Evidence-based verification and correction of textual claims

James Thorne

Department of Computer Science and Technology

University of Cambridge

This dissertation is submitted for the degree of

Doctor of Philosophy

Peterhouse September 2021

Declaration

This thesis is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the preface and specified in the text. It is not substantially the same as any work that has already been submitted before for any degree or other qualification except as declared in the preface and specified in the text. It does not exceed the prescribed word limit for the relevant degree committee.

James Thorne

September 2021

Evidence-based verification and correction of textual claims

James Thorne

Abstract

This thesis considers the task of fact-checking: predicting the veracity of claims made in written or spoken language using evidence. However, in previous task formulations, modelling assumptions ignore the requirement for systems to retrieve the necessary evidence. To better model how human fact-checkers operate, who first find evidence before labelling a claim's veracity, the methodology proposed in this thesis requires automated systems to retrieve evidence from a corpus to justify the veracity predictions made when modelling this task. The primary contribution of this thesis is the development and release of FEVER, a large-scale collection of human-written claims annotated with evidence from Wikipedia. Analysis of systems trained on this data highlights challenges in resolving ambiguity and context, as well as being resilient to imperfect evidence retrieval. To understand the limitations of models trained on datasets such as FEVER, contemporary fact verification systems are further evaluated using adversarial attacks – instances constructed specifically to identify weaknesses and blind spots. However, as automated means for generating adversarial instances induce their own errors, this thesis proposes considering instances' correctness, allowing fairer comparison. The thesis subsequently considers how biases captured in these models can be mitigated with fine-tuning regularised with elastic weight consolidation. Finally, the thesis presents a new extension to the verification task: factual error correction. Rather than predicting the claim's veracity, systems must also generate a correction for the claim so that it is better supported by evidence, acting as another means to communicate the claim's veracity to an end-user. In contrast to previous work on explainable fact-checking, the method proposed in this chapter does not require additional data for supervision.

Acknowledgements

I am grateful to my supervisor, Andreas Vlachos, for helping me grow and develop my passion for research. Without your encouragement and guidance, I would not have developed to become the researcher I am. Thank you for your advice and support.

Many parts of this thesis results from collaboration with colleagues at Amazon. Without Amazon's generous funding, it would not have been possible to construct the FEVER dataset at nearly the same size or depth. Thank you to my mentors Christos Christodoulopoulos and Arpit Mittal, who have shared their expertise throughout my internships and beyond.

Without the support of the members of the FEVER community, the reach and impact of FEVER would not be what it is. I've enjoyed growing the FEVER workshop as a venue to meet new researchers and engage in discussion that has helped propel the natural language processing formulation for fact verification. Specifically, I would like to thank Preslav Nakov, Tuhin Chakrabarty, Isabelle Augenstein, Sebastian Riedel, and Tal Schuster for being open and engaging with interesting technical discussions.

Finally, I wish to thank my family for their love and support throughout my studies.

Contents

List of figures

List of tables

Chapter 1

Introduction

The widespread challenges of misinformation have had significant societal, political and economic impacts. The ability to distribute information on the internet presents opportunities for individuals and organisations to either intentionally or unintentionally disseminate claims which may be false (Richardson et al., 2003, Budak et al., 2011) with the potential to reach large audiences. This was demonstrated in the 2016 US presidential elections, which highlighted how social networks and news outlets were used to communicate political messages (Allcott and Gentzkow, 2017, Guess et al., 2018). And more recently, in 2020, the Coronavirus pandemic highlighted how misinformation relating to the virus, false cures, and imagined harms of vaccination cost lives (Hughes, 2020). Stimulated by increased public awareness of misinformation and 'fake news', there have been demands made to limit their potential harms (Halevy et al., 2020).

Despite the relatively recent publicity of these challenges, verification is a well-established task in the domain of journalism. It is the core discipline underpinning the writing, editorial and monitoring roles that media outlets perform. Reporters and researchers label the veracity of claims made in written or spoken language by considering evidence, coherence and context (Mantzarlis, 2015). Verification is an intellectually demanding, time-intensive process that is performed by trained professionals. Assessing a single claim can take hours or days depending on its complexity and what information needs

to be collected by fact-checkers (Hassan et al., 2015a). Without automation, manual verification can not scale to the volume of misinformation on the internet or be able to quickly react to newly emerging trends. Consequently, there have been calls from the journalism community to provide automation – reducing the human burden in performing this task (Cohen et al., 2011, Babakar and Moy, 2016, Graves, 2018).

A diverse array of approaches for detecting misinformation have been proposed, which vary in their use of evidence and reasoning process. Notably, Rashkin et al. (2017) predict whether articles are fake news without evidence, and Wang (2017) predicts the veracity of claims using only metadata as supporting information. These approaches are the antitheses of how fact-checkers operate in the journalism domain. It is the use of evidence that informs an end-user how the decision of a claim's veracity is reached: a reader should be able to reach the same judgment of veracity for themselves with the same context. While some previous works do consider supporting information, for example, stance classification (Ferreira and Vlachos, 2016, Pomerleau and Rao, 2017), these tasks only consider the interaction between an article body and headlines and do not consider the need to find appropriate evidence as a human would do.

This thesis presents a computational approach to automate verification following the example of journalists who routinely fact-check reported claims by searching for and reasoning with evidence. Incorporating evidence is required for ensuring transparency in the automated decision-making processes within the context of algorithmic accountability (Diakopoulos, 2015). The primary contribution of this thesis is a novel task formulation and dataset for automated verification that considers the need to retrieve evidence from a corpus of trusted information. Unlike previous work, this enables researchers to evaluate systems based on their prediction of veracity as well as the evidence that is used in this process. Furthermore, once evidence is found, this thesis demonstrates how corrections for claims can be generated without the need for additional training data.

1.1 Thesis structure and contributions

Chapter 2 This chapter describes verification within the context of journalism and surveys natural language processing approaches that model parts of this task. The chapter then introduces a taxonomy, allowing for comparison between previous and current works – specifically highlighting how the use of evidence influences modelling decisions. The chapter subsequently discusses related works, comparing the relation between verification and other research for ensuring integrity on online platforms and in news media.

Chapter 3 This chapter describes the construction of the Fact Extraction and VERification dataset (Thorne et al., 2018a, FEVER): a novel task formulation for evidence-based verification of claims against textual sources where evidence must be retrieved from Wikipedia. During the construction of the dataset, annotators recorded the sentences forming the necessary evidence for their judgement which, in contrast to previous work, enables automated evaluation of the evidence selected by systems in addition to the predicted veracity label. This chapter then introduces modelling approaches for the task with contemporary architectures: the key modelling challenge between this task and previous work is the need to retrieve appropriate evidence before reasoning about the veracity of claims with this retrieved evidence.

Chapter 4 Due to the sensitive nature of predicting claim veracity, it is critical to assess the resilience of these systems to patterns that were not captured within the training data. This chapter evaluates models using adversarial attacks with instances designed specifically to make them predict incorrectly and introduces two novel scoring metrics, *attack potency* and *system resilience* which take into account the *correctness* of the adversarial instances. This aspect is often overlooked in adversarial evaluations. Six fact verification systems from the Fact Extraction and VERification (FEVER) shared task (Thorne et al., 2018b) (the four best-scoring ones and two baselines) are evaluated against adversarial instances generated by

several approaches, including a novel rule-based method proposed in this chapter and paraphrasing models representative of the state of the art. The adversarial instances generated by the rule-based method had a higher correctness rate than previous work, resulting in higher potency during evaluation.

Chapter 5 This chapter investigates how model biases can be mitigated as a domain adaptation task with fine-tuning. Without regularisation, catastrophic forgetting is observed, where parameters required for modelling the fact verification task are overridden during fine-tuning. Experimental results indicate that regularising fine-tuning with Elastic Weight Consolidation (Kirkpatrick et al., 2017, EWC) mitigates these biases while being less susceptible to catastrophic forgetting. In evaluation on fact verification, mitigating hypothesis-only bias, fine-tuning with EWC Pareto dominates standard fine-tuning, yielding models with lower levels of forgetting on the original (biased) dataset for equivalent gains in accuracy on the fine-tuning (unbiased) dataset.

Chapter 6 This chapter proposes the novel task of using evidence to make corrections to claims which are not fully supported by retrieved evidence. This extends the well-studied task of fact verification by providing a method to correct written texts that are refuted or only partially supported by evidence. Furthermore, the corrections serve as a mechanism to communicate the interaction between the claim and evidence to an end-user. This modelling approach uses data from FEVER and a distant supervision training objective.

Chapter 7 This chapter discusses the key findings and impact of this thesis, and how the field has progressed with FEVER. The chapter concludes by highlighting limitations and directions for future work.

1.2 Published works

Chapter 2 Part of this literature review in Chapter 2 was published in the following survey paper:

> J. Thorne and A. Vlachos. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3346–3359, Santa Fe, New Mexico, USA, 2018. Association for Computational Linguistics. URL `https://aclanthology.org/C18-1283`

Chapter 3 The construction process for the FEVER dataset, as well as some of the baselines, are published in the following paper:

> J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana, 2018a. Association for Computational Linguistics. doi: 10.18653/v1/N18-1074. URL `https://aclanthology.org/N18-1074`

Chapter 4 The scoring method for evaluating adversarial attacks and findings on FEVER models were published in the following paper:

> J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal. Evaluating adversarial attacks against multiple fact verification systems. 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 2944–2953, Hong Kong, China, 2019b. Association for Computational Linguistics. doi: 10.18653/v1/D19-1292. URL `https://aclanthology.org/D19-1292`

Chapter 5 The experimental results of mitigating model biases using elastic weight consolidation were published in the following paper:

J. Thorne and A. Vlachos. Elastic weight consolidation for better bias inoculation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 957–964, Online, 2021a. Association for Computational Linguistics. URL `https://aclanthology.org/2021.eacl-main.82`

Chapter 6 The task definition and experimental results from factual error correction were published in the following paper:

J. Thorne and A. Vlachos. Evidence-based factual error correction. 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 3298–3309, Online, 2021b. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.256. URL `https://aclanthology.org/2021.acl-long.256`

Additional Contributions The following publications related to information verification that are not included in this thesis are:

1. Generating token-level explanations of natural language inference tasks with thresholded attention:

   J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal. Generating token-level explanations for natural language inference. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 963–969, Minneapolis, Minnesota, 2019a. Association for Computational Linguistics. doi: 10.18653/v1/N19-1101. URL `https://aclanthology.org/N19-1101`

2. The FEVER shared task and FEVER2.0 build-it break-it fix-it style task, described in the following papers:

J. Thorne, A. Vlachos, O. Cocarascu, C. Christodoulopoulos, and A. Mittal. The fact extraction and VERification (FEVER) shared task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 1–9, Brussels, Belgium, 2018b. Association for Computational Linguistics. doi: 10.18653/v1/W18-5501. URL `https://aclanthology.org/W18-5501`

J. Thorne, A. Vlachos, O. Cocarascu, C. Christodoulopoulos, and A. Mittal. The Second Fact Extraction and VERification (FEVER2.0) Shared Task. In *Proceedings of the second workshop on Fact Extraction and VERification at EMNLP-IJCNLP2019*, Hong Kong, China, 2019c. Association for Computational Linguistics. doi: 10.18653/v1/w18-5501

3. Modelling time-sensitive fact verification as relation extraction, published in the following paper:

J. Thorne and A. Vlachos. An extensible framework for verification of numerical claims. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 37–40, Valencia, Spain, 2017. Association for Computational Linguistics. URL `https://aclanthology.org/E17-3010`

4. An entry for the Fake News Challenge (Pomerleau and Rao, 2017) shared task where the task is modelled as stance classification.

J. Thorne, M. Chen, G. Myrianthous, J. Pu, X. Wang, and A. Vlachos. Fake News Detection using Stacked Ensemble of Classifiers. In *Natural Language Processing meets Journalism workshop at EMNLP 2017*, pages 80–83, 2017. URL `https://aclweb.org/anthology/W/W17/W17-4214.pdf`

External Collaborations The following collaborations, not part of the thesis, have taken place:

1. I contributed data and baseline models for the FEVER portion of the KILT dataset for Knowledge Intensive Language Tasks.

   F. Petroni, A. Piktus, A. Fan, P. Lewis, M. Yazdani, N. De Cao, J. Thorne, Y. Jernite, V. Karpukhin, J. Maillard, V. Plachouras, T. Rocktäschel, and S. Riedel. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.200. URL https://aclanthology.org/2021.naacl-main.200

2. I contributed models for hate-speech detection using multi-task learning.

   Z. Waseem, J. Thorne, and J. Bingel. Bridging the Gaps: Multi Task Learning for Domain Transfer of Hate Speech Detection. In J. Golbeck, editor, *Online Harassment*, pages 29–55. Springer International Publishing, Cham, 2018. ISBN 978-3-319-78583-7. doi: 10.1007/978-3-319-78583-7_3. URL https://doi.org/10.1007/978-3-319-78583-7_3

3. I acted in an advisory capacity for an extension to FEVER that considers verification using tables and text.

   R. Aly, Z. Guo, M. Schlichtkrull, J. Thorne, A. Vlachos, C. Christodoulopoulos, O. Cocarascu, and A. Mittal. FEVEROUS: Fact Extraction and VERification Over Unstructured and Structured information. (NeurIPS 2021), 2021. URL http://arxiv.org/abs/2106.05707

Chapter 2

Automated fact verification

This chapter outlines previous work on automated detection and verification of claims made in natural language. It extends my previous survey paper (Thorne and Vlachos, 2018) with more recent methods and datasets, a new discussion, motivating the task, and a new taxonomy of approaches that considers the requirement for evidence and how the domain influences the method of dataset construction.

2.1 Fact-checking in journalism

Fact-checking is an essential component in the process of news reporting. Journalism has been defined by scholars as a "discipline of verification" to separate it from "entertainment, propaganda, fiction, or art" (Shapiro et al., 2013). While the terms *fact-checking* and *verification* are often used interchangeably, recent literature has defined them as two distinct, but complementary processes (Silverman, 2014). In particular, verification is a common task of evaluating veracity (Mantzarlis, 2015), akin to a "scientific-like approach of getting the fact and also the right facts" (Kovach and Rosenstiel, 2014) and can include, but is not limited to, verifying the source, date and location of materials. Fact-checking, on the other hand, is the application of verification to the journalism domain (Silverman, 2014), requiring practitioners to "address the claim's logic, coherence

and context" (Mantzarlis, 2015) and has become synonymous with the *posthoc* assessment of claims made by politicians and pundits.[1] Consequently, verification is a necessary and crucial first step in the process of fact-checking as it assesses the trustworthiness of the contexts considered. Furthermore, verification can serve other applications beyond the media domain, where the veracity of user-generated, or even *machine*-generated (Holtzman et al., 2020), content requires evaluation.

The increased demand for verification and fact-checking has stimulated rapid progress in developing tools and systems to automate parts of the respective tasks (Babakar and Moy, 2016, Graves, 2018). Thus, there has been a diverse array of approaches tailored to different requirements of journalists and for different types of claims. The agency Full Fact,[2] for example, decomposes the task into four components (Babakar and Moy, 2016): monitoring, spotting claims, checking claims and reporting findings.

Monitoring: Monitoring claims entails building tools that capture and ingest content from the sources that may require fact-checking. These sources, such as television broadcasts, newspapers, and the Twitter accounts of politicians, require normalisation and transcribing speech to text before the content can be verified.

Spotting claims: Where large volumes of text are monitored, it is necessary to *identify* sentences containing claims (Hassan et al., 2015b), *compare* the claims to those which have previously been fact-checked or indicate whether a claim is novel, and *prioritise* which claims to fact-check.

In its simplest instantiation, claim detection or identification is a classification task, labelling whether a sentence is a claim or not, performing the first step in an interpretable fact-checking pipeline proposed by Vlachos and Riedel (2014). One of the earliest works on this topic, ClaimBuster (Hassan et al., 2015b), is applied to sentences captured from

[1] Silverman (2014) and Mantzarlis (2015) differ in their definitions of fact-checking and verification. The former considers fact-checking as an application of verification, whereas the latter considers verification as a *prehoc* and fact-checking a *posthoc* assessment of veracity. While this is an important question, it is tangential to this thesis and is not addressed.

[2] `https://fullfact.org`

political debates and, without context, classifies these sentences as non-fact, important or unimportant factual sentences. Gencheva et al. (2017) extend this work by building a larger dataset and considering the context in which claims are made. Finally, Konstantinovskiy et al. (2018) introduce a schema for the classification of sentences as varying types of claims with seven labels and apply this to annotate statements from political television broadcasts. The claim types captured by this schema are quantities, correlation and causation, law, personal experience, predictions, and other. Despite varying numbers of classes and use of context, the modelling approaches between these three works are similar. Simple machine-learning-based techniques such as linear classifiers and support vector machines were used to predict the class of claim using features extracted from the sentences, such as sentiment, presence of named entities, averaged word embeddings, and metadata of the originator.

Identifying claims that have not been fact-checked acts as a filtering step to minimise duplication of effort by fact-checkers. Repetitions and paraphrases of existing fact-checked claims can be checked against a repository of fact-checking reports (Vlachos and Riedel, 2014, Hassan et al., 2017), considering semantic textual similarity (Agirre et al., 2013). Additional considerations would need to be made for time-sensitive claims made against frequently changing properties, for example, by linking information expressed in claims to structured data sources (Hassan et al., 2014).

Prioritising claims allows the limited resources of human fact-checkers to be best used on the most sensitive or pertinent topics. The classification of important versus unimportant by Hassan et al. (2015b) serves as a simple filtering mechanism, and by further considering the per-class scores (Hassan et al., 2016) enables a mechanism to rank by importance. In practice, however, fact-checkers would need to consider the potential harm that misinformation presents to the public (Hill, 2020), which could range from minimal to a risk to life. This topic remains largely unaddressed.

Checking claims: Babakar and Moy (2016) indicate three distinct approaches to automated checking of claims based on: reference evidence, assumption,[3] and social context. The first step in an automated approach, following the model of manual fact-checkers, is to find supporting material that serves as evidence to verify a claim. Human fact-checkers would find appropriate material from a wide range of sources and use this to support their verdicts. If direct evidence is not available, one would have to make a reasoned prediction of veracity given previous observations and past experiences. The final approach, considering social context, elicits the behaviour of users online and exploits the social dynamics, such as comments or how a claim is shared as an indicator to predict its factuality (Derczynski and Bontcheva, 2015).

Reporting claims: After a claim has been fact-checked, the verdict and findings must be communicated to an end-user. Reports written by professional fact-checkers outline the context, which sources were used, and how the decision-making process influenced the verdict. While reporting and understanding user interactions are beyond the scope of this thesis, additional considerations must be made to communicate the algorithmic decision-making progress effectively. When surveying relevant literature and presenting new approaches, this thesis will consider the need for algorithmic accountability (Diakopoulos, 2015, 2016) which mandates that the design decisions leading to a decision-making process are explained, transparent, and interpretable.

2.1.1 Misinformation, disinformation, and fake news

Two related terms, *misinformation* and *disinformation*, are used when describing factually inaccurate content. While the former refers to directly verifiable claims, where inaccurate or incomplete information is distributed (Guess and Lyons, 2020), the latter additionally assumes malicious motives to mislead the reader (Jowett and O' Donnell,

[3]While this is described as a *machine-learning*-based approach by Babakar and Moy (2016) the three approaches discussed by the authors are all amenable to be modelled in some form with machine-learning. The most likely interpretation of this approach is for systems to make reasoned assumptions in the presence of limited information without an explicit reference for evidence.

