaccins et sur l'immigration. Cette étude propose une méthode pour détecter automatiquement ces tropes dans de courts textes provenant des médias sociaux, en utilisant des techniques d'apprentissage supervisé.

Nous définissons les tropes suivantes: Scepticisme envers l'autorité (STA), Défendre les faibles (DTW), Motifs cachés (HM), Liberté, autonomie (LF), Le naturel est meilleur (NIB), Le temps me donnera raison (TPMR), Trop rapide (TF), Bouc émissaire (SC), Injustice perçue (WF). Nous notons ces tropes dans 3300 tweets, et utilisons des modèles de langages pour les détecter de manière automatique.

Nous utilisons les quatre modèles suivants pour détecter les tropes: BERT-FT, CT-FT, Chat-GPT, LLama-3. Nous utilisons les LLMs de manière Zero-Shot et entraînons les modèles basés sur BERT sur 80% des données annotées. Les meilleurs résultats sont obtenus avec le modèle CT.

Nous comparons aussi les tropes avec les théories du complot et les techniques de persuasion.

Nous utilisons notre meilleur modèle pour détecter les tropes sur un les jeux de données de théories du complot et de techniques de persuasion. Inversement, nous utilisons les modèles de détection de théorie du complot et de technique de persuasion sur notre jeu de données de tropes. Nous n'avons analysé aucune corrélation majeure, ce qui prouve que les tropes sont bien orthogonales aux autres caractéristiques.

1.6 Cimple KG

CimpleKG est un graphe de connaissances (KG) public, mis à jour en continu, conçu pour soutenir la recherche sur la désinformation en reliant divers ensembles de données de désinformation statiques précédemment publiées avec des allégations vérifiées quotidiennement par des organisations de vérification des faits. Il intègre des informations supplémentaires telles que les entités nommées, les facteurs contextuels (émotions, sentiment, tendances politiques, théories du complot et techniques de propagande) et normalise les systèmes de notation utilisés par les vérificateurs de faits. Contrairement à d'autres tentatives de création de graphes de connaissances similaires, CimpleKG est considérablement plus grand en termes de couverture temporelle, de pays, de langues, de quantité et de fraîcheur. Le KG comprend plus de 203 000 entrées ClaimReview en 26 langues, émises par 77 vérificateurs de faits de plus de 36 pays, et se met à jour quotidiennement avec plus de 15 millions de triples RDF. Les données sont disponibles publiquement via un point d'accès SPARQL, des fichiers de dump RDF et une API RESTful.

CimpleKG s'appuie sur l'ontologie Schema.org, en particulier le type de données ClaimReview, et l'étend avec de nouvelles fonctionnalités telles que des facteurs textuels (par exemple, émotions et sentiments) et des entités nommées. Ces fonctionnalités supplémentaires aident les chercheurs à explorer les relations entre les allégations, leurs vérifications et les facteurs qui peuvent influencer la perception publique, comme les biais politiques ou les récits complotistes. Les entités nommées, telles que les individus ou les lieux, sont identifiées à l'aide de DBpedia Spotlight, tandis que les fonctionnalités textuelles sont extraites automatiquement des allégations à l'aide de modèles entraînés. Des ensembles de données statiques liées à la désinformation, tels que des publications sur les réseaux sociaux ou des ensembles de données de recherche, sont également intégrés dans CimpleKG, offrant un contexte plus riche pour analyser la désinformation.

Le processus de collecte et d'intégration des données de vérification des faits comprend plusieurs étapes, notamment la collecte des URL ClaimReview à partir des agrégateurs de vérification des faits, l'extraction des données d'allégation à partir des sites Web des vérificateurs de faits, le nettoyage et la validation des données, la cartographie des évaluations des vérificateurs de faits sur une échelle unifiée et le traitement des données pour les intégrer dans le KG. Le processus est automatisé pour mettre à jour le KG avec de nouvelles vérifications

des faits et des allégations quotidiennement, chaque mise à jour incluant plus de 8 millions de triples RDF et de nouvelles informations contextuelles, telles que des publications sur les réseaux sociaux et des articles de presse associés. L'intégration des ensembles de données statiques, avec les mises à jour quotidiennes des vérifications des faits, renforce la capacité du KG à suivre la désinformation et sa propagation au fil du temps.

CimpleKG a été utilisé dans diverses applications de recherche, notamment la correction automatisée de la désinformation via le Misinfome Bot, la surveillance de la propagation de la désinformation via le Fact-Checking Observatory, et l'évaluation de la vérification des faits par la foule par rapport à celle des experts. Le graphe de connaissances alimente également plusieurs systèmes de recherche, tels que l'Iffy Index pour l'évaluation de la crédibilité des sources et Linked Credibility Reviews pour la détection et l'explication de la désinformation. Avec son modèle de données complet et son intégration robuste des sources de désinformation dynamiques et statiques, CimpleKG fournit des informations précieuses aux chercheurs et aux praticiens qui luttent contre la désinformation en temps réel.

1.7 Nouvelles concepts de similarité textuelles

Dans cette section, nous présentons de nouvelles approches pour mesurer la similarité textuelle dans le contexte de la recherche sur la désinformation. Nous proposons plusieurs concepts de similarité, notamment pour les récits et les entités, et comparons des méthodes automatiques de récupération de documents similaires. Enfin, nous introduisons une approche de similarité pour des documents de granularité différente.