2006, Tucker et al., 2018). The term disinformation originates from the Soviet strategy of *dezinformatsia*, where propaganda and political manipulation were directed against the United States (Shultz and Godson, 1984), and is more formally contrasted against misinformation, considering the role of social media, by Guess and Lyons (2020). This thesis will focus on techniques that evaluate the veracity of claims that can be directly verified, identifying misinformation, irrespective of the intention of the author, the users propagating them, or the context they appear in.

One term that has become strongly associated with fact-checking is "fake news" since its use in the context of the 2016 US presidential elections. The most prominent example of such usage is its application to label media outlets of extremely opposing political viewpoints, diluting its meaning to consider aspects not necessarily related to veracity (Vosoughi et al., 2018), which has lead to advice not to conflate fake news with other forms of misinformation (Martens et al., 2018). While there is no unified definition, Zhou and Zafarani (2020) have offered the most concise definition of fake news as false news that a news outlet *intentionally* publishes. While inaccurate content is a part of this problem, the style, presentation, distribution, and social cues in the article, which are beyond the scope of this thesis, also play a significant factor; the most pertinent are discussed as related tasks in Section 2.3.

2.2 A taxonomy for automated fact verification

Many applications and domains share the need to verify information. This multitude of tasks has given rise to a diverse body of works in the natural language processing and social networks research communities. While these works all share a common goal of identifying false or misleading information, these approaches use different definitions of the underlying task and therefore have different requirements. The following discussion highlights the differences between these task definitions and what factors influence the dataset construction, use of evidence, modelling decisions, and how verdicts or justifications are communicated to an end-user.

2.2.1 Domain

The domain influences the availability of data, the dataset construction process, the end-users' expectations of truthfulness, and how the fact-check should be communicated. Previous works can be grouped into the following four categories, which vary with respect to the available types of evidence and requirements for explainability.

News and Politics The strong heritage of claim verification for news and political events has attracted significant research efforts in automating parts of the fact-checking pipeline. Users' expectations for automation in this domain may already be established as fact-checking reports are routinely published. These reports outline the claim's veracity and present a long-form justification that highlights the evidence used. Early datasets (Vlachos and Riedel, 2014, Hassan et al., 2015a, Wang, 2017) have been constructed from these reports, compiled into a common machine-readable format. However, a limitation is that these datasets are capturing only a small part of the fact-checking pipeline, predicting the veracity label, which differs from how human fact-checkers work and how users have become accustomed to seeing veracity communicated.

User-Generated Content Verifying user-generated content, such as social media posts, answers on community forums, or e-commerce reviews, has been studied from different perspectives, including answer accuracy (Zhang et al., 2020, Mihaylova et al., 2018), authenticity (Ott et al., 2011, 2013, Christopher and Rahulnath, 2016), and identifying rumours (Zubiaga and Ji, 2014, Derczynski et al., 2017, Zhou et al., 2019b). User-generated content may have a variety of purposes that differ from the other domains. For example, promoting one's own products, degrading third parties or sharing opinions that unknowingly propagate another actor's rumour. In comparison to the news domain, where verdicts of misinformation may be communicated through lengthy reports, the presence of misinformation in social media posts may be communicated to the author through more concise means, such as information labels, links to authoritative sources displayed alongside the

article, interstitial prompts that make a user think twice before sharing, or may in fact not be communicated, resulting in automatic deletion or demotion of the content (Full Fact, 2019, Halevy et al., 2020).

Scientific Writing The related task of citation recommendation (Ren et al., 2014, Cohan et al., 2019) requires systems to identify appropriate evidence to substantiate claims in scientific articles. While this does not consider the factuality of what's been written, citation recommendation can be used to communicate unsubstantiated claims and appropriate sources to the author. The format of this task is modelled after academic practices, where written reports must reference background literature. Because of the wide availability of academic journals, datasets can be constructed through mining collections of documents without additional annotation. Citation detection has been applied to support authorship tools in Wikipedia, highlighting when citations should be added (Bhagavatula et al., 2018, Redi et al., 2019).

Considering the veracity of claims, Wadden et al. (2020) and Diggelmann et al. (2020) introduce datasets to predict the veracity of claims made in the scientific domain are supported or refuted by evidence. The former considers constructed claims generated from scientific paper abstracts, and the latter considers claims collected from search queries related to climate change. While the source of claims differs between these two datasets, their method of construction, where evidence is selected by annotators, is similar and will be discussed further in Section 2.2.2.

General Domain Verification datasets of factoid claims have been constructed for the purpose of building and evaluating NLP systems. These synthetic verification datasets use the information stored in Wikipedia as the source of evidence, and unlike the other domains, which assume specialist or background knowledge, such knowledge is not assumed. The availability and trust in Wikipedia, coupled with the diversity of articles, allows the datasets to cover a broad range of topics. FEVER, the dataset introduced in Chapter 3, considers claims which are constructed about properties of common entities and verify these selecting evidence and predicting a

veracity label given this evidence. While there was no end-user defined during the construction of this dataset, the annotation task was designed to simulate the role of fact-checkers who find appropriate evidence to support or refute claims.

Even though user expectations between these domains differ, they share a common property where evidence (or lack thereof) can be presented to the user to inform their reasoning. While the domain influences the availability of labelled data and users' expectations, it should not be prescriptive in defining or limiting the information content of the claims – especially with user-generated content, where social media posts can cover multiple domains, from medical topics to current events. The type of reasoning performed by models, as well as the background knowledge required, is also influenced by the domain but is less visible to the user. Transfer between domains presents interesting research challenges where models perform poorly where different patterns are captured in the different respective datasets. For example, experimental results from Wadden et al. (2020) indicate that models pre-trained on FEVER (Thorne et al., 2018a), a general domain fact verification dataset, do not transfer well to claims of scientific facts. Regardless of domain, it is critical to ensure that the models can exploit relevant context: for example, in the news domain, considering current events, and in the scientific domain, considering prerequisite knowledge (Clark, 2015).

2.2.2 Construction

The methods used to construct datasets are influenced by the domain and the availability of pre-existing resources, such as claims or fact-checking reports. Where these resources are not available or insufficient, datasets must be constructed using human annotators, web scraping or alternative means. Construction of the claim, collection of evidence, and annotation of the veracity label will be discussed separately:

Claim Claims can be collected from third-party sources such as fact-checking agencies or manually constructed by annotators. In the news and political domain, datasets that use collected claims are common (Vlachos and Riedel, 2014, Wang, 2017,

Augenstein et al., 2019, Hanselowski et al., 2019) due to the availability of public fact-checking reports. Similar techniques have also been used for building scientific claim verification datasets: Diggelmann et al. (2020) collect and filter claims about climate change by querying a search engine for related keywords and extracting the content of fact-checking sites from the search results. The claims collected in this manner capture a position towards a disputed topic, written by a journalist or researcher, and are an explicit representation of questions that the public may have or topics that readers would find contentious. For citation detection, datasets are constructed by scraping literature containing academic citations (Ren et al., 2014, Bhagavatula et al., 2018). The claims may be implicit and occur within larger sentences with more complex arguments. Citations are used when the author or editor believes facts may be contentious or not widely known, and there is no guarantee that it is representative of an information need that the *reader* may have.

Rather than collecting claims, human annotators can be used to construct short, concise statements for training and evaluating models. In FEVER (Thorne et al., 2018a), annotators were prompted with sentences sampled from popular Wikipedia articles, and in SciFact (Wadden et al., 2020), abstracts from scientific articles are used as source material from which claims were generated. For this method of construction, it is assumed that the original source material is ground truth. In order to build balanced datasets of supported and refuted claims, these positive instances must undergo meaning altering rewrites to generate negative instances.

Using human annotators to generate the negative instances introduces a risk that subconscious biases of the annotators result in patterns with a high degree of association between the claim and the label, meaning that the veracity of a claim can be predicted without the need to consider the evidence. This was first demonstrated for the Stanford Natural Language Inference (Bowman et al., 2015, SNLI) dataset, an application of recognising textual entailment (Giampiccolo et al., 2008, Dagan et al., 2009) where image captions were used as inspiration by

annotators when generating hypothesis sentences that are entailed by, contradicted by, or are neutral with respect to the image caption premise. Gururangan et al. (2018) and Poliak et al. (2018) independently show the impact of this artefact in two separate ways: considering the mutual information between words within the hypothesis and the label, and evaluating the performance of a *hypothesis-only* model, which does not consider the premise sentence, respectively. In FEVER, rather than asking annotators to generate explicit contradictions, the data collection procedure asked annotators to make six types of modifications to claims by replacing entities, making claims more general or specific, paraphrasing, and adding negations (avoiding certain keywords such as *no, not* or *never*). Even with this technique, some common bigrams within claims exhibit a high local mutual information with the veracity label (Schuster et al., 2019), meaning that classifiers may be able to correctly predict the veracity of some claims without the need for evidence and would not be sensitive to changes in world state.

To mitigate claim-only bias in SciFact, Wadden et al. (2020) *specifically trained* annotators to generate negations of true claims, similar to the construction method of SNLI, without inducing a measurable bias. Thorne et al. (2021a) construct a dataset of true and false claims using annotators to rewrite questions with yes or no answers from the BoolQ dataset (Clark et al., 2019) to mitigate this bias. Because the claims are generated from questions that users posed to a search engine, they are representative of real-world information needs. Additionally, as the claim generation was modifying the syntax of the claims without relying on the annotators to change the semantic content, the induction of new claim-only biases is minimal and no more significant than the biases of the users who asked the original questions. To reduce the dependency on annotator imagination when constructing claims, Schuster et al. (2021) use the edit history in Wikipedia to prompt annotators to write pairs of claims for meaning-altering revisions. Annotators write one claim supported by the revised Wikipedia sentence and refuted by the original, and another claim supported by the original sentence and refuted by the revision to

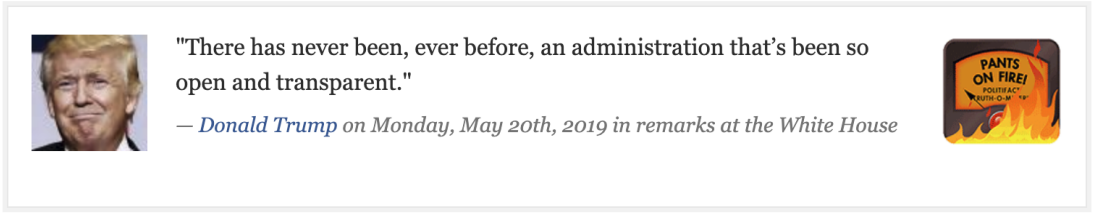


Figure 2.1: Politifact fact-check of a claim made by Donald Trump on May 20th 2019 labelled as 'pants-on-fire'.

Wikipedia. This strategy generates pairs of claims with minimal changes and a high degree of lexical similarity that evaluate fact-verification models' sensitivity to subtle changes in meaning.

Experimental results in Chapter 5 later show that a claim-only bias is also present in datasets of collected political claims, such as Liar (Wang, 2017) and MultiFC (Augenstein et al., 2019). The veracity of statements can be predicted with reasonable accuracy without the need for additional evidence. While these datasets of collected claims do, by definition, consist of real-world claims, this bias is difficult to mitigate as it originates from news agencies, politicians and pundits, on a relatively limited set of topics that are beyond the control of the dataset authors.

Label It is typical for fact-checking reports to provide a label indicating the claim's veracity. Without standardisation, different agencies apply their own labelling scheme. For example, in Figure 2.1 a claim made by Donald Trump was labelled by Politifact as 'pants-on-fire' to indicate that the claim is assuredly false. In most cases, the label schemes are ordinal representations of a claim's veracity ranging from fully supported by evidence to fully refuted, with intermediate labels that indicate half-truths or statements taken out of context. However, not all fact-checking websites publish reports with discrete labels. For example, Full Fact uses a short sentence describing the key findings, illustrated in Figure 2.2, communicating more complex information about the claim's veracity than a label alone can.

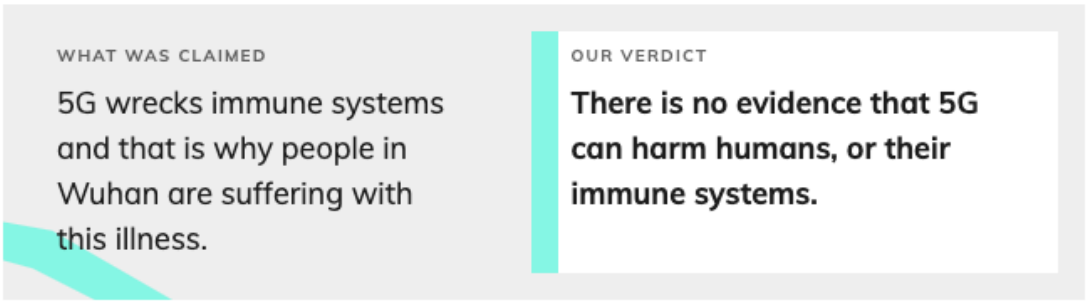


Figure 2.2: A fact-check from Full Fact where a textual description of the claim's veracity is provided rather than a label, published on January 29th 2020.

Datasets of labelled claims can be constructed by simply visiting fact-checking websites and extracting the claims and labels. However, if reports from multiple fact-checking agencies are included in the dataset, the different label schemes and definitions for each label must be explicitly modelled (Augenstein et al., 2019). While textual verdicts are informative to end-users – increasing the transparency of the fact-checking report, they cannot be directly used for generating datasets for claim classification without either using human annotators to discretise the verdicts into labels or modelling the task as explanation generation (Atanasova et al., 2020a, Stammbach and Ash, 2020, Kotonya and Toni, 2020).

For datasets of constructed claims, or where claims are unlabelled, annotators must assign the labels. In previous work, such as FEVER (Thorne et al., 2018a) and SciFact (Wadden et al., 2020), annotators identify evidence and then label whether the claim is supported or refuted given the evidence they find.

Evidence Selection Without evidence, models must memorise facts or predict the likelihood of an event or property *a priori*. It is not possible to capture the entire world's knowledge in a model, and even if it were, the model would be brittle to changes, such as the death of public figures or the election of new officials. Evidence to support veracity prediction can be collected in a number of ways depending on the domain and the underlying modelling assumptions. The approaches surveyed vary in their use of annotators, sources of evidence, granularity, and the potential biases

introduced and can be collected through extraction from fact-checking reports, searching for related pages, and annotating sentences.

Extracted evidence When using claims collected from fact-checking agencies, the fact-checking reports can be used as model inputs (Vlachos and Riedel, 2014, Alhindi et al., 2018, Augenstein et al., 2019). The fact-checking reports contain quotations of and hyperlinks to primary sources as well as the reasoning and rationale used by the fact-checkers to reach their verdict (the primary evidence sources may also be incorporated into the dataset (Hanselowski et al., 2019)). The content of the fact-checking report describes the fact-checker's reasoning process: models may be able to predict the label without the need for the claim by summarising its content. Rather than using the rationale and evidence from the fact-checking report, Wang (2017) only include the claim's metadata, such as originator, their political affiliation, and the media in which the claim was made. While this metadata doesn't explicitly ground the claim, it provides additional context to the model and improves the classification accuracy when predicting the claim's veracity. For citation detection tasks, evidence sources can be trivially extracted from bibliographies of literature.

Searching for evidence In the fact-checking datasets released by Karadzhov et al. (2017) and Augenstein et al. (2019), the claim is input to a search engine, and the related pages are used to provide supporting information to verify the claim. From these returned pages, the relevant passages are selected. For verifying user-generated content on a web forum, Mihaylova et al. (2018) use both search-engine results and related posts from within the forum to aid veracity prediction. This method places the burden on the search engine to return high-quality passages to support the decision-making process. However, without further intervention, it is impossible to discern between sentences related to the claim, discussing it or reporting similar information, and evidence.

Annotation of evidence Evidence can be selected or labelled through annotation, where workers are employed to highlight evidence that supports or refutes claims. Where annotators are paid per task, there are risks of introducing mistakes through workers who are incentivised to quickly perform their duties. Furthermore, annotators' background, viewpoints, and assumed world knowledge can have a measurable impact on the collected data and the resulting models (Geva et al., 2019, Pavlick and Kwiatkowski, 2019, Nie et al., 2020b).

Evidence can be annotated regardless of whether the claims are constructed or collected. However, the search space for evidence is large, and it is intractable to ensure the completeness of annotations. For datasets of constructed claims, such as FEVER (Thorne et al., 2018a), evidence was selected at sentence-level from the introductory sections of Wikipedia articles. Beyond Wikipedia, Wadden et al. (2020) select sentences from scientific paper abstracts for evidence. For datasets of collected claims, Hanselowski et al. (2019) go beyond the annotations collected from fact-checking reports and additionally annotate extracts from the evidence that is reported by the fact-checker, and in ClimateFEVER (Diggelmann et al., 2020) evidence is pre-filtered using an information retrieval system before manually annotating the top five search results.

Evidence by construction Rather than requiring annotators to find and label evidence, datasets with closed-world assumptions are constructed against fixed pieces of evidence. For example, in TabFact, a table-based fact verification dataset (Chen et al., 2020), claims are constructed using a table as evidence. As the table was provided to annotators during claim writing and is known in advance, there is no need for evidence to be retrieved and annotated. A similar property holds with Ciampaglia et al. (2015), who verify claims, represented as atomic subject-predicate-object triples, using the topology of knowledge graphs. As knowledge graphs are a machine-readable representation, it is possible to generate triples that are grounded against that graph without the need for annotators.

2.2.3 Inputs

This section reviews the inputs to fact verification systems, what is claimed and how the claims are represented, and discusses their influence on the modelling. The review focuses on five aspects: modality, granularity, context, claim type, and use of evidence.

Modality Claims can be made in different modalities, including text, speech, and images. This thesis evaluates fact verification from a natural language perspective: considering documents such as news articles and single sentences such as claims made by politicians or pundits. Even when the claims originate in other media, such as audio, they can be transcribed into text (Hassan et al., 2015b).

Aside from using text, a frequently considered input is structured data in the form of subject-predicate-object triples, e.g. `(London, capital_of, UK)`. These triples facilitate fact-checking against knowledge bases such as Freebase (Bollacker et al., 2008) and have been considered by different research communities, including NLP (Nakashole and Mitchell, 2014), data mining (Ciampaglia et al., 2015) and Web search (Bast et al., 2015). Rather than predicting the veracity of textual claims, Ciampaglia et al. (2015) predict the likelihood of a knowledge graph edge between two nodes. To apply this synthetic task variant to real-world settings assumes that the entities, properties and relations can be grounded to the canonical representation used in the graph. This implicitly assumes a non-trivial level of processing in order to convert text, speech or other forms of claims into triples.

While some textual claims from the political domain could be represented as triples, they typically require more complex representations; for example, events typically need to be represented with multiple slots (Doddington et al., 2004) to denote their various participants. Regardless of whether textual claims are verified via a subject-predicate-object triple representation or as text, it is often necessary to disambiguate the entities and their properties. For example, the claim 'Beckham played for Manchester United' is true for the soccer player 'David Beckham', but not true (at the time of writing) for the American NFL player 'Odell Beckham

Jr'. Correctly identifying, disambiguating and grounding entities is the task of Named Entity Linking (Mcnamee et al., 2009). While this task must be performed explicitly if converting a claim to a subject-predicate-object triple with reference to a knowledge base, it may also be performed implicitly through the retrieval of appropriate evidence if fact-checking against textual sources (De Cao et al., 2020).

In images and videos, there are additional challenges that differ from text, including verifying the image's source, context and identifying doctoring. Claims present in an image's caption or overlaid text, extracted using optical character recognition, can be verified independently or using a model that considers the interaction between the image and the text (Kiela et al., 2020, Nakamura et al., 2020). For images that do not contain text, their use may be misleading depending on the context where the images are used. Two early works on detecting *fauxtography* (Zhang et al., 2018, Zlatkova et al., 2019) exploit social reactions to images such as comments and reverse image search, respectively, to resolve whether the image appears in the correct context. For video, further challenges remain, combining the modalities of voice, text, and images over time.