Nous définissons une notion de similarité basée sur la vérification des faits, en introduisant le concept de "fact-checking match", où un document peut être utilisé pour vérifier un autre, ainsi qu'une notion plus souple de "narrative match", où deux documents partagent un récit similaire. Nous avons annoté un jeu de données de 200 paires de tweets et de revendications pour ces deux types de similarité.

Nous explorons la similarité basée sur les entités et concepts. Nous montrons que les entités jouent un rôle crucial dans la manière dont nous percevons la similarité entre documents, et proposons une méthode de comparaison de documents qui prennent en compte des entités ou concepts similaires, même lorsqu'ils apparaissent dans des contextes différents.

Ensuite, nous comparons deux approches automatiques pour la récupération de documents similaires : Sentence-BERT et une méthode basée sur des graphes. Nous trouvons que Sentence-BERT surpasse largement la méthode par graphes en termes de récupération de documents pertinents, bien que cette dernière présente des avantages pour l'extraction de correspondances d'entités spécifiques.

Enfin, nous abordons la question de la comparaison de documents de granularité différente, comme des articles longs et des tweets courts. Nous proposons une approche qui utilise un coefficient local-global pour ajuster l'influence de l'information locale et globale dans l'embedding des documents, et nous montrons que cette méthode permet de récupérer de manière efficace des revendications pertinentes pour des passages d'articles longs.

1.8 Conclusion et Perspectives

Cette thèse présente des outils et des ressources pour aider les vérificateurs de faits dans la lutte contre la désinformation. En utilisant des modèles linguistiques et des graphes de connaissances, nous avons développé des méthodes pour mieux comprendre, détecter et expliquer la propagation de la désinformation en ligne. Nos contributions incluent la détection automatique de caractéristiques textuelles, la modélisation de données via des graphes de connaissances et la mesure de similarité textuelle.

1.8.1 Conclusion

Nous avons répondu aux questions de recherche posées au début de la thèse. Pour la première question, "Comment modéliser les relations entre les différents types de données utilisées pour la vérification des faits ?", nous avons introduit Cimple KG, un graphe de connaissances qui relie différents documents, tels que des articles de presse, des publications sur les réseaux sociaux, des revendications, et bien plus encore. Ce graphe contient plus de 15 millions de triples RDF et est une ressource précieuse utilisée dans de nombreuses études et outils.

En réponse à la deuxième question, "Comment mieux comprendre les documents textuels grâce à des approches automatiques ?", nous avons formé plusieurs modèles CT-BERT pour détecter des théories du complot dans des tweets et avons utilisé des modèles BERT pour détecter des techniques de persuasion dans des mèmes. Nous avons également exploré l'utilisation de grands modèles linguistiques (LLMs), comme GPT-3.5 et LLaMA-2, pour annoter des théories du complot et des techniques de propagande, et avons montré que la définition précise des classes dans les prompts affecte la qualité des résultats de classification.

Enfin, pour la troisième question, "Comment étendre la notion de similarité textuelle pour les applications de vérification des faits ?", nous avons proposé de nouvelles mesures de similarité, prenant en compte des entités, des scénarios de vérification des faits et des documents de différentes longueurs. Nos expériences montrent que Sentence-BERT est l'outil le plus efficace pour récupérer des revendications précédemment vérifiées dans divers contextes.

1.8.2 Perspectives

Nos travaux peuvent être enrichis par de nouvelles caractéristiques textuelles, comme la détection du discours de haine, qui est étroitement liée à la désinformation, en particulier pendant la pandémie de COVID-19. Nous pourrions également améliorer la robustesse des modèles en les entraînant sur des données plus diversifiées, car leurs performances sont actuellement meilleures sur des tweets que sur des articles de presse.

Le travail sur Cimple KG est modulaire, ce qui permet d'ajouter facilement de nouvelles ressources et détections de facteurs, comme les tropes, ou des ensembles de données supplémentaires liés à la désinformation sur des sujets comme le changement climatique. De plus, bien que nous ayons automatisé une partie de la correction des erreurs dans les ClaimReviews de l'API de vérification des faits de Google, il reste encore des améliorations à apporter dans cette phase de nettoyage.

En ce qui concerne les modèles de génération de définitions à partir des LLMs, plusieurs limitations sont à explorer. Par exemple, bien que la classification des théories du complot en zéro-shot soit possible avec des définitions générées par LLM, ces définitions doivent être basées sur des exemples annotés, ce qui nécessite encore des ajustements. De futurs travaux pourraient se concentrer sur l'optimisation de ces méthodes et sur l'amélioration de la classification des techniques de propagande.

Enfin, notre approche de la similarité documentaire présente des limites, notamment avec l'approche par graphes, qui est moins performante que Sentence-BERT pour récupérer les bons documents. Nous pourrions explorer des approches hybrides combinant les deux méthodes pour tirer parti de leurs forces respectives. Le travail sur la comparaison de documents de longueurs différentes pourrait également être approfondi en élargissant le jeu de données pour rendre les résultats plus robustes.

Bibliography

[1] Francisca Adoma Acheampong, Henry Nunoo-Mensah, and Wenyu Chen. Transformer models for text-based emotion detection: a review of bert-based approaches. *Artificial Intelligence Review*, 54(8):5789–5829, Dec 2021.

[2] Acheampong Francisca Adoma, Nunoo-Mensah Henry, and Wenyu Chen. Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition. In *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pages 117–121, 2020.

[3] Mubashara Akhtar, Michael Schlichtkrull, Zhijiang Guo, Oana Cocarascu, Elena Simperl, and Andreas Vlachos. Multimodal automated fact-checking: A survey. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5430–5448, Singapore, December 2023. Association for Computational Linguistics.

[4] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2):211–236, 2017.

[5] Jennifer Allen, Cameron Martel, and David G Rand. Birds of a feather don't fact-check each other: Partisanship and the evaluation of news in twitter's birdwatch crowdsourced fact-checking program. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA, 2022. Association for Computing Machinery.

[6] Rami Aly, Zhijiang Guo, Michael Sejr Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. The fact extraction and VERification over unstructured and structured information (FEVEROUS) shared task. In Rami Aly, Christos Christodoulopoulos, Oana Cocarascu, Zhijiang Guo, Arpit Mittal, Michael Schlichtkrull, James Thorne, and Andreas Vlachos, editors, *Proceedings of the Fourth Workshop on Fact Extraction and VERification (FEVER)*, pages 1–13, Dominican Republic, November 2021. Association for Computational Linguistics.

[7] Anthropic. Introducing Claude. https://www.anthropic.com/news/introducing-claude, 2023.

[8] Isabelle Augenstein. Towards explainable fact checking. *arXiv preprint arXiv:2108.10274*, 2021.

[9] Ajay Bandi and Aziz Fellah. Socio-Analyzer: A Sentiment Analysis Using Social Media Data. In Frederick Harris, Sergiu Dascalu, Sharad Sharma, and Rui Wu, editors, *Proceedings of 28th International Conference on Software Engineering and Data Engineering*, volume 64 of *EPiC Series in Computing*, pages 61–67. EasyChair, 2019.

[10] Adrien Barbaresi. Trafilatura: A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction. In *59th Annual Meeting of the Association for Computational Linguistics (ACL): System Demonstrations*, pages 122–131. Association for Computational Linguistics, 2021.

[11] Corey H. Basch, Zoe Meleo-Erwin, Joseph Fera, Christie Jaime, and Charles E. Basch. A global pandemic in the time of viral memes: Covid-19 vaccine misinformation and disinformation on tiktok. *Human Vaccines & Immunotherapeutics*, 17(8):2373–2377, 2021. PMID: 33764283.

[12] Janek Bevendorff, Xavier Bonet Casals, Berta Chulvi, Daryna Dementieva, Ashaf Elnagar, Dayne Freitag, Maik Fröbe, Damir Korenčić, Maximilian Mayerl, Animesh Mukherjee, Alexander Panchenko, Martin Potthast, Francisco Rangel, Paolo Rosso, Alisa Smirnova, Efstathios Stamatatos, Benno Stein, Mariona Taulé, Dmitry Ustalov, Matti Wiegmann, and Eva Zangerle. Overview of pan 2024: Multi-author writing style analysis, multilingual text detoxification, oppositional thinking analysis, and generative ai authorship verification. In Nazli Goharian, Nicola Tonellotto, Yulan He, Aldo Lipani, Graham McDonald, Craig Macdonald, and Iadh Ounis, editors, *Advances in Information Retrieval*, pages 3–10, Cham, 2024. Springer Nature Switzerland.

[13] Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. Gpt-neox-20b: An open-source autoregressive language model, 2022.

[14] Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, March 2021.

[15] Enguerrand Boitel, Alaa Mohasseb, and Ella Haig. A comparative analysis of gpt-3 and bert models for text-based emotion recognition: Performance, efficiency, and robustness. In Nitin Naik, Paul Jenkins, Paul Grace, Longzhi Yang, and Shaligram Prajapat, editors, *Advances in Computational Intelligence Systems*, pages 567–579, Cham, 2024. Springer Nature Switzerland.

[16] Mostafa Bouziane, Hugo Perrin, Amine Sadeq, Thanh Nguyen, Aurélien Cluzeau, and Julien Mardas. FaBULOUS: Fact-checking based on understanding of language over unstructured and structured information. In Rami Aly, Christos Christodoulopoulos, Oana Cocarascu, Zhijiang Guo, Arpit Mittal, Michael Schlichtkrull, James Thorne, and Andreas Vlachos, editors, *Proceedings of the Fourth Workshop on Fact Extraction and VERification (FEVER)*, pages 31–39, Dominican Republic, November 2021. Association for Computational Linguistics.

[17] Alexandre Bovet and Hernán A. Makse. Influence of fake news in twitter during the 2016 us presidential election. *Nature Communications*, 10(1):7, Jan 2019.

[18] J S Brennen, F M Simon, P N Howard, and R K Nielsen. Types, sources, and claims of COVID-19 misinformation. *Reuters Institute for the Study of Journalism*, 2020.

[19] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020.

[20] Grégoire Burel, Tracie Farrell, and Harith Alani. Demographics and topics impact on the co-spread of covid-19 misinformation and fact-checks on twitter. *Information Processing & Management*, 58(6), 2021.

[21] Grégoire Burel, Tracie Farrell, Martino Mensio, Prashant Khare, and Harith Alani. Co-spread of misinformation and fact-checking content during the covid-19 pandemic. In *12th International Conference on Social Informatics (SocInfo)*, pages 28–42. Springer, 2020.