Granularity For each modality, verification can take place at different levels of granularity. With text documents, for example, sentences containing claims could individually undergo verification, similar to the role performed by human fact-checkers, or the entire document could be used as input, considering its discourse. Recent fact verification datasets (Wang, 2017, Karadzhov et al., 2017, Thorne et al., 2018a, Augenstein et al., 2019, *inter alia*) consider individual textual claims as input. To apply models trained on these data to longer documents such as news articles or debate transcripts, which could contain multiple claims, a pipeline of claim identification can serve as a filtering step prior to verification (Babakar and Moy, 2016). Hassan et al. (2014) identify claims using a supervised classifier trained to predict check-worthiness. Alternatively, rhetorical structure (Palau and Moens, 2009, Levy et al., 2014) can be exploited to identify statements or arguments that

could be checked, similar to its application in detecting deceptive language (Rubin and Vashchilko, 2012, Rubin and Lukoianova, 2014) where the consistency and relation between argumentation units are considered. Finally, Li and Zhou (2020) apply a text summarisation system to news articles and perform claim verification over the generated summaries. Without extracting claims or identifying arguments, predicting the veracity of news can be reduced to a document classification task (Rashkin et al., 2017) based on linguistic features such as emotive language.

Claim Type There may be additional restrictions on the types of claims that can be checked. For example, with verifying textual claims, previous works have mainly considered properties of events, both in the general domain (Thorne et al., 2018a), scientific domain (Wadden et al., 2020, Diggelmann et al., 2020), for user-contributed answers to community questions or product questions (Mihaylova et al., 2018, Zhang et al., 2020), and in some instances for claims in the political domain (Wang, 2017, Karadzhov et al., 2017, Augenstein et al., 2016b). However, what is fact-checked in textual claims can go beyond trivial properties. Entity and event properties was only one of the four claim types considered in the HeroX fact-checking challenge,[4] with the other three being claims involving numerical comparison, position statements (whether a political entity is in support or against a policy), and quote verification (assessing whether a quote is accurate in source, content, and context). For claims in the political and news domain, Konstantinovskiy et al. (2018) formalise a schema of claim types, introducing seven broad categories which such as: verification of laws, making predictions, personal experiences and correlation/causation. Across these seven categories, their schema further breaks claim types into 19 subcategories that each require different verification strategies.

For graphs and structured data, similar restrictions on the types of claims exist that inform the verification method. For simple atomic facts, checking the existence of a relation or property may be sufficient, and for numerical claims, percentage error

[4] `https://www.herox.com/factcheck`

may be required for reasoning (Vlachos and Riedel, 2015, Thorne and Vlachos, 2017). For properties that are not captured as a single atomic relation, checking the likelihood of a path (Ciampaglia et al., 2015) or validating a fact against the graph's constraints (Kim and Choi, 2020) may be required.

For images, different types of verification may need to be performed dependent on what is claimed. Verification would require checking whether the image is used out of context (Zhang et al., 2018), considering the consistency of media alongside an article or post (Zlatkova et al., 2019), evaluating whether the image is manipulated or doctored (Fridrich et al., 2003, Avcibas et al., 2004, Popescu and Farid, 2005), or verifying the extracted text in isolation or considering multiple modalities.

Context Claims may have a time-dependent or context-dependent meaning that alters the meaning. For example, with temporal dependence, a claim such as the President is a Democrat would hold in the United States, at the time of writing, since the election of Joe Biden, but not for the time when Donald Trump was presiding. The verification system must consider the state of the world at the time and the place the claim was made. While Wang (2017) provides metadata detailing the time, claim originator and political affiliation to resolve context, other datasets are much more limited in this regard. Gencheva et al. (2017) include the context in which claims are made in political debates through providing and modelling conversation history and speaker but not affiliation or time.

In datasets where evidence and context are not explicitly provided, the systems are required to resolve the context. The way systems resolve evidence may yield different predictions of claim veracity considering different perspectives (Chen et al., 2019). For example, the claim *Beckham is a football player* would hold true for both David Beckham, the British soccer player, as well as Odell Beckham Jr, the American football player. For a claim that considers which *team* Beckham plays for, different veracity predictions could be reached depending on the context of the evidence that the system retrieves from its knowledge source. While the entity in

the claim undergoing verification is explicitly (i.e. not referred to using a pronoun) mentioned in some datasets, it may not necessarily refer to a unique entity. Even within the context of sports, the name Beckham is ambiguous as the same surface form resolves to different pages on Wikipedia.

2.2.4 Evidence

Task signatures that incorporate evidence retrieval or search enable fact verification systems to be applied to claims that are not captured within datasets used to train or evaluate the systems. If evidence is provided explicitly as an input, for example, in the setting used by the Fake News Challenge (Pomerleau and Rao, 2017), application to real-world settings would require an external component to retrieve it at run-time.

When modelling veracity prediction as a classification task without evidence, features in the claims alone are associated with the predicted veracity rather than considering the world state (Rashkin et al., 2017). In contrast, when journalists are fact-checking, they must find knowledge relating to the fact and evaluate the claim given the evidence and context when deciding whether it is true or false. While these features may provide indications of deceptiveness or satire, they do not necessarily indicate a claim's veracity as true information can be expressed in a satirical writing style, and misinformation can be written with an authoritative writing style tricking the reader into believing an article is credible. Fake news articles generated using Grover (Zellers et al., 2019), an adversarially trained language model with controllable style, deceived human raters into believing that its articles were more trustworthy than manually written fake news. Similarly, a study into stylometric analysis (Schuster et al., 2020) found that stylometry alone is insufficient to distinguish between verified claims and misinformative content generated from a pre-trained language model.

The use of large-scale language models for fake news detection and fact verification has been explored by Zellers et al. (2019) and Lee et al. (2020). Language models predict the likelihood of observing a sequence of tokens, and it is hypothesised that, if

trained on sufficient data covering world knowledge, the model could be used to identify misinformation. While large-scale language models do capture facts and knowledge (Petroni et al., 2019), verifying claims by decoding tokens from a language model exploits surface-level features in language that are decoupled from grounded communicative intents (Bender and Koller, 2020). Furthermore, as the state of the world changes, the description of the world encoded within the model parameters would also require updating. Changing model parameters without side-effects would require identifying a *minimal* set of parameters to be updated to incorporate new knowledge. However, this may prove difficult as several studies, surveyed by Rogers et al. (2020), indicate contributions of *many* layers and attention heads for classification.

Rather than memorising world knowledge and facts, journalists find authoritative sources relating to the fact and evaluate the claim given the evidence and context. This could either be a repository of previously fact-checked claims or a third-party information source: Hassan et al. (2017) integrate both methods in the ClaimBuster automated fact-checking platform. Comparing new claims to those that have been previously fact-checked is an instance of searching by semantic textual similarity (Agirre et al., 2013) and evaluating claims with new evidence is an instance of stance classification (Ferreira and Vlachos, 2016) or recognising textual entailment and contradiction (de Marneffe et al., 2008, Giampiccolo et al., 2008, Dagan et al., 2009).

As discussed in Section 2.2.2, there are a number of modalities for claims aside from text. Just as the modality of claims influences the modelling choices, the modality of evidence impacts how the models assign veracity labels to claims too:

Knowledge Graphs Knowledge graphs provide a rich collection of structured canonical information about the world stored in a machine-readable format that could support the task of fact-checking. Two task formulations using this evidence media have been explored: direct observation of atomic triples and exploiting graph topology. *Direct observation* selects or retrieves a triple from the knowledge graph that provides the information supporting or refuting the claim. For example, Vlachos and Riedel

(2015) and Thorne and Vlachos (2017) identify the subject-predicate-object triples from tables parsed as small knowledge graphs to fact-check numerical claims. Once an evidential triple has been found, a veracity label is assigned by considering exact match or rule-based reasoning over the difference between what is claimed and what is retrieved. The key limitations in using knowledge graphs as evidence are that it assumes the exact triple is present, the entity can be correctly grounded, and the reasoning is simple enough to be expressed using rules. However, it is not feasible to capture and store every conceivable fact in the graph in advance of fact-checking and entity grounding is a necessary pre-processing step. The alternative use of a knowledge graph as evidence is to consider its *topology*, such as semantic proximity (Ciampaglia et al., 2015), use of predicate-types (Shi and Weninger, 2016), network flow (Shiralkar et al., 2017), and rule induction (Kim and Choi, 2020). However, while these features can indicate the plausibility of a fact, ruling-out misinformation, they alone are not sufficient to assert truthfulness.

Social Signals Aggregate information on the distribution of posts on social networks can provide insights into the veracity of the content. Rumour veracity prediction is the assessment of the macro-level behaviours of users' interactions to predict whether the claims in posts are true or false (Derczynski and Bontcheva, 2015, Derczynski et al., 2017). This crowd behaviour provides insights into claim veracity, especially in cases where textual sources or structured knowledge bases may be unavailable. Signals such as tweet content (Zubiaga and Ji, 2014), responses (Lukasik et al., 2016), context (Zubiaga et al., 2017) conversation structure (Wei et al., 2019) have been evaluated. Rather than predicting veracity, these approaches identify credibility indicators for the tweets (Liu et al., 2015) or media outlets publishing an article (Nakov, 2020) without necessarily considering the content.

2.2.5 Outputs

This section considers which outputs of fact verification systems can be returned to end-users and how these outputs can be incorporated into the evaluation of systems.

Label The simplest model for fact-checking is to label a claim as true or false as a binary classification task (Nakashole and Mitchell, 2014). However, claims can have varying degrees of truthfulness. Journalistic fact-checking agencies, such as Politifact, model the degree of truthfulness on a multi-point scale (ranging from true, mostly-true, half-true, etc.). In datasets that capture journalistic fact-checking, Wang et al. (2018), Rashkin et al. (2017), Karadzhov et al. (2017) and Augenstein et al. (2019) model the task as multi-class classification, using the labels assigned by fact-checkers. Similarly, Vlachos and Riedel (2014) suggested modelling this degree of truthfulness as an ordinal classification task.

The reasoning behind the choice of why fine-grained labels are assigned by fact-checkers is complex and differs between agencies.[5][6] Even though different fact-checking agencies apply different labels under different protocols, Augenstein et al. (2019) indicate that modelling choices, such as the use of label embeddings (Augenstein et al., 2018), can be made to exploit the overlap for similar label types.

For stance classification (Ferreira and Vlachos, 2016), the task is also a multi-class classification: predicting whether a claim is supported, refuted or observed by a news article headline. In the fake news stance classification task, Pomerleau and Rao (2017) added an extra label for when the article was unrelated to the claim, which could be used to support the filtering of documents. While not directly verifying claim content, similar label schemas are used for stance classification of rumours in user-generated social media content (Castillo et al., 2013).

[5] `https://www.snopes.com/fact-check-ratings/` [Accessed 18th June 2021]

[6] `https://www.politifact.com/article/2018/feb/12/principles-truth-o-meter-politifacts-methodology-i/` [Accessed 18th June 2021]

Similar to stance classification, for fact verification for claims in the general and scientific domains, Thorne et al. (2018a), Wadden et al. (2020) and Diggelmann et al. (2020) use a 3-way label schema considering whether the claim is supported or refuted by evidence. The third class is also used to label if there is not enough information to certainly label the claim either way. This third label can be applied by a classifier, considering retrieved evidence, or by the system if no relevant evidence can be found. This common label schema allows comparison of models and transfer-learning between datasets for claim verification in different domains (Wadden et al., 2020, Thorne et al., 2021a) as well as between stance classification models (Suntwal et al., 2019). Where evidence is annotated, the predicted veracity label can be conditionally scored dependent on whether retrieved evidence matches the annotated evidence for the claim.

The triple scoring task of the WSDM cup (Bast et al., 2017) was a regression task rather than a classification task. Systems assigned a score within a numerical range indicating how likely it is that the triple is true. The evaluation consisted of two components: calculating the differences between the predicted and the manually annotated scores, as well as the correlation between the rankings produced by the systems and the annotators.

Explanation Fact verification has sensitive applications, and thus it is essential that systems are *right for the right reasons* (Ross et al., 2017, McCoy et al., 2019, Stammbach and Ash, 2020), expressing why a decision was reached to the end-user (a property that is also indicated in EU law under the right to explanation: GDPR Recital 71). Being able to interpret system predictions is essential, not only for scientific understanding of the systems, but also under an ethical obligation to people affected by these predictions (Doshi-Velez and Kim, 2017). Transparency in design decisions, data collection and modelling all provide mechanisms for accountability (Diakopoulos, 2015, 2016) and can be communicated to the end-user using a model card (Mitchell et al., 2019), or as part of a system's output.

Explanations for fact verification have taken different forms, including generating textual descriptions of the reasoning process with supervision (Atanasova et al., 2020a, Kotonya and Toni, 2020) or from a language model (Stammbach and Ash, 2020), extracting evidence from a corpus (Thorne et al., 2018a), linking claims to users' comments (Shu et al., 2019), and generating corrections for claims (Thorne and Vlachos, 2021b). While interpretable machine learning provides insights into what influenced a classification, this family of techniques primarily acts as a diagnostic for the classifier, indicating which features, such as tokens, are important (Ribeiro et al., 2016, Aubakirova and Bansal, 2016, Linzen et al., 2016, Li, 2016) to the model and may not correspond to the features that end-users deem important (Nguyen, 2018, Thorne et al., 2019a).

Fact-checking agencies generate reports outlining which evidence is considered and how this influenced the checker's decision-making with respect to the claim. Explainable fact verification systems should aim to emulate this part of the established fact-checking pipeline (Babakar and Moy, 2016), discussed in Section 2.1. The techniques such as returning evidence or generating descriptions of a reasoning process are features built into the dataset design and modelling process that enable the system to inherently generate an output that is more similar to what a user would expect a human fact-checker to write. While datasets and tasks may allow automated evaluation of these explanations, understanding whether an explanation generated by a model or system is acceptable for an end-user extends into the study of user interaction and depends on the target user's demographic and prior knowledge as well as social context.

2.3 Related tasks

The following related tasks apply machine learning and natural language processing to protect the integrity of media and communications. Most tasks listed here are supervised

sentence or document classification tasks. Without considering evidence, these tasks describe orthogonal properties of texts without necessarily verifying the factuality.

Subjectivity and emotive language detection Rashkin et al. (2017) assess the reliability of entire news articles by predicting whether the document originates from a website classified as Hoax, Satire or Propaganda. This work is an instance of subjective language detection and does not represent evidence-based fact-checking. The authors used supervised classifiers augmented with lexicons, including the Linguistic Inquiry and Word Count (LIWC) (Pennebaker et al., 2015), a sentiment lexicon (Wilson et al., 2005), a hedging lexicon (Hyland, 2015), and a novel 'dramatic' language lexicon to identify emotive, subjective and sensational language in the article bodies. The most distinguishing features for classifying unreliable news articles included the use of hedge words (such as 'reportedly') or words pertaining to divisive topics (such as 'liberals' or 'Trump') and the addition of the LIWC lexicon, which provides an indication of the emotive tone and authenticity, marginally improved the classification accuracy.

Deceptive language detection Linguistic cues and features help identify deception (Zhou et al., 2004). Mihalcea and Strapparava (2009) model the task as classification using words as features without further feature engineering. Analysis of their model identifies a number of word classes of the LIWC lexicon (Pennebaker et al., 2007) which pertain only to the deceptive texts and also found that truthful texts were more likely to contain words belonging to the 'optimistic' LIWC class such as 'best', 'hope', and 'determined'. Ott et al. (2011) identify three aspects to deceptive language detection: genre classification, psycholinguistic analysis (using lexicons such as LIWC), and imaginative writing detection; finding that highly-weighted features of deceptive language classifiers were common with classifiers trained for imaginative writing detection. Subsequent studies have considered stylometry (Feng et al., 2012) and sentiment (Ott et al., 2013). Hai et al. (2016) also identify deceptive reviews using word-based features. In contrast to Mihalcea and Strapparava (2009), who rely on labelled data, the authors induce labels through a semi-supervised learning approach that exploits a minimal amount of labelled

data from related tasks in a multi-task learning setting. Even though linguistic content, emotive language and syntax are useful indicators for detecting deceit, the veracity of a statement cannot be determined without considering whether or not it is supported by evidence.

Propaganda detection Communications designed to influence audiences are often associated with fake news and disinformation. Automated methods for detecting propaganda model the task as text classification, either performed at a document level or a fine-grained level. Document-level propaganda detection depends on identifying stylistic features, such as emotive language (discussed earlier). In contrast, fine-grained techniques identify argumentation structures used in propaganda and neuro-linguistic programming (Torok, 2015, Da San Martino et al., 2019) at a sentence or fragment level.

Click-bait detection Intentionally misleading headlines or titles that are designed specifically to encourage a user to click through and visit a website are called click-bait. Studies into the detection of click-bait have yielded positive results from relatively simple linguistic features (Chen et al., 2015, Potthast et al., 2016, Chakraborty et al., 2016). However, these only consider the article headline and do not make use of evidence. In contrast, the task of detecting headlines that are incongruent to the document body (Chesney et al., 2017), is similar to the stance classification task which was adopted by the Fake News Challenge (Pomerleau and Rao, 2017). Using the responses on social networks, including the content of comments and response time, Glenski et al. (2018) predict whether websites are shared deceptively (i.e. click-bait) or whether the content is shared with genuine intent.

Rumour detection Rumour detection (Qazvinian et al., 2011) is the task of identifying unverified reports circulating on social media. From an NLP perspective, the content of posts and tweets, such as the choice of words, emojis and hashtags, serve as features in supervised classification (Tam et al., 2018). Alternatively, the growth of readership through a social network and share patterns (Mendoza et al., 2010), third party reporting

(Procter et al., 2013), and user responses (Zhang et al., 2015) act as indicators to whether a post is a rumour. While these are important factors to consider, a sentence can express a true or false claim independent of whether it is a rumour (Zubiaga et al., 2018).

Speaker profiling Identifying claims that do not fit with the originator's profile could predict veracity: determining the behaviour of a user allows for a risk-based approach to fact-checking. Long et al. (2017) introduced the notion of credit history that incorporates reputation into fake news detection. Relying only on the source's overall trustworthiness score may be problematic as inaccurate information may be found even on the most reputable sources. Furthermore, this approach does not account for which topics the originator persistently lies about.

For predicting the credibility of users on web forums, Mihaylova et al. (2018) use the reputation history of an author, such as reader-assigned feedback, as an indicator to predict veracity. Castillo et al. (2011) and Mu and Aletras (2020) construct profiles for users on Twitter, considering the reliability of news sources that are shared. Users are labelled as reliable or not based on the number of unreliable tweets made and used to identify topics that correlate with user reliability. Inversely, Feng and Hirst (2013) apply profiling to aid deceptive review detection in product reviews by creating an average profile for products rather than of reviewers. Building these profiles suffers from a cold-start problem: for new topics or users, insufficient data may prohibit the construction of an accurate profile.

Pérez-Rosas and Mihalcea (2015) identify author characteristics (such as age and gender) that influence the linguistic choices made by the authors when fabricating information in product reviews. The affiliation, age and gender of most politicians is publicly available; it is conceivable that these features may be incorporated and assist the verification of political claims. The use of such meta-data does improve the classification accuracy for the LIAR! fact-checking task (Wang et al., 2018, Long et al., 2017). However, a claim originator deviating from their profile does not always imply misinformation, and regressing to the average profile does not necessarily imply truth.