[22] Grégoire Burel, Martino Mensio, Youri Peskine, Raphael Troncy, Paolo Papotti, and Harith Alani. Cimplekg: A continuously updated knowledge graph on misinformation, factors and fact-checks. In *The Semantic Web – ISWC 2024: 23rd International Semantic Web Conference, Baltimore, MD, USA, November 11–15, 2024, Proceedings, Part III*, page 97–114, Berlin, Heidelberg, 2024. Springer-Verlag.

[23] Grégoire Burel, Mohammadali Tavakoli, and Harith Alani. Exploring the impact of automated correction of misinformation in social media. *AI Magazine*, 45(2):227–245, 2024.

[24] Chen-Hsi Chang, Hung-Ting Su, Jui-Heng Hsu, Yu-Siang Wang, Yu-Cheng Chang, Zhe Yu Liu, Ya-Liang Chang, Wen-Feng Cheng, Ke-Jyun Wang, and Winston H. Hsu. Situation

and behavior understanding by trope detection on films. In *Proceedings of the Web Conference 2021*, WWW '21, page 3188–3198, New York, NY, USA, 2021. Association for Computing Machinery.

[25] Giovanni Da San Martino, Alberto Barrón-Cedeño, and Preslav Nakov. Findings of the NLP4IF-2019 Shared Task on Fine-Grained Propaganda Detection. In Anna Feldman, Giovanni Da San Martino, Alberto Barrón-Cedeño, Chris Brew, Chris Leberknight, and Preslav Nakov, editors, *Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 162–170, Hong Kong, China, 2019. Association for Computational Linguistics.

[26] Giovanni Da San Martino, Alberto Barrón-Cedeño, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. SemEval-2020 Task 11: Detection of Propaganda Techniques in News Articles. In Aurelie Herbelot, Xiaodan Zhu, Alexis Palmer, Nathan Schneider, Jonathan May, and Ekaterina Shutova, editors, *14th International Workshop on Semantic Evaluation (SemEval)*, pages 1377–1414. International Committee for Computational Linguistics, December 2020.

[27] Giovanni Da San Martino, Stefano Cresci, Alberto Barrón-Cedeño, Seunghak Yu, Roberto Di Pietro, and Preslav Nakov. A survey on computational propaganda detection. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20, 2021.

[28] Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeño, Rostislav Petrov, and Preslav Nakov. Fine-grained analysis of propaganda in news article. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5636–5646, Hong Kong, China, November 2019. Association for Computational Linguistics.

[29] Raj Dabre, Chenhui Chu, and Anoop Kunchukuttan. A Survey of Multilingual Neural Machine Translation. *ACM Comput. Surv.*, 53(5), sep 2020.

[30] Ronald Denaux and Jose Manuel Gomez-Perez. Linked credibility reviews for explainable misinformation detection. In *19th International Semantic Web Conference (ISWC)*, pages 147–163. Springer, 2020.

[31] Ronald Denaux and José Manuél Gómez-Pérez. Sharing retrieved information using linked credibility reviews. In *ROMCIR@ ECIR*, pages 59–65, 2021.

[32] Ronald Denaux, Martino Mensio, Jose Manuel Gomez-Perez, and Harith Alani. Weaving a semantic web of credibility reviews for explainable misinformation detection. In *13th International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 2021.

[33] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics (ACL)*, pages 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics.

[34] Dimitar Dimitrov, Firoj Alam, Maram Hasanain, Abul Hasnat, Fabrizio Silvestri, Preslav Nakov, and Giovanni Da San Martino. SemEval-2024 Task 4: Multilingual Detection of Persuasion Techniques in Memes. In *18th International Workshop on Semantic Evaluation (SemEval)*, SemEval 2024, Mexico City, Mexico, June 2024.

[35] Dimitar Dimitrov, Bishr Bin Ali, Shaden Shaar, Firoj Alam, Fabrizio Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. SemEval-2021 Task 6: Detection of Persuasion Techniques in Texts and Images. In Alexis Palmer, Nathan Schneider, Natalie Schluter, Guy Emerson, Aurelie Herbelot, and Xiaodan Zhu, editors, *15th International Workshop on Semantic Evaluation (SemEval)*, pages 70–98. Association for Computational Linguistics, August 2021.

[36] Renee DiResta. 'Prebunking' Health Misinformation Tropes Can Stop Their Spread. *Wired*, 2021.

[37] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning, 2024.

[38] Jingcheng Du, Sharice Preston, Hanxiao Sun, Ross Shegog, Rachel Cunningham, Julie Boom, Lara Savas, Muhammad Amith, and Cui Tao. Using machine learning–based approaches for the detection and classification of human papillomavirus vaccine misinformation: Infodemiology study of reddit discussions. *J Med Internet Res*, 23(8):e26478, Aug 2021.

[39] Riba Edgar, Mishkin Dmytro, Ponsa Daniel, Rublee Ethan, and Gary Bradski. Kornia: an Open Source Differentiable Computer Vision Library for PyTorch. In *Winter Conference on Applications of Computer Vision*, 2020.

[40] Mattias Ekman. Anti-immigration and racist discourse in social media. *European Journal of Communication*, 34(6):606–618, 2019.

[41] Ashraf Elnagar, Ridhwan Al-Debsi, and Omar Einea. Arabic text classification using deep learning models. *Information Processing & Management*, 57(1):102121, 2020.

[42] OpenAI et al. GPT-4 Technical Report, 2023.

[43] Robert Faris, Hal Roberts, Bruce Etling, Nikki Bourassa, Ethan Zuckerman, and Yochai Benkler. Partisanship, propaganda, and disinformation: Online media and the 2016 us presidential election. *Berkman Klein Center Research Publication*, 6, 2017.

[44] Alessandra Flaccavento, Youri Peskine, Paolo Papotti, Riccardo Torlone, and Raphael Troncy. Automated detection of tropes in short texts. In *The 31st International Conference on Computational Linguistics (COLING), Abu Dhabi, UAE, January 19–24, 2025*, 2025.

[45] Elia Gabarron, Sunday Oluwafemi Oyeyemi, and Rolf Wynn. COVID-19-related misinformation on social media: a systematic review. *Bull World Health Organ*, 99(6):455–463A, March 2021.

[46] Dhruvil Gala, Mohammad Omar Khursheed, Hannah Lerner, Brendan O'Connor, and Mohit Iyyer. Analyzing gender bias within narrative tropes. In *Proceedings of the Fourth Workshop on Natural Language Processing and Computational Social Science*, pages 212–217, Online, November 2020. Association for Computational Linguistics.

[47] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling, 2020.

[48] Michele Gelfand, Ren Li, Eftychia Stamkou, Dylan Pieper, Emmy Denison, Jessica Fernandez, Virginia Choi, Jennifer Chatman, Joshua Jackson, and Eugen Dimant. Persuading republicans and democrats to comply with mask wearing: An intervention tournament. *Journal of Experimental Social Psychology*, 101:104299, 2022.

[49] Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. Chatgpt outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120, 2023.

[50] Kyle Glandt, Sarthak Khanal, Yingjie Li, Doina Caragea, and Cornelia Caragea. Stance Detection in COVID-19 Tweets. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1596–1611, Online, August 2021. Association for Computational Linguistics.

[51] Lucas Graves and Federica Cherubini. The rise of fact-checking sites in europe. *Digital News Project Report*, 2016.

[52] Nir Grinberg, Kenneth Joseph, Lisa Friedland, Briony Swire-Thompson, and David Lazer. Fake news on twitter during the 2016 u.s. presidential election. *Science*, 363(6425):374–378, 2019.

[53] Aditya Grover and Jure Leskovec. Node2vec: Scalable feature learning for networks. In *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855—-864. Association for Computing Machinery, 2016.

[54] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 34(8):3549–3568, 2022.

[55] Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206, 2022.

[56] Raj Gupta, Ajay Vishwanath, and Yinping Yang. COVID-19 Twitter Dataset with Latent Topics, Sentiments and Emotions Attributes, 2022.

[57] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. Exploring network structure, dynamics, and function using networkx. In *7th Python in Science Conference*, pages 11–15, Pasadena, CA USA, 2008.

[58] Max Hänska and Stefan Bauchowitz. Tweeting for Brexit: how social media influenced the referendum. 2017.

[59] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021.

[60] Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. TaPas: Weakly supervised table parsing via pre-training. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online, July 2020. Association for Computational Linguistics.

[61] Brian Hughes, Cynthia Miller-Idriss, Rachael Piltch-Loeb, Beth Goldberg, Kesa White, Meili Criezis, and Elena Savoia. Development of a codebook of online anti-vaccination rhetoric to manage covid-19 vaccine misinformation. *International Journal of Environmental Research and Public Health*, 18(14):7556, Jul 2021.

[62] Maximilian Höller. The human component in social media and fake news: the performance of uk opinion leaders on twitter during the brexit campaign. *European Journal of English Studies*, 25(1):80–95, 2021.

[63] Makarov Ilya, Kiselev Dmitrii, Nikitinsky Nikita, and Subelj Lovro. Survey on graph embeddings and their applications to machine learning problems on graphs. *PeerJ Computer Science*, 2021.

[64] Tunazzina Islam and Dan Goldwasser. Discovering latent themes in social media messaging: A machine-in-the-loop approach integrating llms, 2024.

[65] Amelia Jamison, David A. Broniatowski, Michael C. Smith, Kajal S. Parikh, Adeena Malik, Mark Dredze, and Sandra C. Quinn. Adapting and extending a typology to identify vaccine misinformation on twitter. *American Journal of Public Health*, 110(S3):S331–S339, 2020. PMID: 33001737.

[66] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b, 2023.

[67] Javad Hassannataj Joloudari, Sadiq Hussain, Mohammad Ali Nematollahi, Rouhollah Bagheri, Fatemeh Fazl, Roohallah Alizadehsani, Reza Lashgari, and Ashis Talukder. Bert-deep cnn: state of the art for sentiment analysis of covid-19 tweets. *Social Network Analysis and Mining*, 13(1):99, Jul 2023.

[68] Anna Kata. Anti-vaccine activists, web 2.0, and the postmodern paradigm – an overview of tactics and tropes used online by the anti-vaccination movement. *Vaccine*, 30(25):3778–3789, 2012. Special Issue: The Role of Internet Use in Vaccination Decisions.

[69] Kiana Kheiri and Hamid Karimi. Sentimentgpt: Exploiting gpt for advanced sentiment analysis and its departure from current machine learning, 2023.

[70] Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes, 2021.

[71] Jae Yeon Kim and Aniket Kesari. Misinformation and hate speech: The case of anti-asian hate speech during the covid-19 pandemic. *Journal of Online Trust and Safety*, 1(1), Oct. 2021.

[72] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[73] Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[74] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations, 2020.