Evidence credibility and reputation Where external evidence is used for verification, its accuracy must be trusted. Methods of predicting the authoritativeness of web pages such as PageRank (Brin, 1998), which considers hyperlink topology, have a strong heritage in information retrieval and web search to yield higher quality search results. While this only considers the hyperlink topology, variants such as TrustRank (Gyöngyi et al., 2004) provide a framework that incorporates annotated information to predict the trustworthiness of pages based on graph-connectedness known-bad nodes rather than the information content. Beyond topology, knowledge-based scoring (Dong et al., 2015), predict the trustworthiness of a source, given how facts extracted from it compared to a graph of ground truth facts as an indicator for credibility.

2.4 Summary

This chapter introduced the tasks of verification and fact-checking from the domain of journalism. Both these tasks serve utility in identifying misinformation but are performed at considerable human cost (Hassan et al., 2015a) and have been subject to calls for automation (Babakar and Moy, 2016). Automated system development and evaluation has been supported by the introduction of many large-scale datasets which vary in their domain and construction. The choice of construction method impacts the scale at which data can be collected and annotated, the potential biases that can be introduced, as well as the reasoning process and how evidence is incorporated.

Within the natural language processing community, many approaches aim to predict the veracity of claims represented in text, images and other modalities such as knowledge graph triples. However, their use of evidence and task-specific training data varies. Modelling the prediction of a claim's veracity only as text classification, without external knowledge, is insufficient as this requires the model to memorise facts or exploit surface-level features that are not predictors of veracity. Incorporating external information with repositories of previously fact-checked claims or external evidence allows a model to reason about the state of a continuously changing world and phenomena that were

not observed during training. Human fact-checkers apply the same reasoning process (Babakar and Moy, 2016), finding reference material to support the verification of claims before presenting a report of the claim's veracity to the end-user. The researching process taken by fact-checkers is resource-intensive and acts as a bottleneck that limits how quickly fact-checkers can respond to newly arising misinformation (Hassan et al., 2015a). Systems that are able to retrieve appropriate evidence to predict the veracity of claims may serve utility in assisting this process.

Predictions of veracity or trustworthiness can be used to label, prioritise or delete content on social media platforms (Halevy et al., 2020). Therefore it is crucial to ensure algorithmic accountability (Diakopoulos, 2015): where systems performing decision-making are transparent in their processes. Beyond outputting a single label, model explanations serve as just one mechanism to communicate features of importance. However, these act as a diagnostic of the model and may not always align with features that an end-user would deem important. In contrast, generating appropriate explanations is dependent on the system's purpose, the demographics of the end-user and the context in which the claim is made. Two sources of explanation that capture some part of a user's decision-making process are (1) highlighting and returning a limited set of appropriate evidence spans retrieved from corpus, which acts as an extractive summary of the veracity of the claim, and (2) further training to generate abstractive summaries of evidence, explaining why a veracity decision is predicted. Both explanation sources are captured during dataset construction: the first being an artefact of using annotators to select evidence to verify a claim, and the second being an artefact of collecting fact-checking reports as training data, which have a communicative intent of informing an end-user as to how the evidence impacts a claim's veracity.

Chapter 3

Constructing a dataset for fact extraction and verification

This chapter introduces the Fact Extraction and VERification dataset (FEVER), which was constructed under the advice of external collaborators Christos Christodoulopoulos and Arpit Mittal and published as Thorne et al. (2018a). With their advice, I formulated the task definition. My contributions also include the design of the data collection and quality control procedures, dataset construction, analyses, and modelling. This chapter extends the description of the dataset and task presented in the original paper. It additionally presents a new suite of experiments with contemporary retrieval methods and classifiers with a more detailed analysis of how modelling choices impact the performance. The combination of higher document recall (+39 percentage points) and label accuracy with retrieved evidence (+36 percentage points) result in a FEVER Score of 69%, which is at least double that published in 2018.

3.1 Introduction

This chapter introduces a novel task formulation for claim verification that requires models to retrieve appropriate evidence from a corpus of encyclopaedic knowledge that

FEVER Claim: Bullitt is a movie directed by Phillip D'Antoni
Evidence Source: `https://en.wikipedia.org/wiki/Bullitt, sentence 0` **Evidence Text:** Bullitt is a 1968 American action thriller film directed by Peter Yates and produced by Philip D'Antoni
Label: REFUTED

Figure 3.1: Example of instance from the FEVER dataset showing a claim that is refuted by a single sentence extracted from a Wikipedia page

supports or refutes a claim. The Fact Extraction and VERification (FEVER) dataset consists of 185,445 manually constructed claims, annotated with evidence from relevant Wikipedia pages that either SUPPORTS or REFUTES the claims. If no evidence could be found by the annotators, the claim is labelled as NOTENOUGHINFO without evidence. An instance from the FEVER dataset, illustrated in Figure 3.1, contains a claim and annotations of which sentences from Wikipedia were highlighted as evidence to justify the assigned label.

Predicting the veracity of a claim is an extension of several related tasks: recognising textual entailment (Giampiccolo et al., 2008, Dagan et al., 2009), natural language inference (Bowman et al., 2015), and identifying contradiction (de Marneffe et al., 2008). These text-pair classification tasks require models to predict whether a hypothesis sentence is entailed or contradicted by a premise sentence. For example, the sentence "A man walks his dog in the park" entails that "The man is outside". Verifying claims c against evidence E, $f(c, E) \in \{Supported, Refuted, NotEnoughInfo\}$, has the same task signature (a multi-class text-pair classification) as these previous works, allowing existing model architectures to be extended for fact verification. Like Natural Language Inference, the SUPPORTED and REFUTED labels are applied considering the interaction between a hypothesis posed in the claim and a premise from the retrieved evidence. The SUPPORTED label should be applied when the average person would have reason to believe

the claim is *true* given the evidence, and the REFUTED label should be applied when the average person would have reason to believe the claim is *false* given the evidence.[1]

A novel feature of FEVER is the need for systems to retrieve evidence from Wikipedia before predicting the claim's veracity label. The task formulations introduced in Chapter 2, in contrast, consider evidence that is provided as an input to the model, which is an unrealistic task formulation as it does not reflect the challenges of identifying appropriate source material to verify a claim. Formally, given the claim, a system must identify evidence at a sentence level from a pre-processed corpus of Wikipedia pages, $\mathcal{W}$. For example, a system may identify relevant pages $W \subset \mathcal{W}$ and then extract evidence at a sentence level from these pages, $E \subset \bigcup_{w \in W} \{e \in w\}$.

Because FEVER contains annotations of which sentences in Wikipedia are evidence for a claim, systems can be evaluated for both predicting the correct label and retrieving the correct evidence. This evidence acts as a means of supporting the model's prediction of the claim's veracity and acts as a diagnostic of whether models are using the correct evidence when making predictions. To characterise the challenges posed by this task formulation, Section 3.5 presents a suite of experiments considering evidence retrieval and veracity classification. Both sub-tasks are evaluated independently and in combination to study how errors from evidence retrieval propagate to the claim verification task.

3.2 Annotation procedure

The dataset was constructed through two separate data collection processes: claim generation and evidence labelling, illustrated in Figure 3.2. The full guidelines and screenshots of both annotation tasks are provided in Appendices A.1 and A.2. For the claim generation process, annotators were presented with randomly sampled sentences from Wikipedia and asked to extract factoid statements from the sentence called claims. For each claim, the same annotator was asked to make meaning altering changes called

[1]While easy to conflate true and false with predicting the veracity of a claim, our objective is not to build a truth-teller machine but rather assess the relation between the claim and evidence

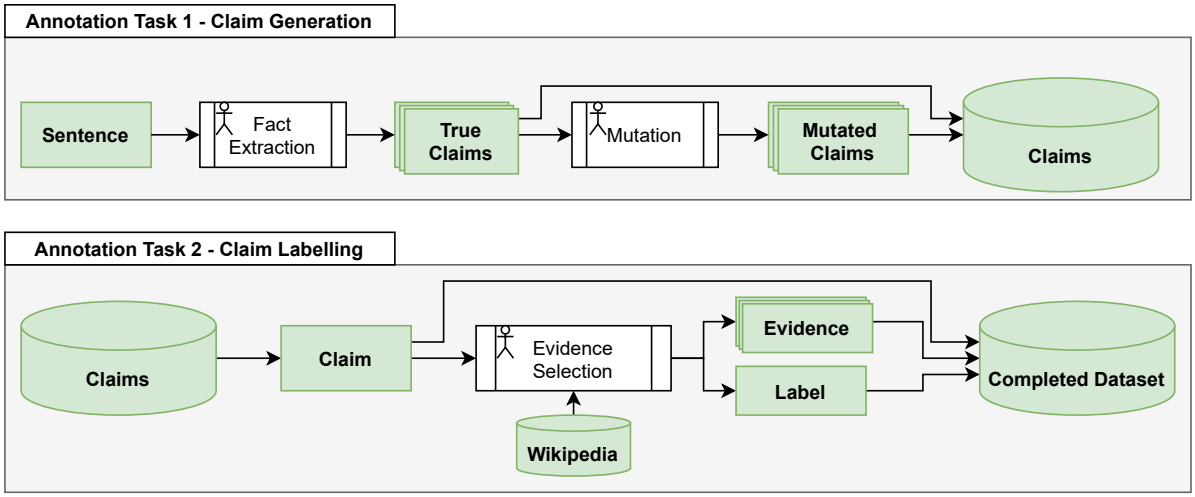


Figure 3.2: Overview of the FEVER dataset annotation procedure. The procedure is comprised of two workflows completed by different groups of annotators.

mutations, yielding a set of claims which may be contradictory to the original claim. In the separate evidence labelling annotation task, annotators were asked to label whether a claim is supported or refuted by selecting the evidence from Wikipedia for it or labelling if there is not enough information to make a decision.

The claims were generated and labelled with evidence collected from the June 2017 Wikipedia dump, sentence-split using Stanford CoreNLP (Manning et al., 2014). Only the introductory sections[2] were considered. This was because they were information-dense: meaning that annotators could generate meaningful claims and reliably find evidence to support or refute these claims in a time-efficient manner.

The annotation interface was designed with the support of two *beta tester* annotators locally before scaling the task up to a larger pool of annotators. The testers were presented with initial guidelines and asked to perform the task without being able to ask for support or clarifications to find weaknesses in the documentation and usability issues with the web-based annotation interface. Both beta testers were experienced annotators who were native British English speakers and local to Cambridge, UK. Several iterations of the guidelines and software were made, considering the ease of use, average annotation

[2]The paragraphs that appear before the table of contents or first subheading of the Wikipedia article

time, and the consistency of the evidence retrieved from the second annotation workflow task. These test annotations were not incorporated into the final dataset.

The main body of annotations was collected with the support of an external annotation service that managed the recruitment and training of the annotators. The annotation managers, trainers, and initial senior annotators were first advised of the task requirements and guidelines over the course of several onboarding meetings. This initial core team were then used to train the pool of workers, ramping up to the full-size team over a period of approximately four weeks for each annotation task. Minor adjustments were made to the annotation system and guidelines following questions from workers or patterns observed in the collected data. Annotators were able to seek clarification between themselves (including the trainers) using a private chatroom. Queries that could not be resolved locally were escalated to a weekly review meeting where issues and performance were discussed. The annotation team consisted of 50 members, 25 were involved in the first task, and 40 were involved in the second task. All annotators were native US English speakers, based in Cambridge, MA, and were trained by myself as part of the on-boarding with the external annotation service or subsequently by the experienced annotators.

3.2.1 Annotation task 1 - claim generation

The objective of this task is to generate claims from information extracted from Wikipedia. Sentences sampled from the introductory sections of approximately 50,000 popular pages[3] were presented to annotators who were first tasked with generating a set of *claims*. Claims are factoid statements pertaining to a single piece of information, focusing on the entity that the sampled Wikipedia page was about. If only the source sentences were used only to generate claims by extracting facts, the resulting dataset would consist of trivially verifiable claims. The claims would, in essence, be simplifications and paraphrases. To increase the diversity and complexity of claims, annotators were asked to incorporate additional knowledge and make mutations, outlined below:

[3]Taken from the 5,000 most accessed pages in June 2017 and the pages hyperlinked from them.

Additional knowledge If annotators were allowed to freely incorporate arbitrary world knowledge, claims would be hard to verify on Wikipedia alone. To balance diversity and verifiability against Wikipedia, annotators were presented with a *dictionary*: a list of terms compiled from the hyperlinked pages from the sampled sentence allowing the annotators to expand upon the claims by incorporating world knowledge in a controlled manner. For each entry in the dictionary, the first sentence from the corresponding Wikipedia page was shown as this was typically sufficient to define the term and provided a reasonable amount of background information.

Mutation The annotators were also asked to generate *mutations* of the claims: potentially meaning-altering rewrites which may or may not change whether the claims are supported by Wikipedia; or even if they can be verified against it. To balance the diversity in the generated claims and minimise the time taken to perform the task, annotators were provided six prompts, inspired by the operators used in Natural Logic Inference (Angeli and Manning, 2014), to generate the mutations. These were: paraphrasing, negation, substitution of an entity or relation claim with either a similar or dissimilar one, and making the claim more general or specific.

3.2.2 Annotation task 2 - claim labelling

The annotators were tasked with labelling evidence for each individual claim generated from the first annotation task. There were three choices for the label: SUPPORTED, REFUTED or NOTENOUGHINFO. For the first two labels, the annotators were asked to find evidence using a standard of proof similar to what a 'reasonable average person' would use to justify their belief that the claim is true or false. Specifically, the annotators were asked to assign the labels as supported if, given the evidence sentences they select, they would have a strong reason to believe the claim was true, and vice versa for the refuted label. The NOTENOUGHINFO label was assigned when no matching evidence was found. There was no pre-determined time limit for this task, but the annotators were advised not to spend more than 2-3 minutes per claim.

The default source of evidence was the introductory section of the Wikipedia page. This section was sentence split, and each sentence could be expanded with the option to create an evidence set (the minimal set of evidence sentences, in the event that one sentence alone was insufficient to verify the claim) that either supports or refutes the claim. Additional sentences could be added to the evidence set from a bank of sentences pre-populated from hyperlinked pages, or to allow exploration beyond linked pages, additional Wikipedia pages of the annotator's choosing could be added by providing its URL. Although this decision was not explicitly recorded, annotators were allowed to use the page title to resolve co-reference.

3.2.3 Data validation

All claims generated from the first annotation task were checked by the annotators performing the subsequent evidence labelling task: annotators were asked to flag any claims which were not intelligible or contained typographical errors. For the second annotation task, two different data validation processes were maintained throughout the data collection process to measure the label agreement and the appropriateness of evidence returned by the annotators on a sample of the data. A final meta-annotation, considering the claims and the evidence, was performed over the compiled dataset.

Task 1: claim validity

To validate claims generated from the first annotation task, the annotator of the second annotation task could skip or flag claims. This provided a check to ensure that the claims undergoing verification complied with the task guidelines (such as being free of typographical errors) and were not too vague. The annotators skipped 0.09% of claims without giving a reason, with an additional 1.8% of claims were flagged as containing typographical errors and 5.5% of the generated claims being flagged as too vague or being ambiguous. All skipped and flagged claims were excluded from the dataset.

Mutation Type	Proportion (%)	Skip Rate (%)		
		No Reason	Ambiguous	Typo
Original	18.1	0.06	1.6	1.1
Rephrase	13.7	0.09	3.7	2.9
Negate	14.0	0.18	5.9	1.6
General	13.5	0.11	6.5	1.6
Specific	13.4	0.10	4.9	2.4
Substitute Similar	13.6	0.07	4.9	1.6
Substitute Dissimilar	13.6	0.08	12.4	1.6
Average	-	**0.09**	**5.5**	**1.8**

Table 3.1: Proportion of claims skipped for claims extracted (original) or mutated by the annotators. Meaning altering changes introduced ambiguity, and for the dissimilar entity substitution, this resulted in nonsensical claims that were excluded from the dataset.

Examples of claims skipped for being vague include: "Chris Evans worked.", "The Godfather Part II is a work.", and "Sons of Anarchy premiered." which were written in the 'generalise' mutation text box from the first annotation task. While these instances are grammatical, their meaning depends on the evidence that is found (for example, movies can premiere) and may also be trivial to verify (for example, 'worked' could be verified by any sentence indicating a job role for Chris Evans). Many more of the claims written in the 'dissimilar entity substitution' were labelled as nonsensical: for example, replacing the noun *truck* with *cat* in a claim about 'articulated trucks'. The annotators flagged the claims generated for this mutation type for being ambiguous twice as frequently as most of the other mutation types, shown in Table 3.1.

Task 2: label agreement

To measure inter-annotator agreement for the labels assigned to claims (SUPPORTED / REFUTED/ NOTENOUGHINFO), 4% ($n = 7506$) of claims not skipped or flagged were sampled to undergo repeated annotation by five different annotators. The Fleiss κ score (Fleiss, 1971) was used to measure the agreement at $\kappa = 0.6841$. This is in line with similar annotation tasks: Bowman et al. (2015) reported a κ of 0.7 for a more straightforward task where annotators were labelling the relation between premise and

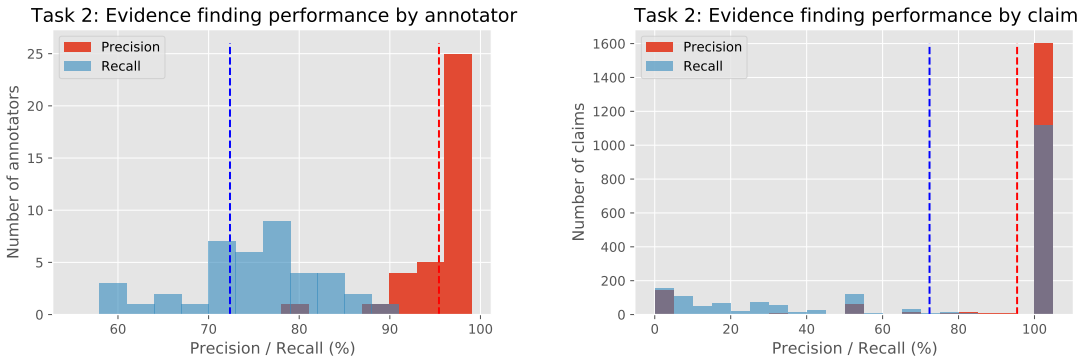


Figure 3.3: Precision and recall of evidence selected by individual annotators against super-annotators for the evidence finding component of task 2.

hypothesis sentences in the construction of the SNLI dataset – even without complexity introduced by the additional task of finding evidence.

Task 2: evidence finding

A further 1% of claims (n = 2079) were sampled and annotated by a group of five *super-annotators*. Super-annotators were expert annotators and trainers for the task and were instructed to find all evidence with no suggested time restrictions. The purpose of this exercise was to provide as much coverage of evidence as possible. This evidence would be used to measure the precision and recall of individual annotators against the union of all annotations from the super-annotators.

The precision and recall of the evidence annotation was 95.42% and 72.36%, respectively. This indicates that, for approximately 5% of evidence, annotators selected additional sentences that the super-annotators did not, and the annotators did not select 28% of the evidence selected by super-annotators. The distribution over recall scores, illustrated in Figure 3.3, is bimodal when broken down by claim. The majority of claims have perfect recall, and a small number of instances with low recall were observed where annotators missed or did not select all of the evidence identified by the super-annotators. The majority of the low-recall cases (but non-zero recall) are for claims such as "Akshay Kumar is an actor." where the super-annotator added 34 sentences as evidence, most of

the selected sentences being entries from filmography listings (e.g. "In 2000, he starred in the Priyadarshan-directed comedy Hera Pheri"). In contrast, the regular annotators added a minimal set. Most annotators had high precision for the evidence finding task, with only two attaining under 90%. Considering the distribution of precision by claims, annotators identified new evidence for approximately 200 claims. The majority of these cases were claims that were labelled as NotEnoughInfo by super-annotators, whereas the regular annotators found supporting information, resulting in zero precision.