[75] Johannes Langguth, Daniel Thilo Schroeder, Petra Filkuková, Stefan Brenner, Jesper Phillips, and Konstantin Pogorelov. Coco: an annotated twitter dataset of covid-19 conspiracy theories. *Journal of Computational Social Science*, Apr 2023.

[76] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[77] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327, 2020.

[78] Darren L. Linvill, Brandon C. Boatwright, Will J. Grant, and Patrick L. Warren. "THE RUSSIANS ARE HACKING MY BRAIN!" investigating Russia's internet research agency twitter tactics during the 2016 United States presidential campaign. *Computers in Human Behavior*, 99:292–300, 2019.

[79] Darren L. Linvill and Patrick L. Warren. Troll Factories: Manufacturing Specialized Disinformation on Twitter. *Political Communication*, 37(4):447–467, 2020.

[80] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for GPT-3? In Eneko Agirre, Marianna Apidianaki, and Ivan Vulić, editors, *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online, May 2022. Association for Computational Linguistics.

[81] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

[82] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9), January 2023.

[83] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach, 2019.

[84] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, 2019.

[85] Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. The Natural Language Decathlon: Multitask Learning as Question Answering, 2018.

[86] Julia Mendelsohn, Ceren Budak, and David Jurgens. Modeling framing in immigration discourse on social media. In *Proceedings of the 2021 Conference of the North American*

Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2219–2263, Online, June 2021. Association for Computational Linguistics.

[87] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: Shedding light on the web of documents. In *7th International Conference on Semantic Systems*, page 1–8, New York, NY, USA, 2011. Association for Computing Machinery.

[88] Martino Mensio and Harith Alani. News source credibility in the eyes of different assessors. In *International Conference for Truth and Trust Online*, 2019.

[89] Areeb Mian and Shujhat Khan. Coronavirus: the spread of misinformation. *BMC Medicine*, 18(1):89, Mar 2020.

[90] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[91] Alistair Miles and Sean Bechhofer. SKOS simple knowledge organization system reference. Working draft, W3C, 2008.

[92] Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. Deep Learning–Based Text Classification: A Comprehensive Review. *ACM Computer Survey*, 54(3), 2021.

[93] Kostadin Mishev, Ana Gjorgjevikj, Irena Vodenska, Lubomir T. Chitkushev, and Dimitar Trajanov. Evaluation of sentiment analysis in finance: From lexicons to transformers. *IEEE Access*, 8:131662–131682, 2020.

[94] Amy Mitchell, Jeffrey Gottfried, Galen Stocking, Mason Walker, and Sophia Fedeli. Many americans say made-up news is a critical problem that needs to be fixed. *Pew Research Center*, 5:2019, 2019.

[95] Martin Müller, Marcel Salathé, and Per E Kummervold. COVID-Twitter-BERT: A Natural Language Processing Model to Analyse COVID-19 Content on Twitter, 2020.

[96] Preslav Nakov, Alberto Barrón-Cedeño, Giovanni da San Martino, Firoj Alam, Julia Maria Struß, Thomas Mandl, Rubén Míguez, Tommaso Caselli, Mucahid Kutlu, Wajdi Zaghouani, Chengkai Li, Shaden Shaar, Gautam Kishore Shahi, Hamdy Mubarak, Alex Nikolov, Nikolay Babulkov, Yavuz Selim Kartal, Michael Wiegand, Melanie Siegel, and Juliane Köhler. Overview of the CLEF–2022 CheckThat! Lab on Fighting the COVID-19 Infodemic and Fake News Detection. In Alberto Barrón-Cedeño, Giovanni Da San Martino, Mirko Degli Esposti, Fabrizio Sebastiani, Craig Macdonald, Gabriella Pasi, Allan Hanbury, Martin Potthast, Guglielmo Faggioli, and Nicola Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 495–520. Springer International Publishing, 2022.

[97] Preslav Nakov, David Corney, Maram Hasanain, Firoj Alam, Tamer Elsayed, Alberto Barrón-Cedeño, Paolo Papotti, Shaden Shaar, and Giovanni Da San Martino. Automated Fact-Checking for Assisting Human Fact-Checkers. In Zhi-Hua Zhou, editor, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4551–4558. International Joint Conferences on Artificial Intelligence Organization, 8 2021. Survey Track.

[98] Preslav Nakov, Giovanni Da San Martino, Firoj Alam, Shaden Shaar, Hamdy Mubarak, and Nikolay Babulkov. Overview of the clef-2022 checkthat! lab task 2 on detecting previously fact-checked claims. In *Conference and Labs of the Evaluation Forum*, 2022.

[99] Usman Naseem, Imran Razzak, Matloob Khushi, Peter W Eklund, and Jinman Kim. COVIDSenti: A large-scale benchmark Twitter data set for COVID-19 sentiment analysis. *IEEE Transactions on Computational Social Systems*, 8(4):1003–1015, 2021.

[100] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022.

[101] Maria Leonor Pacheco, Tunazzina Islam, Lyle H. Ungar, Ming Yin, and Dan Goldwasser. Interactive concept learning for uncovering latent themes in large text collections. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 5059–5080. Association for Computational Linguistics, 2023.

[102] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[103] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[104] Youri Peskine, Giulio Alfarano, Ismail Harrando, Paolo Papotti, and Raphael Troncy. Detecting covid-19-related conspiracy theories in tweets. In CEUR, editor, *MediaEval 2021, MediaEval Benchmarking Initiative for Multimedia Evaluation Workshop, 13-15 December 2021 (Online Event)*, 2021. CEUR.

[105] Youri Peskine, Damir Korenčić, Ivan Grubisic, Paolo Papotti, Raphael Troncy, and Paolo Rosso. Definitions Matter: Guiding GPT for Multi-label Classification. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4054–4063, Singapore, December 2023. Association for Computational Linguistics.

[106] Youri Peskine, Paolo Papotti, and Raphaël Troncy. Detection of COVID-19-Related Conpiracy Theories in Tweets using Transformer-Based Models and Node Embedding Techniques. In *Multimedia Benchmark Workshop*, 2022.

[107] Youri Peskine, Raphael Troncy, and Paolo Papotti. Analyzing COVID-Related Social Discourse on Twitter using Emotion, Sentiment, Political Bias, Stance, Veracity and Conspiracy Theories. In *3rd International Workshop on Knowledge Graphs for Online Discourse Analysis (BeyondFacts)*, 2023.

[108] Youri Peskine, Raphaël Troncy, and Paolo Papotti. Eurecom at semeval-2024 task 4: Hierarchical loss and model ensembling in detecting persuasion techniques. In *Proceedings of the 18th International Workshop on Semantic Evaluation*, SemEval 2024, Mexico City, Mexico, June 2024.

[109] Jürgen Pfeffer, Daniel Matter, Kokil Jaidka, Onur Varol, Afra Mashhadi, Jana Lasser, Dennis Assenmacher, Siqi Wu, Diyi Yang, Cornelia Brantner, Daniel M. Romero, Jahna Otterbacher, Carsten Schwemmer, Kenneth Joseph, David Garcia, and Fred Morstatter. Just another day on twitter: A complete 24 hours of twitter data. *Proceedings of the International AAAI Conference on Web and Social Media*, 17(1):1073–1081, Jun. 2023.

[110] Jakub Piskorski, Nicolas Stefanovitch, Giovanni Da San Martino, and Preslav Nakov. SemEval-2023 task 3: Detecting the category, the framing, and the persuasion techniques in online news in a multi-lingual setup. In Atul Kr. Ojha, A. Seza Doğruöz, Giovanni Da San Martino, Harish Tayyar Madabushi, Ritesh Kumar, and Elisa Sartori, editors, *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2343–2361, Toronto, Canada, July 2023. Association for Computational Linguistics.

[111] Konstantin Pogorelov, Daniel Thilo Schroeder, Stefan Brenner, , Asep Maulana, and Johannes Langguth. Combining Tweets and Connections Graph for FakeNews Detection at MediaEval 2022. In *Multimedia Benchmark Workshop*, 2022.

[112] Konstantin Pogorelov, Daniel Thilo Schroeder, Stefan Brenner, and Johannes Langguth. FakeNews: Corona Virus and Conspiracies Multimedia Analysis Task at MediaEval 2021. In *Multimedia Benchmark Workshop*, 2021.

[113] Konstantin Pogorelov, Daniel Thilo Schroeder, Stefan Brenner, Asep Maulana, and Johannes Langguth. Combining Tweets and Connections Graph for FakeNews Detection at MediaEval 2022. In CEUR, editor, *MediaEval Benchmarking Initiative for Multimedia Evaluation Workshop*, 2022.

[114] Nusrat Jahan Prottasha, Abdullah As Sami, Md Kowsher, Saydul Akbar Murad, Anupam Kumar Bairagi, Mehedi Masud, and Mohammed Baz. Transfer learning for sentiment analysis using bert based supervised fine-tuning. *Sensors*, 22(11), 2022.

[115] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.

[116] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[117] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024.

[118] Ashwin Rao, Fred Morstatter, Minda Hu, Emily Chen, Keith Burghardt, Emilio Ferrara, and Kristina Lerman. Political partisanship and antiscience attitudes in online discussions about COVID-19: Twitter content analysis. *J. Med. Internet Res.*, 23(6):e26692, June 2021.

[119] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics.

[120] Claire Wardle Rory Smith, Seb Cubbon. Under the surface: Covid-19 vaccine narratives, misinformation and data deficits on social media. *First Draft*, 2020.

[121] Mark Rothermel, Tobias Braun, Marcus Rohrbach, and Anna Rohrbach. InFact: A strong baseline for automated fact-checking. In Michael Schlichtkrull, Yulong Chen, Chenxi Whitehouse, Zhenyun Deng, Mubashara Akhtar, Rami Aly, Zhijiang Guo, Christos Christodoulopoulos, Oana Cocarascu, Arpit Mittal, James Thorne, and Andreas Vlachos, editors, *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*, pages 108–112, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[122] Mohammed Saeed. *Employing Transformers and Humans for Textual-Claim Verification.* PhD thesis, Sorbonne Université, 2022.

[123] Mohammed Saeed, Nicolas Traub, Maelle Nicolas, Gianluca Demartini, and Paolo Papotti. Crowdsourced fact-checking at twitter: How does the crowd compare with experts? In *31st ACM International Conference on Information & Knowledge Management*, CIKM '22, page 1736–1746, New York, NY, USA, 2022. Association for Computing Machinery.

[124] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter, 2020.