Final validation

As a final quality control step, 227 instances were sampled from the compiled dataset and annotated by myself and project advisers considering both the accuracy of the labels and the evidence provided. Each instance was annotated by two people, verifying the claim and annotation against the full guidelines listed in Appendix A.1. Disagreements were resolved through the use of a chatroom. The meta-annotation found that 91.2% of the examples were annotated correctly. Of the errors, 3% of instances were found to contain mistakes in claim generation (such as typos) that had not been flagged during labelling, 2% had an incorrect label assigned (e.g. claim was labelled as Supported, but the evidence suggests that it is Refuted), and the remaining claims had evidence that would not be considered sufficient by the annotation guidelines.

Evidence sufficiency was one of the overbearing issues in designing the second annotation task: annotators' backgrounds contribute different knowledge leading to different labels being applied during annotation – more formally explored in subsequent work by Pavlick and Kwiatkowski (2019) and Nie et al. (2020b). To ensure annotation consistency, the guidelines asked the annotators to err on the side of caution of what was common knowledge versus specialist knowledge and make use of evidence from the linked Wikipedia page to incorporate facts that an average person may not know.

Split	SUPPORTED	REFUTED	NOTENOUGHINFO
Training	80,035	29,775	35,639
Development 1	3,333	3,333	3,333
Development 2	3,333	3,333	3,333
Blind test set (shared task)	6,666	6,666	6,666

Table 3.2: Dataset split sizes for SUPPORTED, REFUTED and NOTENOUGHINFO (NEI) classes. The development and test splits are balanced through subsampling.

Evidence Sets Per Instance	Count
0 (NEI)	48,971
1	110,395
2	12,679
3	4,700
4	2,599
5	1,554
6	1,113
7	747
8	609
9	432
10	349

(a) Number of evidence sets

Size of Evidence Set	Count
0 (NEI)	*48,971*
1	175,320
2	30,488
3	3,513
4	822
5	362
6	*184*
7	*137*
8	*62*
9	*42*
10	*34*

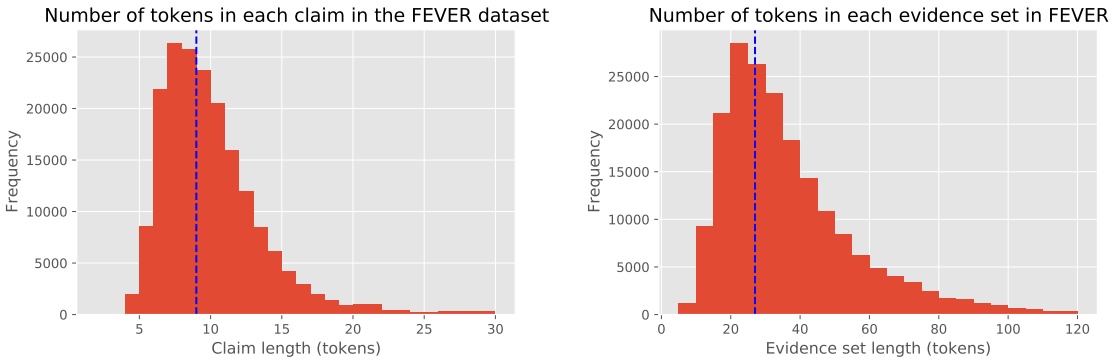
(b) Size of evidence sets

Table 3.3: Number of evidence sets and number of sentences in each evidence set.

3.3 Dataset statistics

The completed dataset was partitioned into training, development and test sets. Each partition was constructed to be pairwise disjoint by the Wikipedia page used to generate the claim (so that the page used to generate claims occurs in exactly one set). For the dataset splits used for evaluation (development and the blind test set), the classes were balanced through down-sampling instances with the majority class labels.

In each instance, evidence is provided as a minimal set of sentences that support or refute the claim. The distribution of the evidence set sizes and number of evidence sets is reported in Table 3.3. Most instances only consist of a single evidence set containing a single evidence sentence – meaning that it is sufficient for systems to retrieve one contiguous span as evidence for most claims. A further 16.8% of evidence sets require



(a) Claim (b) Evidence sets

Figure 3.4: Number of tokens in each instance's claim and evidence set in FEVER

the combination of two or more sentences in order to be sufficient to verify the claim, and 12.2% of instances require the combination of evidence sets from different pages, presenting an extension challenge for top-performing systems. The majority of evidence sets contain fewer than five sentences. For the shared task test set, evidence sets of six or more sentences were discarded (thirteen in total). 19.1% of supported and refuted instances contain more than one evidence set, meaning that Wikipedia contains multiple pieces of information that are independently sufficient to verify the claim. Instances with a high number of evidence sets were likely to be collected by the super-annotators.

The mean generated claim length in FEVER is 9 tokens, with a bias towards shorter claims (skewness=4.46). This is comparable to SNLI (Bowman et al., 2015), a related natural language inference dataset where premise sentences were written by annotators. However, in contrast, the length of evidence that models must use when reasoning is much higher than SNLI (mean length in FEVER is 25 tokens, whereas the mean premise sentence length in SNLI is 14 tokens). Histograms showing the number of tokens for claim and evidence are plotted in Figure 3.4.

3.4 Baseline approaches

A reasonable baseline system would consist of a pipeline of an evidence retrieval component (at document and sentence level) and claim veracity classifier. This section introduces approaches for these two sub-tasks, highlights related work, and discusses how features of these approaches influence their performance on FEVER.

3.4.1 Evidence retrieval

Given a claim c, the document retrieval system returns a shortlist of pages from a Wikipedia corpus $W \subset \mathcal{W}$ which contain evidence. This approach can be modelled in two ways: considering either document similarity to the claim or as an entity disambiguation task – predicting the canonical document for an entity mention (Bunescu and Paşca, 2006) using the context it appears in (Le and Titov, 2018).

At the time of the dataset release, similarity-based approaches for retrieval considered token-level features common between an input query (such as a claim) and documents within a corpus. For example, a related Wikipedia-based question answering system, DrQA (Chen et al., 2017a), retrieves Wikipedia pages for open-domain question answering using TF-IDF. This retrieval method scores a document's relevance using the cosine similarity of vector representations of a query and the document. Elements of these vectors are, for each token t, the product of a term's frequency $f_{t,d}$ for a document d and inverse document frequency (Spärck Jones, 1972) which increases the precision of retrieval by reducing the scores of non-relevant documents. Document frequency is the number of documents containing the token, N_t, in a collection of size N. The variant of TF-IDF used in this chapter is provided in Equation (3.1) which makes use of log normalisation, reducing the impact of very frequent terms, and one of the variants of IDF proposed by Spärck Jones (1972) which assigns negative weight to terms occurring in more than half of documents.

$$\text{TF-IDF}_d(t) = \log(f_{t,d} + 1) \cdot \log \frac{N - N_t + 0.5}{N_t + 0.5} \tag{3.1}$$

BM25 (Robertson and Walker, 1994), an extension to TF-IDF, also contains a document-level saturation control s_d for the term-frequency component. This term considers the document's length $\#(d)$ compared to the average document length in the corpus $s_d = \frac{\#(d)}{\frac{1}{N}\sum_{i=1}^{N}\#(d_i)}$, which reduces the weighting on term frequency for longer documents:

$$\text{BM25}_d(t) = \log \frac{f_{t,d} * (k_1 + 1)}{f_{t,d} + k_1 * (1 - b + b * s_d)} \cdot \log \frac{N - N_t + 0.5}{N_t + 0.5} \tag{3.2}$$

More recent retrieval methods consider the similarity between dense representations of documents and queries. For example, DPR (Karpukhin et al., 2020) uses a transformer-based neural network to independently encode passages from documents and queries, achieving high recall on a number of Wikipedia-based question answering tasks (Petroni et al., 2021). Two independent encoders, $E_P(\cdot)$ and $E_Q(\cdot)$, return low dimensional real-valued vector representations of the input strings. These are trained with supervision for which passages p relevant to a query q using a triplet loss objective which maximises $sim(p, q) = E_P(p)^{\mathrm{T}} E_Q(q)$ for valid passages and minimises $sim(p, q)$ for negative instances which are sampled from BM25 search results.

For the entity disambiguation formulation for retrieval, systems predict the Wikipedia article for the entities or concepts in the claim. The state-of-the-art for this task is to use an encoder-decoder architecture, such as GENRE (De Cao et al., 2020), to predict the sequence of tokens of a Wikipedia article's title given a claim. Two simpler baselines that use a named entity recogniser and a noun-phrase chunker will also be compared. These will return pages from Wikipedia by considering matches against an inverted index. To allow for variations of entity names (such as John F Kennedy versus JFK), these baselines will resolve canonical names by following Wikipedia redirects.

Sentence selection

The sentence selection sub-task requires the model to filter appropriate evidence from the retrieved documents. The selection of sentences, $\hat{E} \subset \bigcup_{w \in W} \{e \in w\}$, may be modelled through techniques used from the document retrieval component discussed in this section (for example, considering semantic similarity using TF-IDF) or through the model architectures used for predicting the claim's veracity, later described in Section 3.4.2, labelling whether a sentence is evidence or not.

3.4.2 Claim veracity prediction

Claim veracity prediction is modelled as a supervised text-pair classification task of a claim, c, given evidence, E. This section will survey key advances in modelling approaches for text-pair classification. While these share the common approach of using a feed-forward neural network (FFNN) to perform multi-class classification of an encoded representation of both string inputs, the method for generating these encodings varies, considering the interaction between the two sentences. For this family of models, it is assumed that the evidence (or premise) is provided as a single passage rather than a set of sentences. To adapt these to FEVER, where multiple text spans are evidence, these evidence passages will be concatenated into a single string, $e = \text{concat}(E)$.

One of the top-scoring systems (Riedel et al., 2017) in the Fake News Challenge (Pomerleau and Rao, 2017) used a bag of words encoding with one additional scalar feature for sentence similarity. The input to the FFNN was the concatenation (denoted as $[a; b] \in \mathbb{R}^{|a|+|b|}$) of the term frequency vectors from both of the text inputs, denoted Φ, and the cosine similarity of their TF-IDF vectors, denoted Ψ, described in Equation (3.3). This rudimentary approach to encoding the texts does not capture certain phenomena, such as word order. Furthermore, the input dimension to the feed-forward network was fixed, meaning that only 5000 of the most popular tokens could be represented as input.

$$\text{FFNN}([\Phi(c); \Phi(e); \frac{\Psi(c) \cdot \Psi(e)}{\|\Psi(c)\|\|\Psi(e)\|}]) \tag{3.3}$$

For the related Natural Language Inference task, many high performing architectures operate using a feed-forward classifier over encoded representations generated from deep neural networks. Starting with the siamese encoder used for SNLI (Bowman et al., 2015), the sentences are *independently* encoded with a Long-Short Term Memory (LSTM) model (Hochreiter and Schmidhuber, 1997), a gated recurrent neural network that captures long-range token interactions, yielding a vector for every token in a string of length N (specifically, the LSTM *hidden state*). The inputs to the FFNN are the hidden states of the final tokens encoded by the LSTM, denoted LSTM_N.

$$\text{FFNN}([\text{LSTM}_N(c); \text{LSTM}_N(e)]) \tag{3.4}$$

Rocktäschel et al. (2016) demonstrated additional gains in accuracy on SNLI using conditional encoding and attention. With conditional encoding, the hypothesis is encoded using a model initialised with the encoded premise: a crucial modelling assumption when there is a dependency between the two sentences (Augenstein et al., 2016a). In practice, the final cell state of the LSTM encoded representation of the premise is used to initialise the LSTM encoder for the claim: $\text{LSTM}_{CE}(\text{LSTM}_N^{cell}(e), c)$. Attention additionally reduces the burden on the encoder to incorporate all semantic information within a single vector. Rather than using the final hidden state of an encoder $\text{LSTM}_N(x) = \mathbf{h}_N$, attention considers a weighted sum of each token's vector $\text{LSTM}(x) = [\mathbf{h_1}, \ldots, \mathbf{h_N}]$, yielding an attended representation $\mathbf{r} = \sum_{i=1}^{N} \mathbf{h_i}\alpha_i$. The scalar attention weights α_i are a distribution over tokens that is computed considering the interaction between the token's vector and another object, such as the final hidden state of the representation of the claim. The distribution is generated by computing a softmax function over the outputs of an alignment model (Bahdanau et al., 2015): a feed-forward neural network ($a : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \rightarrow \mathbb{R}$) that is jointly trained with the classifier and that outputs a scalar score describing the importance of the interaction.

For the published FEVER baselines (Thorne et al., 2018a), a Decomposable Attention network (Parikh et al., 2016, DA) was used as it was the highest accuracy model for the

SNLI task at the time that was not an ensemble and did not require syntactic parsing of the inputs. The DA architecture is a feed-forward classifier over attended representations of the two input sentences and is agnostic to the choice of encoder. The model has shown high accuracy when run over pre-trained word embeddings, generated through a method such as GloVe (Pennington et al., 2014), without the need for a recurrent encoder such as an LSTM, making parallelisation trivial. Furthermore, the DA architecture factorises the alignment model $a_{i,j} = F'(e_i, c_j) = F(e_i)^T F(c_j)$, reducing the complexity of computing attention by introducing a low-rank assumption over alignment.

This chapter will make additional comparisons to more recent text-pair encoders such as ESIM (Chen et al., 2017b) and BERT-based (Devlin et al., 2019) transformer models (Vaswani et al., 2017) which, when combined with a feed-forward classifier, have attained higher accuracies than the previously discussed models. ESIM is an LSTM model with factorised alignment for attention, similar to DA (Parikh et al., 2016), where rather than only inputting attended representations of the claim and evidence, the encoded representations consider the concatenation of the representation prior to attention, the attended representation, as well as the element-wise difference and element-wise products of the two. Furthermore, rather than using weighted summation, ESIM also considers average and max-pooling when generating sentence-level representations, resulting in higher accuracy. Transformer architectures (Vaswani et al., 2017) are feed-forward neural models with multiple attention heads at each layer, with tokens encoded with position embeddings. The lack of recurrence allows models with large numbers of parameters to be trained efficiently, enabling pre-training on a language model task prior to fine-tuning on a specific task (Radford et al., 2018, Devlin et al., 2019). This pre-training captures lexico-semantic information – improving accuracy when fine-tuned with task-specific training – and also results in a smoother error surface and more stable optima that are more robust to over-fitting (Hao et al., 2019).

3.5 Experiments

3.5.1 Scoring

Predicting whether a claim is SUPPORTED, REFUTED, or NOTENOUGHINFO is a 3-way classification task that is evaluated using accuracy, outlined in Equation (3.5), which compares the actual label y and the predicted label $\hat{y}$. Given that the development and test datasets have balanced class distributions, a random baseline should attain 33% accuracy if one ignores the requirement for systems to return correct evidence.

$$\text{Accuracy}(\mathcal{Y}) \triangleq \frac{1}{|\mathcal{Y}|} \sum_{(y,\hat{y})\in\mathcal{Y}} \mathbb{I}[y = \hat{y}] \tag{3.5}$$

For claims that are labelled as SUPPORTED or REFUTED, systems must return evidence as well as the label. In the FEVER Score, presented in Equation (3.6), an accuracy point is only awarded if both the correct label and, with the exception of the NOTENOUGHINFO class (abbreviated NEI), correct evidence is returned, as outlined in Equation (3.7).

$$\text{FEVER}(\mathcal{Y}) \triangleq \frac{1}{|\mathcal{Y}|} \sum_{(y,\hat{y},\mathbf{E},\hat{E})\in\mathcal{Y}} \mathbb{I}[\text{Instance\_Correct}(y, \hat{y}, \mathbf{E}, \hat{E})] \tag{3.6}$$

$$\text{Instance\_Correct}(y, \hat{y}, \mathbf{E}, \hat{E}) \triangleq (y = \hat{y}) \wedge \Big(y = \text{NEI} \vee \text{Evidence\_Correct}(\mathbf{E}, \hat{E})\Big) \tag{3.7}$$

As there may be multiple correct combinations of evidence sentences that can be used, the evidence returned by the system $\hat{E}$ is considered sufficient if it is a superset of at least one of the evidence sets selected by the annotators $\mathbf{E} = [E_1, \ldots, E_k]$.

$$\text{Evidence\_Correct}(\mathbf{E}, \hat{E}) \triangleq (\exists E \in \mathbf{E})(\hat{E} \supseteq E) \tag{3.8}$$

As it is not feasible to ensure completeness of the evidence annotations, there is no penalty for systems returning false-positive evidence. However, to ensure that a high-recall system does not abuse this limitation for scoring, the number of evidence sentences considered for scoring is capped at 5, as this is sufficient for most claims (for the shared task test set, evidence sets of 6 or more sentences were discarded).

3.5.2 Implementation

Retrieval

The implementation of the TF-IDF retrieval system uses code released for DrQA (Chen et al., 2017a). This version considers both token unigrams and bigrams for retrieval. And, to handle out of vocabulary tokens, performs a 24-bit hash over these. This implementation was extended to perform BM25 retrieval. Both the GENRE retriever and the DPR retriever use the code and model weights released by the respective authors (De Cao et al., 2020, Karpukhin et al., 2020). For the baseline entity linking retrievers, the redirects from the Wikipedia snapshot used to construct the dataset were extracted and used to create an index that maps from phrases to page titles. Named entities and noun phrases from the claim were tagged with SpaCy,[4] and the retrieved pages were ordered by the matching indexed phrases with the highest character-level similarity. Where combinations of two retrieval methods are used, the ranked outputs from both methods are interleaved into a list before deduplication.

Classification

The experiments will compare the Decomposable Attention (Parikh et al., 2016), ESIM (Chen et al., 2017b) and two variants of the transformer models: BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019b). All models use pre-built implementations from the AllenNLP framework (Gardner et al., 2018). The BERT and RoBERTa are initialised from the pre-trained 'base' versions. Models are trained using default hyperparameter

[4] `https://spacy.io/`

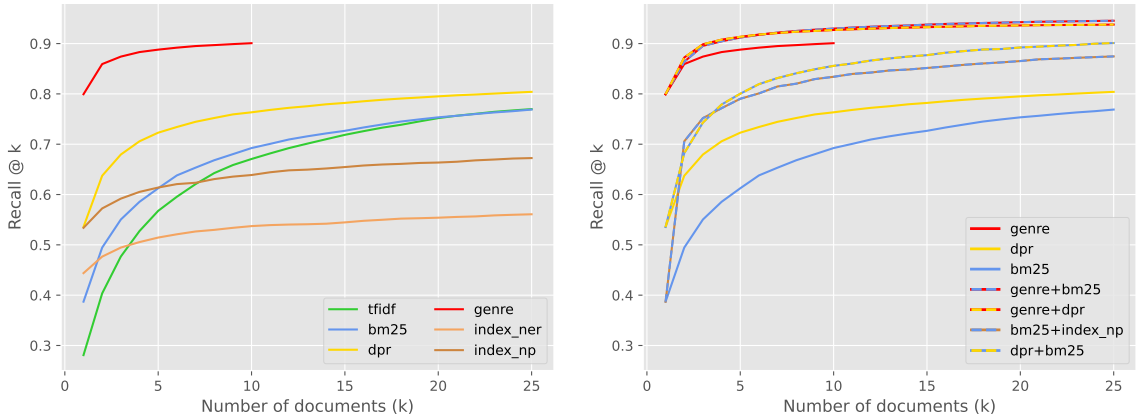
choices, documented in Appendix A.3, selecting the highest performing model (either considering label accuracy for classification or recall@k for sentence selection) on the validation dataset. All results reported are on the validation dataset, with the final results for the best performing system being reported on the shared task test set.