[125] Michael Schlichtkrull, Yulong Chen, Chenxi Whitehouse, Zhenyun Deng, Mubashara Akhtar, Rami Aly, Zhijiang Guo, Christos Christodoulopoulos, Oana Cocarascu, Arpit Mittal, James Thorne, and Andreas Vlachos. The automated verification of textual claims (AVeriTeC) shared task. In Michael Schlichtkrull, Yulong Chen, Chenxi Whitehouse, Zhenyun Deng, Mubashara Akhtar, Rami Aly, Zhijiang Guo, Christos Christodoulopoulos, Oana Cocarascu, Arpit Mittal, James Thorne, and Andreas Vlachos, editors, *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*, pages 1–26, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[126] Michael Schlichtkrull, Zhijiang Guo, and Andreas Vlachos. Averitec: A dataset for real-world claim verification with evidence from the web. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 65128–65167. Curran Associates, Inc., 2023.

[127] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

[128] Shaden Shaar, Nikolay Babulkov, Giovanni Da San Martino, and Preslav Nakov. That is a known lie: Detecting previously fact-checked claims. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3607–3618, Online, July 2020. Association for Computational Linguistics.

[129] Shadi Shahsavari, Pavan Holur, Tianyi Wang, Timothy R Tangherlini, and Vwani Roychowdhury. Conspiracy in the time of corona: automatic detection of emerging covid-19 conspiracy theories in social media and the news. *Journal of computational social science*, 3(2):279–317, 2020.

[130] Michael Shliselberg and Shiri Dori-Hacohen. Riet lab at checkthat!-2022: Improving decoder based re-ranking for claim matching. In *Conference and Labs of the Evaluation Forum*, 2022.

[131] Felix Stahlberg. Neural machine translation: A review. *Journal of Artificial Intelligence Research*, 69:343–418, 2020.

[132] Hung-Ting Su, Po-Wei Shen, Bing-Chen Tsai, Wen-Feng Cheng, Ke-Jyun Wang, and Winston H. Hsu. Truman: Trope understanding in movies and animations. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 4594–4603, New York, NY, USA, 2021. Association for Computing Machinery.

[133] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to Fine-Tune BERT for Text Classification? In Maosong Sun, Xuanjing Huang, Heng Ji, Zhiyuan Liu, and Yang Liu, editors, *Chinese Computational Linguistics*, pages 194–206, Cham, 2019. Springer International Publishing.

[134] Shardul Suryawanshi, Bharathi Raja Chakravarthi, Mihael Arcan, and Paul Buitelaar. Multimodal Meme Dataset (MultiOFF) for Identifying Offensive Content in Image and Text. In Ritesh Kumar, Atul Kr. Ojha, Bornini Lahiri, Marcos Zampieri, Shervin Malmasi, Vanessa Murdock, and Daniel Kadar, editors, *Second Workshop on Trolling, Aggression and Cyberbullying*, pages 32–41, Marseille, France, May 2020. European Language Resources Association (ELRA).

[135] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

[136] Andon Tchechmedjiev, Pavlos Fafalios, Katarina Boland, Malo Gasquet, Matthäus Zloch, Benjamin Zapilko, Stefan Dietze, and Konstantin Todorov. Claimskg: A knowledge graph of fact-checked claims. In *18th International Semantic Web Conference (ISWC)*, pages 309–324. Springer, 2019.

[137] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul R. Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, and Clemens Meyer et al. Gemini: A family of highly capable multimodal models, 2024.

[138] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

[139] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models, 2023.

[140] Kathie M d'I Treen, Hywel TP Williams, and Saffron J O'Neill. Online misinformation about climate change. *Wiley Interdisciplinary Reviews: Climate Change*, 11(5):e665, 2020.

[141] Kathie M d'I Treen, Hywel TP Williams, and Saffron J O'Neill. Online misinformation about climate change. *Wiley Interdisciplinary Reviews: Climate Change*, 11(5):e665, 2020.

[142] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. Zephyr: Direct distillation of lm alignment, 2023.

[143] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

[144] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2017.

[145] Michela Del Vicario, Alessandro Bessi, Fabiana Zollo, Fabio Petroni, Antonio Scala, Guido Caldarelli, H. Eugene Stanley, and Walter Quattrociocchi. The spreading of misinformation online. *Proceedings of the National Academy of Sciences*, 113(3):554–559, 2016.

[146] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *science*, 359(6380):1146–1151, 2018.

[147] Ben Wang and Aran Komatsuzaki. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax, May 2021.

[148] Claire Wardle and Hossein Derakhshan. Information disorder: Toward an interdisciplinary framework for research and policymaking, 2017.

[149] Derek Weber, Lucia Falzon, Lewis Mitchell, and Mehwish Nasim. Promoting and countering misinformation during australia's 2019–2020 bushfires: a case study of polarisation. *Social Network Analysis and Mining*, 12(1):64, Jun 2022.

[150] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2024. Curran Associates Inc.

[151] Steven Lloyd Wilson and Charles Wiysonge. Social media and vaccine hesitancy. *BMJ Global Health*, 5(10):e004206, Oct 2020.

[152] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Conference on Empirical Methods in Natural Language Processing (EMNLP), System Demonstrations*, pages 38–45. Association for Computational Linguistics, 2020.

[153] BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, et al. Bloom: A 176b-parameter open-access multilingual language model, 2023.

[154] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. Deep Modular Co-Attention Networks for Visual Question Answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[155] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. Opt: Open pre-trained transformer language models, 2022.

[156] Xiaohan Zou. A survey on application of knowledge graph. In *Journal of Physics: Conference Series*, volume 1487, page 012016. IOP Publishing, 2020.