3.5.3 Document retrieval

The document retrieval component of the baseline system is first evaluated in isolation, considering the recall@k. The recall@k is further used to indicate an upper bound FEVER Score that assumes a perfect classifier and sentence selection module. The document recall@k scores for all techniques are plotted in Figure 3.5, varying the number of evidence documents, k.

Modelling retrieval as named entity linking with GENRE (De Cao et al., 2020) yielded the highest recall: considering the top $k = 5$ pages resulted in a recall@k of 89% (this indicates an upper-bound FEVER Score of 93%). Rather than generating a vector representation of the claim, and retrieving the document with the most similar vector, like the TF-IDF, BM25 and DPR, GENRE predicts a sequence of tokens that exactly refer to one document resulting in a very high recall, even when only considering the first generated document (recall@1 was 80%). Performing entity linking with simple baselines, considering noun-phrases (index_np) or named-entities (index_ner) was less effective than using GENRE. However, for low values of k, the recall for these two index-based methods was also higher than the TF-IDF and BM25 retrieval.

The two token-based document similarity metrics, TF-IDF and BM25, exhibit relatively low recall for low values of k. For example, the document-level recall@k for TF-IDF, considering the $k = 5$ top documents is 57% (this would result in an upper-bound FEVER Score of 71%). For larger values of k, both TF-IDF and BM25 exhibit diminishing returns in recall. While returning more documents increases the chance that correct evidence is found, this comes at the cost of inputting more sentences to the downstream sentence-selection component, which not only has a computational overhead but also has the



(a) Retrieval methods in isolation (b) Combination of multiple retrieval methods

Figure 3.5: Performance of evidence retrieval methods considering the recall@k by considering the top-k documents returned by the retrieval system.

potential of returning more false-positive evidence to the veracity classifier. A limitation of these token-based retrieval methods is that they depend on exact matches between tokens in the claim and tokens in the documents and would not be able to return similar documents if an alternative phrasing of an entity is used (for example, President Kennedy versus JFK). The Dense Passage Retrieval (DPR) approach, in contrast, uses transformers to encode a low-dimensional real-valued vector representation of the claim and passages within the documents, which are more robust to different ways of phrasing the same information, resulting in higher recall (at $k = 5$, the recall of DPR was 72% (indicating an upper bound FEVER Score of 82%)).

Combining retrieval methods yields improvements in recall, plotted in Figure 3.5b. Marginal improvements were made to GENRE by combining the predicted documents with retrieval results from BM25 and DPR. The recall@5 increased by approximately one percentage point, covering cases where GENRE does not correctly predict the page title. More substantial improvements were made to the token-based document similarity methods, TF-IDF and BM25. Combining BM25 with the index-based retrieval increased recall@5 by approximately twenty percentage points because the index is able to resolve spelling variations for some entities by following Wikipedia redirects.

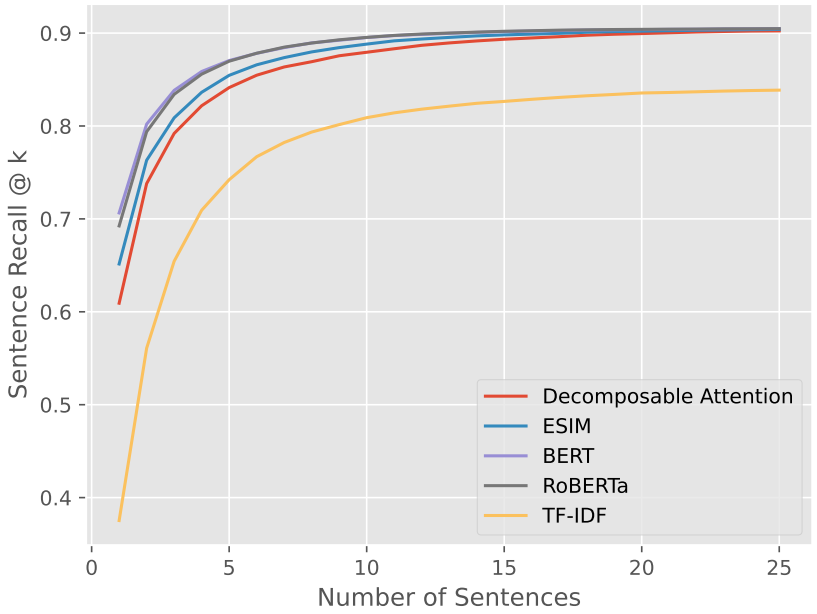
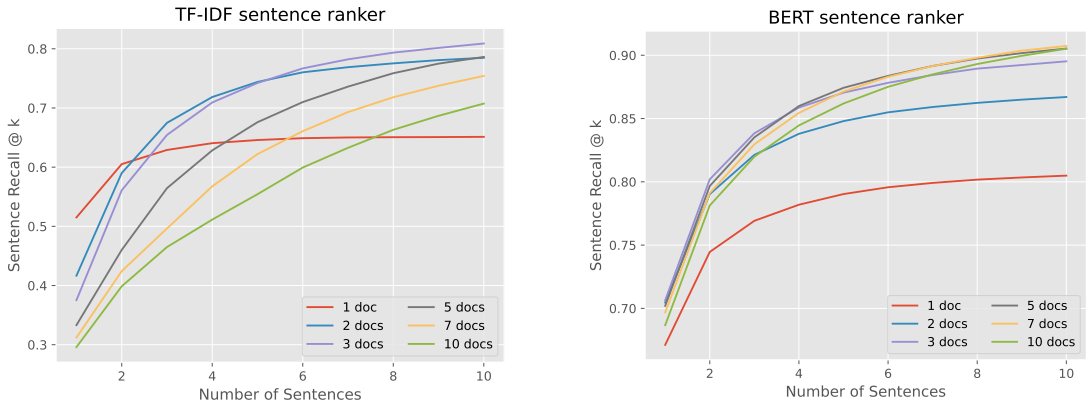


Figure 3.6: Sentence-level recall@k considering the top three documents returned from GENRE+BM25, comparing binary classifiers and TF-IDF to select sentences.

3.5.4 Sentence selection

Sentence selection is modelled in two distinct ways: either as retrieval of semantically similar sentences to the claim (mirroring document retrieval) or as a classification where a model predicts whether a given sentence is evidence for a claim (mirroring the approaches used in the claim veracity classification). Both will be compared in combination with the highest-recall document retrieval system: combining predicted documents from GENRE with retrieved documents from BM25 with results plotted in Figure 3.6.

When modelling retrieval as classification, the most relevant sentences are returned and ordered by $P(is\_evidence|claim, sentence)$. The difference in recall between different model architectures (for example, Decomposable Attention and RoBERTa) is substantial when considering smaller numbers of evidence sentences (such as the top one or two predicted sentences). However, in the limit, considering larger numbers of evidence sentences, the recall converges to the upstream document retriever's recall of 90%. Considering the TF-IDF-based sentence retrieval, the recall, in contrast, converges to a



(a) TF-IDF sentence retriever

(b) BERT sentence classifier

Figure 3.7: Increasing the number of documents input to the sentence retrieval module yields diminishing returns for the BERT-based retriever and harms the TF-IDF retriever.

value that is five percentage points lower than the classifier-based approaches as sentences with no tokens in common between the claim are not returned at all by TF-IDF.

The number of retrieved documents considered when performing sentence selection is an influential design choice when optimising sentence recall@k. While using sentences from more documents increases the upper bound for the sentence-level recall@k, the higher number of non-evidence sentences increases the potential for the sentence selection module to incorrectly rank non-evidence sentences higher than evidence sentences. This is demonstrated for TF-IDF in Figure 3.7 where the recall@5 is highest when considering top two or three documents, and considering more documents decreases the gradient of the recall curve. For the BERT-based sentence classifier, marginal gains in recall@5 can be attained by considering more documents. However, there are diminishing returns, and inference time increases linearly with respect to the number of documents.

3.5.5 Recognising textual entailment

The RTE component is a text-pair classifier that is trained to predict the veracity of claims given evidence. Evidence is selected as multiple, possibly discontiguous, sets of sentences from the evidence documents and are concatenated as input to the model. Instances for claims labelled as NotEnoughInfo contain no annotations for evidence

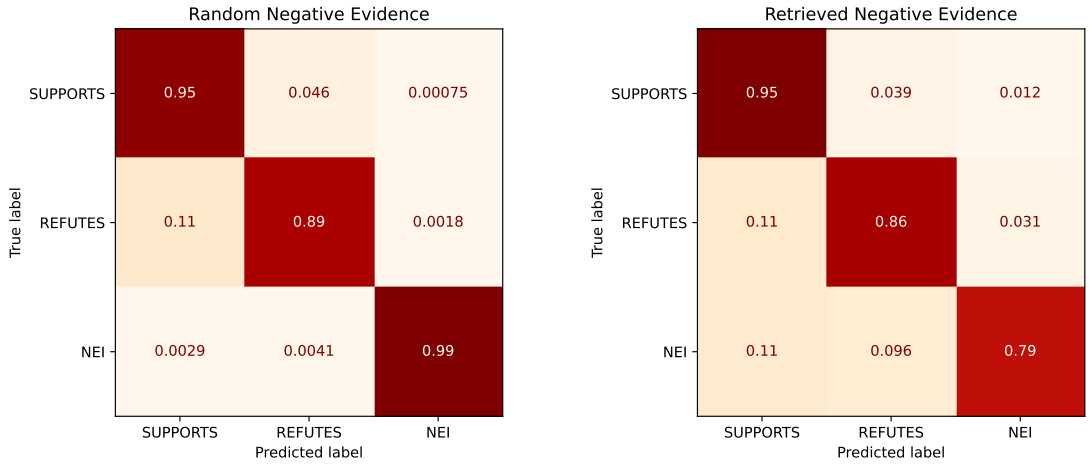
Model	Label Accuracy (%)			
	FEVER:RandNEI	FEVER:RetrNEI	SNLI	MultiNLI
DA	75.10	63.76	45.18	63.24
ESIM	90.28	81.31	55.71	65.31
BERT	94.07	87.51	30.52	40.96
RoBERTa	94.86	88.54	32.35	44.23

Table 3.4: Oracle classification accuracy on claims in the dev set using gold evidence

and thus, negative training instances must be sampled for this class. This section first evaluates classification accuracy on the development set in an oracle evaluation that assumes a perfect evidence retrieval system and then evaluates the impact of retrieved evidence and the model's robustness to noise (such as false-positive evidence) introduced by the evidence retrieval components in Section 3.5.6.

Sampling evidence for the NotEnoughInfo class uniformly at random from Wikipedia (denoted RandNEI in Table 3.4) yielded sentences that were unrelated to the claim, resulting in high accuracy for this NEI class. While the accuracy of the models trained with this sampling approach is higher in the oracle evaluation setting, this may not necessarily yield a better system in the pipeline setting where retrieved evidence is used. In contrast, sampling negative evidence from the retrieved documents (denoted RetrNEI) simulates finding related information that may not be sufficient to support or refute a claim, resulting in training instances more representative of working with retrieved evidence, resulting in lower accuracy. This is highlighted in the confusion matrices plotted in Figure 3.8, showing the more challenging instances for the NEI class. Both RandNEI and RetrNEI sampling strategies will be evaluated in the full pipeline setting in Section 3.5.6, which accuracy using retrieved evidence.

To compare the challenges in FEVER to other natural language inference tasks, pre-trained models for SNLI (Bowman et al., 2015) and MultiNLI (Williams et al., 2018) are used to make predictions on the FEVER instances, mapping the Supported label to Entailment, the Refuted label to Contradiction and the NEI label to Neutral,



(a) Negative evidence is randomly sampled from all of Wikipedia.

(b) Negative evidence is sampled from retrieved pages.

Figure 3.8: Confusion matrices highlighting how randomly sampled negative evidence for NEI instances results in near-perfect accuracy for this class.

Model	Accuracy (%)			FEVER Score (%)		
	RandNEI	**RetrNEI**	**RetrAll**	**RandNEI**	**RetrNEI**	**RetrAll**
DA	48.88	49.15	50.93	43.02	43.24	45.44
ESIM	57.02	59.26	67.57	50.40	52.72	62.00
BERT	60.85	64.95	75.59	54.32	58.73	70.11
RoBERTa	61.48	65.61	77.22	54.95	59.33	71.75

Table 3.5: Full pipeline results, predicting claim veracity with retrieved evidence

without any other changes or task-specific fine-tuning. While the transformer-based architectures yielded the highest label accuracy when trained on FEVER, their accuracy was low when considering pre-trained natural language inference models. This indicates that the entailment rules captured when training the transformer models on SNLI or MutliNLI don't generalise well to the challenges present in FEVER. In contrast, the older model architectures, DA and ESIM, attained higher accuracy on FEVER when pre-trained on the NLI datasets, indicating the lack of ability to fit the data well that implicitly regularises the model.

3.5.6 Full pipeline

The model evaluated in Section 3.5.5 assumed perfect information retrieval for the evidence for the supported and refuted classes. In practice, however, the models evaluated for FEVER must make predictions using retrieved evidence from Wikipedia. In all cases, for models reported in Table 3.5, using retrieved evidence rather than the oracle evidence reduced the label accuracy of the classifier in comparison to the results reported in Table 3.4. Using randomly sampled negative evidence for the NotEnoughInfo class resulted in the lowest label accuracy when combining the classifier with evidence retrieval, regardless of model choice. Even though this model attained the highest accuracy in the oracle evaluation, errors from the sentence retrieval propagated to cause errors when predicting the label. Another model was trained using the only predicted evidence (denoted RetrAll in Table 3.5) and did not use the labelled evidence from the dataset for training the NLI classifier. This was more resilient to the noise introduced by the sentence retrieval module, and despite using the same evidence as the models trained with randomly sampled and retrieved evidence for the Nei class, yielded higher classification accuracies and FEVER Scores, regardless of model architecture, as the model is exposed to predicted evidence for all classes during training.

The FEVER Score, which conditionally awards points for accuracy when sufficient supporting information is returned, is lower than label accuracy for all models and configurations, indicating that while a model was able to predict an appropriate label, this was not always as a result of considering the interaction between the evidence and the claim. It is evident that considering the retrieved evidence when training the model improved the models' ability to capture these interactions. However, there are still many opportunities to build models that can better use the retrieved evidence.

Shared task test set results

The best-performing development set FEVER Score of 71.75% was attained using GENRE and BM25 for document recall, a RoBERTa classifier to predict evidence sentences from

top-3 evidence pages, and a second RoBERTa trained using the predicted evidence sentences to predict claim veracity given the retrieved evidence. Evaluating this pipeline of components on the shared task development set resulted in a FEVER Score of 69.13%. Both label accuracy (74.80%) and evidence recall@5 (87.34%) were approximately 2 percentage points lower on the test set than the development set, contributing to this lower FEVER Score.

3.5.7 Learning curves

To evaluate whether the size of the dataset is suitable for training the RTE component of the pipeline, the learning curves for four model architectures are plotted in Figure 3.9. These models were trained and evaluated with the oracle evidence for supported and refuted classes and negative evidence for the NEI class sampled from retrieved pages.

All learning curves exhibit typical diminishing returns with accuracy increasing with respect to the number of training instances, indicating that the dataset is large enough to demonstrate the differences of models with different learning capabilities. The transformer-based models attained higher accuracies with fewer training data whereas, in contrast, the ESIM-based model did not learn anything useful with fewer than 800 instances. Regardless of the number of training instances, the accuracy of DA is unstable.

3.6 Discussion

This chapter introduced a new dataset and task formulation for the verification of textual claims against evidence. The key difference between the FEVER task formulation and previous fact-checking datasets is that systems are provided only with a claim and must identify appropriate evidence from a corpus of articles at test time. The task can be decomposed into information retrieval task and a text-pair classification similar to NLI, considering whether the claim is supported or refuted by the evidence.

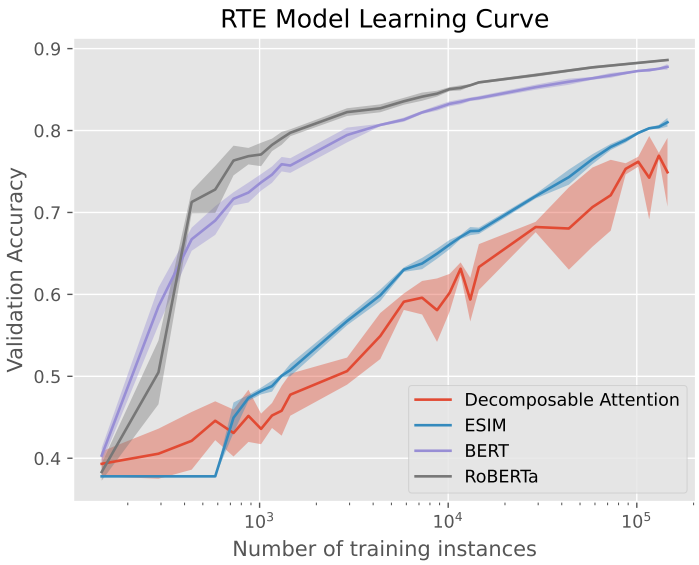


Figure 3.9: Learning curves for the RTE models, considering decomposable attention, the LSTM-based ESIM model and the transformer-based BERT and RoBERTa models

The claim construction process in FEVER is an extension of the methods used for constructing datasets for recognising textual entailment and performing natural language inference, such as SNLI (Bowman et al., 2015). SNLI was constructed by annotators making three changes to captions from the Flikr30K (Young et al., 2014) and Visual Genome (Krishna et al., 2016) corpora. Captions from each photograph from the corpus were provided to annotators as a premise sentence, and the annotators were tasked with generating derivative hypothesis sentences: one which is ENTAILED by the premise, one which is CONTRADICTED by the premise, and one which is NEUTRAL with respect to the hypothesis sentence. The definitions provided to the annotators in SNLI when generating the hypothesis sentences are *definitely true*, *definitely false*, and *might be true*, respectively, given the premise. In contrast, the FEVER annotators were prompted with six mutation types when generating the claims that would help a worker without a background in NLP create a more diverse range of meaning-altering modifications. The label for the claim-evidence pairs in FEVER was applied dependent on the evidence that was found by another annotator in a subsequent task.

FEVER used labels which were a variation on the set of labels typically associated with natural language inference tasks: SUPPORTED, REFUTED and NOTENOUGHINFO. These labels document a weaker semantic relation between the claim and evidence rather than the stronger definition of *definite* entailment, which captured certainty in NLI relations. In NLI, the types of reasoning are simple by comparison to FEVER and are made with limited world knowledge. However, in FEVER, reasoning with a constrained set of world knowledge is an essential task component. Strict reasoning about entailment could lead to epistemic regress, needing to recursively find all necessary supporting knowledge, increasing annotation burden and defeating the purpose of the dataset. The SUPPORTED and REFUTED labels instead provide a relaxation to *plausible entailment* (Resnik, 1993), where one might reasonably assume that the hypothesis in the claim follows from the premise in the evidence. These labels capture the factual relation between the claim and evidence, defined by Spärck Jones (1986, pp. 48-54), where 'one sentence implies another if in saying the one we are prepared to say the other'.

While the format of the evidence selection task is novel, asking annotators to select sentences from Wikipedia, the underlying task is well explored in dataset construction methods used by the recognising textual entailment (RTE) community. The RTE-4 dataset (Giampiccolo et al., 2008) required annotators to perform a three-way labelling whether hypothesis sentences were entailed, contradicted, or neutral with respect to a premise paragraph, and earlier iterations of the RTE datasets (Dagan et al., 2006) consider the task as a binary annotation of entailment. While the premise-hypothesis text pairs in the RTE datasets were constructed prior to annotation, the annotators in FEVER had free reign to choose which sentences act as a premise, containing evidence that supports or refutes what is hypothesised in the claim.

Despite the differing construction methodology, some of the challenges of detecting contradiction (de Marneffe et al., 2008) are present within FEVER and manifested in the annotation process. While the annotators were instructed not to use factives or modal

verbs when generating the claims, other challenges such as entity resolution and applying world knowledge were present identifying evidence and labelling the claim:

Entity resolution A claim such as "Beckham was with United.", might be trivial for an annotator to label as supported given evidence stating "David Beckham made his European League debut playing for Manchester United." However, this annotation implicitly assumes that "United" refers to "Manchester United" and "Beckham" refers to "David Beckham". However, there are many Uniteds in Wikipedia and not just football clubs, e.g. United Airlines, and even within the sports domain, Beckham can refer to Odell Beckham Jr, the American football player or David Beckham, the British soccer player.

Entity resolution was managed for the annotators by construction; the initial set of evidence undergoing annotation was shown from the page used to generate the claim as well as the hyperlinked pages from Wikipedia. This meant that it was relatively easy to resolve ambiguous entities as they were all provided in context, mitigating the challenges identified by de Marneffe et al. (2008)

World knowledge Balancing world knowledge and assumptions needed to reach an inference remained a challenge when constructing the FEVER dataset. To ensure that the necessary knowledge was incorporated into evidence sets rather than relying on assumed knowledge, annotators were asked to err on the side of caution if adequate evidence could not be found, i.e. labelling the claim as NotEnoughInfo.

The need for assumed knowledge was more challenging for refuted claims than supported claims, as lack of evidence does not necessarily imply contradiction. For example, a contentious case was the claim "Shakira is Canadian" which could be labelled as Refuted by the sentence "Shakira is a Colombian singer, songwriter, dancer, and record producer". However, the guidelines advocated that unless more explicit evidence is provided (e.g. "She was denied Canadian citizenship"), the evidence is insufficient since dual citizenship is permitted in these countries, and the annotators' world knowledge should not be factored in.

The experiments in this chapter indicate that it is feasible to model the component sub-tasks for FEVER, such as evidence retrieval and claim veracity prediction with contemporary model architectures. For the models built on token-based retrieval methods (such as BM25 and TF-IDF), retrieving evidence to verify a claim within a budget of five sentences is a challenge due to variations in spelling and the presence of synonyms. However, with more recent model architectures such as GENRE (De Cao et al., 2020) and RoBERTa (Liu et al., 2019b), a sentence-level recall@5 of 90% can be attained, which far exceeds the TF-IDF-only baseline recall of 44% published in Thorne et al. (2018a). Section 3.5.6 highlighted how the use of evidence in training and different sampling strategies for negative evidence impacted the classification accuracy, and ultimately the FEVER Score, when considering a pipeline of retrieval with classification. Challenges still remain on how models can best make use of retrieved evidence and whether different model architectures or training regimens can allow the veracity classifiers to attain the upper-bound FEVER Scores.

Chapter 4

Adversarial evaluation of fact verification models

This chapter considers the evaluation of adversarial attacks against fact verification systems and formed part of a shared task (Thorne et al., 2019c) with the collaboration of Christos Christodoulopoulos and Arpit Mittal, who acted in an advisory capacity. To support the shared task, two new scoring metrics were introduced and discussed in a paper which was presented at EMNLP (Thorne et al., 2019b). In addition, part of the survey in Section 4.2 was made available as an arXiv pre-print to support the shared task's participants (Thorne and Vlachos, 2019).

4.1 Introduction

This chapter considers adversarial attacks against models trained for fact verification as a means for probing model behaviour and identifying limitations in their reasoning. As fact verification is a task with potentially sensitive applications, it is critical to understand how systems and models behave when exposed to real-world data and how deficiencies in their training data may contribute to this. In related NLP tasks, it has been observed that, as models become more complex, it is difficult to fully understand

and characterise their behaviour (Samek et al., 2017). Ongoing discussions in the NLP community investigate to what extent models understand language (Jia and Liang, 2017) or exploit unintentional biases and cues present in the datasets they are trained on (Poliak et al., 2018, Gururangan et al., 2018, McCoy et al., 2019).

One of the diagnostic tools for understanding how models behave is *adversarial evaluation*, where data that is deliberately designed to induce classification errors is used to expose "blind spots" of a system. Adversarial examples have been initially studied in the field of computer vision where Szegedy et al. (2014) identified an over-sensitivity in some classifiers where imperceptible perturbations to the input image (such as altering pixel intensities by a minute amount) resulted in the models predicting different labels. These perturbations were generated by altering the model input to maximise the potential for classification error. While the proposed method of altering pixel intensities was generally imperceptible to humans, making similar perturbations to text is more challenging due to the discrete symbol space and the need to preserve grammaticality: modifying a single token may either change the label of the instance or introduce grammatical errors. There are many recently proposed techniques for generating adversarial instances for NLP tasks (surveyed in Section 6.2). These methods typically assume that the perturbations preserve the semantics of the original instance. However, they vary in the degree to which newly generated instances are grammatical and faithful to the original, i.e. didn't inadvertently cause a label-changing perturbation.

When evaluating models for claim verification with adversarial instances, it is crucial that the generated instances are correctly labelled, considering the semantics of the claim and evidence. However, as the degree of automation for attack generation increases, there may be more chances for meaning altering changes to be introduced that would inadvertently cause the label for the generated instances to change. To enable fairer comparison of both the adversarial instance generators and the systems that they are tested against, this chapter introduces two metrics: attack *potency* and system *resilience* which consider

how frequently adversarial instances induce misclassifications while considering whether the adversarial instances are correct through a separate evaluation.

This chapter evaluates three methods for generating adversarial attacks with varying degrees of automation. It further considers how fact verification systems, such as the one introduced in Chapter 3 as well as the four top-scoring systems from participants of the first FEVER shared task (Thorne et al., 2018b), perform under adversarial evaluation considering the correctness of the generated instances. The first attack, informed by model behaviour, uses Semantically Equivalent Adversarial Rules (SEARs) (Ribeiro et al., 2018), a state-of-the-art method for generating rules that perform meaning-preserving transformations to instances that induce classification errors. The second attack is informed by dataset biases, where common patterns and constructions in the claims of the FEVER dataset are identified and used to inspire the creation of a number of hand-crafted rules that are applied to the claims. The final attack is a lexically-informed approach that makes use of a paraphrase model to generate new instances with meaning-preserving variations of the claims.

4.2 Generating adversarial attacks

In contrast to the computer vision domain, where models make predictions over continuous inputs, models in the natural language processing domain operate over strings of discrete tokens. Adversarial attacks can be made by making changes to these strings, such as inserting tokens, rephrasing inputs, or appending distractor information. In the example in Figure 4.1, two different attacks are shown considering a claim from the FEVER dataset: one is a meaning-preserving paraphrase, retaining the same label as the original instance, and the second is meaning-altering where the newly generated instance has a different label to the original. As the discrete domain of NLP precludes making imperceptible changes to strings, the methods used to generate adversarial instances differ from the vision domain. The NLP community has proposed various methods for successful attacks, including manual construction, character perturbations, addition of

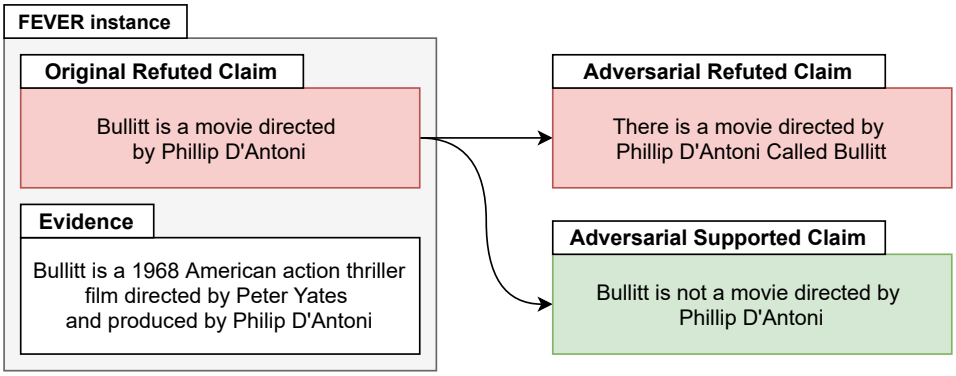


Figure 4.1: Adversarial instances generated through rule-based transformations of claims

distractor information, rule-based transformations and paraphrasing, and automated generation. In the following brief survey, common approaches are discussed, considering the trade-off between the level of automation (which allows both scale and diversity when generating new claims) and whether the perturbation unintentionally changes the label of the instance or induces a grammatical error, which would require additional human annotation to identify and resolve.

Manual construction Small adversarial datasets have been manually constructed and successfully used to identify limitations in Machine Translation (Burlot and Yvon, 2017, Isabelle et al., 2017), Sentiment Analysis (Mahler et al., 2017, Staliūnaitė and Bonfil, 2017) and Natural Language Understanding (Levesque, 2013, Bartolo et al., 2020) systems. Instances are generated that exploit world knowledge, semantics, pragmatics, morphology and syntactic variations. By manually constructing adversarial instances, the attacker would have a high degree of confidence that the text will be grammatical and that instances will be correctly labelled. However, exploiting human knowledge of language is comparatively expensive and is difficult to scale to construct larger datasets.

Distractor information Jia and Liang (2017) evaluated the addition of distractor information in reading comprehension systems. Adversarial instances are generated for the SQuAD (Rajpurkar et al., 2016) shared task (question answering against a short passage of text) by concatenating short distractor sentences to the passage. The

distractor sentences are generated by perturbing the question with entity substitution and generating a false answer which has a similar form to the actual answer with rule-based substitutions. Furthermore, the additional information concatenated to the original passage of text is by construction irrelevant, as it is about a different entity. Thus it is unlikely to cause a change to the meaning of the text requiring the instance to be labelled. Additionally, this approach does not require manual generation of the instances (human annotators are only used for filtering out ungrammatical distractors). This would be less expensive than manual construction of adversarial instances, meaning that greater scale can be achieved without the risk of annotators unintentionally introducing a bias.

Programmatic construction of adversarial dataset Naik et al. (2018) introduce a stress test evaluation dataset for NLI generated by several programmatic methods that exploit limitations and biases present in models in the context of the MultiNLI (Williams et al., 2018) shared task. The adversarial dataset was constructed by applying three types of transformations to the MultiNLI development data split: meaning-altering transformations are applied to instances that require numerical reasoning through rule-based transformation; distractor phrases that preserve meaning are appended to instances (that exploit models' biases for strong indicators for negation and sentence length); finally, perturbations to some instances are introduced to mimic typographical errors. While the rule-based changes would preserve the label and are grammatical, some of the changes in this dataset are not natural (for example, the word overlap rule which appended 'and true is true' tautology is logically correct but unlikely to occur in everyday language). This approach is relatively cheap in comparison to manual construction, as one rule can be applied to many instances.

Template-based construction Template-based generation of adversarial instances, using replacement rules written (or selected) by humans, combines the precision of manual construction and the scale of programmatic construction. Ribeiro et al. (2019) programmatically perturb dependency-parsed questions and generate new question-answer pairs with varied implications. CheckList (Ribeiro et al., 2020) allows for

human-assisted programmatic construction of templates that may be filtered by a user supervising several classes of attack (including distractors, semantic preserving changes, and semantic altering changes) to create large numbers of diverse adversarial instances which are more likely to be free of errors.

Character-level perturbation Character-level attacks have also highlighted the brittleness of NLP systems: by making letter swaps or insertions, Belinkov and Bisk (2018) and Ebrahimi et al. (2018) generated distorted examples which cause misclassifications or translation errors. While it is unlikely that a single character can unintentionally change the meaning of a sentence, this method is still *intentionally* introducing errors.

Paraphrasing Iyyer et al. (2018) and Ribeiro et al. (2018) apply paraphrase-based transformations to generate adversarial instances using alignments from parallel corpora for translation tasks. Iyyer et al. (2018) attack sentiment analysis and textual entailment recognition systems, generating paraphrases of instances with an encoder-decoder model architecture. In the process of generating adversarial instances, the meaning could be altered requiring relabelling (for example, a sentence pair with an ENTAILMENT relation may become NEUTRAL after perturbation), or the newly generated text could be ungrammatical. In an error analysis, the authors identified that in 17.7-22.3% of cases, the generated examples were not paraphrases, and in 14.0-19.3% of cases, the paraphrases were ungrammatical. Ribeiro et al. (2018) evaluate phrase substitutions, obtained from alignments from a translation task, as a method of generating semantically equivalent instances for question answering, sentiment analysis, and reading comprehension. The authors incorporate additional filtering to remove ungrammatical or unnatural instances. Rather than using human annotators, instances are ranked by the probability of obtaining the paraphrase when back-translating the instance through a pivot language.

Fully automated generation Zhao et al. (2018) generate natural language adversaries for an NLI task through the use of an auto-encoder architecture. The new instances exhibit high diversity and are regarded as "natural" with 86% of human annotators

stating that the new instances were grammatical and 81% stating that the new instances were similar to the original on a small study with 13 annotators and 20 examples. However, while this method will generate instances that are similar to the original, it is not certain that the label for the newly generated instances will be preserved as similarity does not guarantee semantic equivalence. Zellers et al. (2018) introduce an adversarial method for generating negative examples for a multiple-choice answer selection task through generating probable sentences (with the aid of a language model) that induce misclassifications. Using this approach, it is not possible to generate examples that preserve or deterministically alter the entailment relation, again requiring the newly generated instances to be labelled by humans.

4.3 Evaluating adversarial attacks

Consider a method for generating adversarial instances (hereafter referred to as an adversary), a, that generates a set of instances $X_a = \{x_{a,i}\}_{i=1}^{N}$ with accompanying labels $Y_a = \{y_{a,i}\}_{i=1}^{N}$. To evaluate such adversaries, both their correctness and their effect on a system under test must be considered. Instances are correct if they are grammatical, appropriately labelled, and meet the task requirements. The correctness rate for an adversary, c_a, is estimated through a binary annotation of instances generated by the adversary. As it is not always feasible to annotate every instance generated by an adversarial attack, the correctness of adversaries can be estimated on a sample.

The effectiveness of an adversary, against a system's, s, evaluation measure f (such as F_1 or FEVER Score), on a set of predictions made by the system, represented $\hat{Y}_{s,a}$, is measured through *potency*, defined in Equation (4.1). Intuitively, better adversarial instances induce more misclassifications, resulting in a lower evaluation measure. Assuming the evaluation measure is a real value in the range $[0, 1]$, the potency score is the average failure rate $(1 - f(\cdot))$ across all systems, $s \in S$ weighted by the correctness rate of the adversary, c_a. To illustrate the need to incorporate correctness into evaluation, the *raw potency* (where the correctness rate c_a is set to 1) will be compared.

$$\text{Potency}(a) \triangleq c_a \frac{1}{|S|} \sum_{s \in S} \Big(1 - f(\hat{Y}_{s,a}, Y_a)\Big) \tag{4.1}$$

A system that is resilient will have fewer errors induced by the adversarial instances. Systems should be penalised more for making mistakes on instances from adversaries with a higher correctness rate. In contrast, an adversary that generates incorrect instances shouldn't strongly impact a system's resilience. *Resilience* is defined in Equation (4.2) as the average score, weighted by the correctness rate for each adversary, $a \in A$:

$$\text{Resilience}(s) \triangleq \frac{\sum_{a \in A} c_a f(\hat{Y}_{s,a}, Y_a)}{\sum_{a \in A} c_a} \tag{4.2}$$

4.4 Generating adversarial attacks for FEVER

Each FEVER instance comprises a claim, a label, and evidence for the SUPPORTED and REFUTED classes. Adversarial attacks could be applied either to the claim or the evidence, making modifications to these inputs. However, within the context of the FEVER shared task, the evidence that the models consider is retrieved from a fixed snapshot of Wikipedia. While it is important to understand how changes to evidence affects the model (explored in Chapter 5), this chapter will focus on adversarial attacks by modifying the claim and studying how this impacts evidence retrieval and claim veracity prediction (with retrieved or oracle evidence).

The adversarial instances generated in this chapter will modify the claims but retain the evidence from the original instance. A new label will be assigned, considering the interaction between the new claim and the original evidence. This is to mitigate the need to find new evidence for the generated adversarial instances, a process that is rather error-prone if done automatically or costly if done manually. Three methods for generating adversarial instances by modifying existing dataset instances will be evaluated: manually crafted rule-based transformations, informed by the training data; a recently proposed method for automatically generating Semantically Equivalent Adversarial Rules

Transformation	Pattern	Template
Entailment Preserving	(.+) is a (.+) (.+) (?:was | is)? directed by (.+)	There exists a $2 called $1 $2 is the director of $1
Simple Negation	(.+) was an (.+) (.+) was born in (.+)	$1 was not an $2 $1 was never born
Complex Negation	(.+) (?:was | is)? directed by (.+) (.+) an American (.+)	There is a movie called $1 which wasn't directed by $2 $1 $2 that originated from outside the United States.

Table 4.1: Example rule-based attacks that preserve the entailment relation of the original claim (within the definition of the FEVER shared task), perform simple negation and more complex negations. The matching groups within the regular expression are copied into the template (variables begin with $).

(Ribeiro et al., 2018, SEARs) that target a model; and a lexically-informed method which generates meaning-preserving modifications of claims using a paraphrasing model. Each method will be applied to the FEVER development dataset yielding a set of new adversarial instances. An adversarial dataset will be constructed by combining a stratified sample of 5000 instances from each adversary, balanced by class distribution.

Dataset-targeted adversary

This adversary assumes access to the dataset used to train the models, and it identifies lexico-syntactic patterns in the claims that are highly frequent. The claims following these patterns undergo rule-based transformations to generate new instances with patterns not encountered in the training data. These rule-based transformations rewrite the claim and can either preserve the entailment relation between the claim and the evidence or negate the claim, as illustrated in Table 4.1. The rules were created through manual inspection of the most frequent bigrams within the claims and, for the most common patterns, writing regular expressions that match the subject of the claim and its properties, listed in Appendix A.4, taking advantage of the fact that the annotators were asked to make claims referring to a single property of an entity during the claim construction process.

Using the subject and properties from the claim matched by the regular expression, templates were written that incorporate the matched tokens into a new sentence. The entailment-preserving transformations use straightforward rules such as switching from active to passive voice while retaining the original label or using alternative phrasings for a claim. For transformations that change the labels from SUPPORTED label to REFUTED and vice versa, negations were applied by changing the claim's verb phrase. More complex negations were introduced by combining both rules. In total, 103 rules were constructed: 41 were entailment preserving, 25 were simple changes to negate claims, and 37 were complex changes, combining the alternative phrasings from the entailment preserving changes with negations, listed in Appendix A.5. For the entailment-preserving rules, instances may be generated from all three classes. However, as the entailment-changing negations would require re-annotating evidence for the NOTENOUGHINFO classes, it is not possible to reliably generate adversarial instances where negation is required.

Lexically-informed adversary

This family of adversaries generates label-preserving paraphrases of the claim using lexical resources without human intervention. Two methods will be evaluated: the first method, following Iyyer et al. (2018), uses a paraphrasing model to translate a claim to a foreign language and back-translate it to English. To encourage variation, instances (that differ from the original claim) will be sampled from a beam-search with 5 beams.

More variation can be encouraged by substituting tokens with similar ones from WordNet (Miller, 1992). The SpaCy parser is used to part-of-speech tag the claim, c, and while retaining the named entities, nouns and adjectives are replaced with lemmas sampled from matching synsets. To realise the surface forms for each of the lemmas, the same back-translation model is used to translate the modified claim c' to pivot language and back-translating into English, yielding c''. As inadvertently matching the wrong synset may cause a meaning-altering change, the instances are scored and filtered using the ratio of translation probabilities between the paraphrased and the original sentences $\frac{P(c''|c)}{P(c|c)}$, similar to (Ribeiro et al., 2018), and the top-scoring 5% of are retained.

Model-targeted adversary

Targeting adversarial attacks against a model, SEARs (Ribeiro et al., 2018) generates transformation rules for creating adversarial instances by perturbing instances and evaluating whether the perturbations induce misclassifications in the model. The perturbations used for generating the transformation rules are generated by paraphrasing instances using a back-translation model. A shortlist of potent non-redundant rules is collected by ranking the rules which induce the most misclassifications. The SEARs are generated by targeting three models for FEVER: the Decomposable Attention model, ESIM and BERT models, presented in Chapter 3, which allows evaluation of how the different model properties affect the potency of the generated attacks. In addition, to evaluate the importance of generating adversarial instances specific to the task and domain, adversarial instances from SEARs generated by querying a classifier trained on an unrelated sentiment analysis task will be also be compared.

4.5 Experimental setup

The adversarial instances are generated by applying each adversary to claims sampled from the FEVER development set and making modifications to the claim and label where appropriate, retaining the original evidence. The potency of the adversarial instances is computed by evaluating predictions made using the highest-performing model from Chapter 3, which was based on GENRE (De Cao et al., 2020), BM25 (Robertson and Walker, 1994), and RoBERTa (Devlin et al., 2019), and the four highest-performing models from the FEVER shared task: three of which were ESIM-based (Chen et al., 2017b) and one which was based on the OpenAI Transformer (Radford et al., 2018). The transformer-based model was from Team Papelo (Malon, 2018), and the ESIM-based models were the Neuro-Semantic Matching Network from Team UNC (Nie et al., 2019b), the winner of the task; HexaF from Team UCLMR (Yoneda et al., 2018); and the Enhanced ESIM model from Team Athene (Hanselowski et al., 2018), which were ranked second and third place.

In Section 4.6.1, a further evaluation on the component parts of the systems will be performed. Because the components considered in Chapter 3 have the same experimental controls, they will be directly compared in this study. In contrast, the systems used in the shared task were engineered by different teams to compete on a leaderboard with fewer controls. This section will consider combinations of claim-veracity classifiers (ESIM, BERT and RoBERTa) against oracle evidence (which assumes perfect information retrieval), and the TF-IDF, BM25, and GENRE evidence retrieval approaches, evaluating how errors are introduced in systems built on pipelines of these components.

Instance correctness, which calibrates the potency and resilience scores, is measured through blind annotation of a sample of 900 instances (approximately 100 per adversary). Each of these instances is annotated for grammaticality and whether the evidence supports the labelled claim, following a variation of the manual error coding process described for validating the FEVER dataset construction, listed in Section Appendix A.6.

4.6 Results

Applying each adversary to the instances in the FEVER development set generated numerous new adversarial instances. The training data targeted rule-based adversary created 110,000 new claims. The lexically-informed adversary created new 270,000 instances. And, using SEARS generated approximately 300,000 instances for each of the models used to initialise the rules (the BERT, ESIM and DA fact verification models and a pre-trained Sentiment Analysis model). A balanced adversarial test set was created through a stratified sample of 5000 instances from each of these methods. The *potency* of each adversary and *resilience* of each system, considering instance correctness, is reported in Table 4.2 and Table 4.3 respectively.

Considering potency, the highest-scoring adversary was the *negate* rule-based method, where manually constructed templates that negated the claims were constructed, informed by observations on the training set. While the raw potency (not considering the correctness of instances) of SEARS targeting a Decomposable Attention model was higher than the

Rank	Method	Raw Potency (%)	Correct Rate (%)	Potency (%)
1	Rules Negate	42.14	95.7	40.33
2	Rules Simple	43.15	91.8	39.61
3	Rules Complex	44.60	88.5	39.47
4	Paraphrase	46.30	79.4	36.76
	Unmodified instances	*37.04*	*91.2*	*33.78*
5	SEARS Sentiment	34.91	66.9	23.36
6	Paraphrase + WordNet	57.28	33.3	19.08
7	SEARS BERT	49.58	27.5	13.63
8	SEARS DA	55.39	18.3	10.13
9	SEARS ESIM	51.13	19.6	10.02

Table 4.2: Potency of adversaries where correctness rate is estimated using inspection of the generated instances. The baseline method of sampled instances is not used for scoring resilience. Raw potency is potency score without considering the correctness of instances.

Rank	System	Architecture	FEVER Score (%)	Resilience (%)
1	Pipeline	Transformer	69.13	60.74
2	Papelo	Transformer	57.36	55.52
3	UNC	ESIM	64.21	55.02
4	UCLMR	ESIM	62.52	54.43
5	Athene	ESIM	61.58	51.25

Table 4.3: Systems ranked by the resilience to adversarial attacks. The FEVER Score column uses reported scores from the shared task.

rule-based adversary, a large number of instances generated from this method failed to meet the guidelines for the task, resulting in the lowest potency when accounting for correctness. Similar findings were observed for SEARS generated on other models, where the adversaries had potency scores lower than a sample of unmodified instances from the shared task due to low correctness rates.

For the rule-based method, the largest proportion of erroneous instances were generated by rules that inadvertently matched sentences that weren't encountered in the observations used to design the rules. For example, the rule that transforms the pattern: 'X is a Y' to 'There is a Y called X' correctly generates instances in most cases, but for the sentence "Chinatown's producer is a Gemini" which contains a compound noun, the adversary "There is a Gemini called Chinatown's producer" was generated. Similar errors were

System	FEVER Score (%)								
	Rules			Paraphrase		SEARS			
	Simple	Negate	Cmplx	Raw	+WN	BERT	ESIM	DA	Sentiment
Pipeline	60.47	65.41	63.99	58.63	45.55	52.99	51.69	49.03	69.29
Papelo	62.71	52.34	47.92	56.65	43.39	53.69	52.67	46.89	68.91
UNC	56.33	58.76	56.16	52.71	41.85	49.19	46.19	43.37	63.91
UCLMR	58.41	56.74	54.72	51.37	41.51	48.55	46.91	42.93	63.09
Athene	46.33	56.04	54.22	49.15	41.27	47.69	46.97	40.81	60.23

Table 4.4: Breakdown of FEVER Scores of each system to each adversarial attack prior used for calculating resilience and potency. Lower scores indicate stronger attacks (contributing to potency). Higher scores indicate stronger systems (contributing to resilience). Scores in this table do not account for the correctness of instances.

produced when the regular expressions also matched determiners and included these in the new sentence, which altered the semantics or rendered the claim ungrammatical.

The potency of SEARs is dependent on the model and dataset used for generating the Semantically Equivalent Adversarial Rules. When the rules were generated using the transformer-based fact verification model, the rules induced more errors than rules generated against an out-of-domain sentiment analysis model. In contrast, when the rules were generated using the sentiment analysis model, the instances were correctly predicted by the classifier more often for two reasons: the perturbations were not targeted against this class of model, and furthermore, the newly generated instances were more semantically similar, such as the replacement of 'movie' with 'film'. A common failure mode from all variants of SEARs was replacing indefinite articles (such as *a*) with definite articles (such as *the*), making claims nonsensical or altering the semantics to the point where the label changed. The SEARs also often made changes to determiners and quantifiers or deleted terms such as *only*, which altered the semantics of a claim, making it incorrectly labelled. The SEARS targeting the FEVER classifier introduced other phrases and distractor information, such as replacing a period with ', not a movie' *sic*, which was not always grammatical and only supported by the evidence in limited circumstances resulting in a much lower correct rate and corresponding potency.

With the exception of Papelo, the resilience of the systems is well correlated with the performance on the FEVER shared task with the system ranking preserved. Both the transformer-based models: the RoBERTa-based one introduced in Chapter 3 and Papelo, which was initialised with weights from Radford et al. (2018), had the highest resilience scores as they were more accurate for the correct adversarial instances. Meanwhile, the other systems from the shared task, which are all based on ESIM, have moderately lower accuracy. Breaking down the FEVER Scores by system and adversary (reported in Table 4.4), most models were worst by the instances generated by SEARS. However, as the correctness rate for these instances was low, these had a lesser impact on the resilience score than the adversaries that were generating valid instances.

4.6.1 Component-wise evaluation of rule-based adversary

All the systems under test were pipelines of an information retrieval system, shortlisting documents and evidence sentences, and a claim veracity classifier. These components may behave differently under adversarial evaluation, with errors propagating from one component to the next. This section will present a more detailed evaluation of the systems, considering the behaviour of these components in isolation as well as when part of a pipeline. For this study, *only* the rule-based adversaries will be used because the instances generated with this deterministic method had high correctness rates. A direct comparison will be made between the system's performance on both the newly generated adversarial instances (reported in *adversarial* columns) as well as the original instances that these were derived from (reported in *original* columns).

Claim veracity classifier

Adversarial instances caused a reduction of the systems' FEVER Score, which accounts for label accuracy and the recall of the evidence sentences. Using claim veracity models from Chapter 3, based on RoBERTa, BERT, and ESIM, combined with oracle evidence (simulating perfect information retrieval), the label accuracy, presented in Table 4.5, decreased by 27 percentage points for ESIM and approximately 20 percentage points

Model	Accuracy (%)			FEVER Score (%)		
	Original	Adversarial	Delta	Original	Adversarial	Delta
Oracle + ESIM	76.82	49.15	-27.68	=	=	=
Oracle + BERT	85.37	61.96	-23.41	=	=	=
Oracle + RoBERTa	86.92	67.37	-19.55	=	=	=
Papelo	74.49	55.36	-19.13	72.11	54.35	-17.76
UNC	72.03	49.95	-22.08	69.44	48.65	-20.79
UCLMR	76.61	53.56	-23.05	68.19	46.65	-21.54
Athene	67.85	37.94	-29.91	61.84	32.73	-29.11

Table 4.5: Summary of label accuracy and FEVER Scores for instances used in rule-based adversarial attacks. For the case of the oracle evidence retrieval component, FEVER Score is equal to accuracy.

for the transformer variants. The RoBERTa model, which has the same underlying architecture as BERT, but with different pre-training, attains a higher accuracy on the original task as well as a lower delta under adversarial attack. This indicates the benefits of the modified pre-training, which combines a different masking objective, more data and larger batch sizes, on the model's resilience when applied to out-of-domain data.

Considering the shared task models, the ESIM-based classifier from the Athene system had the highest reduction in accuracy. Even though evidence recall wasn't greatly affected (a reduction of 1.7%, discussed in the following section), the FEVER Score reduced by 29.91%. On inspection, this model is mostly predicting Supported for the adversarial instances. In contrast, the transformer-based model from Papelo had the lowest reduction in FEVER Score (19.13%) despite a decrease in evidence recall of 21.05%. Papelo exhibited similar behaviour by predominantly predicting one class. However, as this class was NotEnoughInfo, which doesn't require evidence for scoring, the resulting FEVER Score was higher, despite similar deficiencies in the system.

Evidence retriever

Most of the systems from the shared task either incorporated TF-IDF or keyword matching in their information retrieval component for document retrieval. This design choice made systems resilient to the rule-based transformations, which mostly added

Model	Precision (%)			Recall (%)		
	Original	**Modified**	**Delta**	**Original**	**Modified**	**Delta**
Papelo	94.41	96.45	2.04	62.75	41.70	-21.05
UNC	41.99	45.06	3.07	72.31	67.29	-5.02
UCLMR	47.66	44.44	-3.22	70.79	68.50	-2.30
Athene	24.80	21.94	-2.86	78.23	74.84	-3.39

Table 4.6: Effect of rule-based adversarial attacks on the evidence retrieval component of the pipelines considering sentence-level accuracy of the evidence.

stop words and reordered the words within the sentence. Most systems exhibited a minor reduction in recall, listed in Table 4.6, but only Papelo was greatly affected. Papelo maintained very high precision with the newly generated instances, but the loss of recall indicates a brittleness towards instances that differed from the distribution of language patterns present in the training set. In contrast to the retrievers listed in Table 4.7 which perform binary classification for relevance, Papelo models sentence selection as a claim veracity prediction, discarding the evidence which causes a NotEnoughInfo prediction. Similar patterns were observed when using transformers on out of domain data in Chapter 3, where the transformer-based entailment classifiers had near-random accuracy on the out-of-domain FEVER data whereas ESIM models, which couldn't model entailment as accurately, were more resilient to the out-of-domain data. The other shared task systems were based on an ESIM binary classifier and used other techniques which all may contribute to their resilience: training the retriever using a hinge-loss objective, increasing the margin between relevant and irrelevant evidence sentences (Hanselowski et al., 2018, Athene); aggregating sentence retrieval results using a downstream model (Yoneda et al., 2018, UCLMR); or consider the unnormalised logits in retrieval rather than the probability of being evidence (Nie et al., 2019b, UNC).

For combinations of page retrievers and sentence classifiers considered in Chapter 3, listed in Table 4.7, the precision and recall of the selected evidence sentences are not greatly affected. These systems used a two-step retrieval that first retrieved relevant pages and then independently performed a binary classification over sentences from the top k retrieved documents. The best performing system used GENRE (De Cao et al.,

Retriever		Precision@5 (%)			Recall@5 (%)		
Page	**Sentence**	**Original**	**Modified**	**Delta**	**Original**	**Modified**	**Delta**
GENRE	ESIM	30.19	28.33	-1.86	86.56	84.86	-1.70
	BERT	35.33	31.81	-3.52	87.93	85.35	-2.59
	RoBERTa	36.47	37.27	0.80	87.57	84.1	-3.47
GENRE+BM25	ESIM	31.34	29.37	-1.97	87.22	85.54	-1.68
	BERT	35.84	31.59	-4.25	88.43	86.01	-2.42
	RoBERTa	37.57	38.11	0.54	88.22	85.08	-3.14
BM25	ESIM	32.42	29.39	-3.03	62.84	57.74	-5.10
	BERT	43.40	37.82	-5.58	63.75	58.22	-5.54
	RoBERTa	47.65	45.97	-1.69	63.57	58.52	-5.05
TF-IDF	ESIM	36.95	32.97	-3.98	58.47	59.52	1.05
	BERT	48.33	43.32	-5.00	59.19	60.13	0.94
	RoBERTa	52.08	50.68	-1.39	59.13	60.33	1.20

Table 4.7: *Sentence-level* evidence retrieval precision@5 and recall@5 under rule-based adversarial attack (for the shared task, the top-5 sentences are considered for scoring).

Retriever	Precision@3 (%)			Recall@3 (%)		
	Original	**Adversarial**	**Delta**	**Original**	**Adversarial**	**Delta**
GENRE	22.13	20.95	-1.17	93.33	94.01	+0.68
GENRE+BM25	22.28	21.05	-1.23	93.73	94.31	+0.58
BM25	12.42	9.58	-2.84	65.91	60.94	-4.97
TF-IDF	10.99	10.71	-0.28	61.64	64.20	+2.56

Table 4.8: *Page-level* retrieval modules on claims before and after adversarial rules are evaluated, considering precision and recall at 3 (the optimal value used in Chapter 3).

2020), a sequence-to-sequence neural model, to predict the Wikipedia page, which has not been subject to adversarial evaluation in previous literature. The recall of the best performing system from Chapter 3, GENRE+BM25 with RoBERTa, was only reduced by 3.14%. Even though the adversarial instances introduced new linguistic patterns in the claim, the sentence classifiers were largely resilient to this change.

The page-level retrieval is evaluated in isolation in Table 4.8. Adversarial instances did not affect TF-IDF or GENRE to the same extent as the NLI or sentence selection components. Surprisingly, the recall increased for all page-level retrievers except for BM25. With BM25, pages containing entity mentions related to the query were often retrieved. However, while these pages contained entity mentions, they were not evidence.

4.7 Summary

Adversarial evaluation provides one mechanism for understanding model behaviour when exposed to patterns not encountered during training. As automated means for generating adversarial instances do not always produce grammatical or correctly labelled instances, it is important that any evaluation (either of the system or the adversary) considers how often the generated instances meet the requirements of the underlying task. This chapter introduced two metrics for the *potency* of attack and *resilience* of systems to these attacks, considering the instance correctness.

Four top-performing systems from the FEVER shared task (Thorne et al., 2018b) and the models from Chapter 3 were evaluated under adversarial evaluation. The instances were generated from three families of methods for adversarial attacks that varied in the degree of automation. While the manually-written rules exhibited a high correctness rate, the automated methods often made grammatical errors or label-altering changes which lowered the correctness rate. This shortcoming was captured by these metrics and the trade-off between correctness and impact on system scores: despite the automated SEARS-based adversarial instance generation method having the greatest impact on systems under test (causing the most errors), it was not the most potent, owing to the low correctness rate.

Chapter 5

Mitigating biases captured in claim verification models

This chapter considers mitigating the biases captured in the text-pair classification models used for tasks such as claim verification and natural language inference and was published as Thorne and Vlachos (2021a).

5.1 Introduction

A number of recent works have illustrated shortcomings in sentence-pair classification models used in tasks such as natural language inference and claim verification. These arise from limited or biased training data and the lack of suitable inductive bias in models. Naik et al. (2018) demonstrated that phenomena such as the presence of negation or a high degree of lexical overlap induce misclassifications on models trained on the MultiNLI dataset (Williams et al., 2018). Poliak et al. (2018) and Gururangan et al. (2018) identified biases introduced during the construction of NLI datasets that were exploited by models to learn associations between the label and the hypothesis sentence without considering the premise – known as *hypothesis-only* bias. This hypothesis-only bias has also been shown to affect models trained on the FEVER dataset (Schuster et al., 2019), where

models accurately predict the veracity of the claim without considering the interaction between the claim and the evidence.

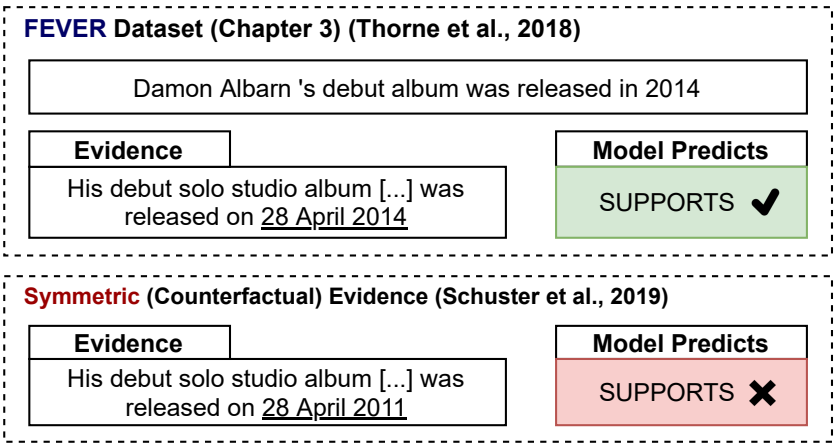


Figure 5.1: Hypothesis-only bias in FEVER contributes to low accuracy when testing against counterfactual evidence (i.e. evidence that is modified to change the label of the instance's veracity). In this example, making a change to the evidence so that the claim is refuted does not cause the model to change the prediction.

Undesirable behaviours in text-pair classifiers, such as the hypothesis-only bias, under-sensitivity to negation or lack of numerical reasoning, can be mitigated through techniques that modify the weights within models, during or after training. Training methods such as Debiased Focal Loss (DFL) and Product of Experts (POE) (Mahabadi et al., 2020) require architectural changes to separately encode and penalise biases: the networks are augmented with an additional classifier that is used to quantify the bias and reduce the weight of the loss for 'biased' instances during training. Schuster et al. (2019) similarly re-weight the loss of instances during training. Rather than dynamically computing instance weights by augmenting the model, instances are statically weighted before training using the mutual information between tokens and the instance labels. Suntwal et al. (2019) delexicalise instances by replacing tokens with placeholders which prevents classifiers from exploiting mutual information between domain-specific noun-phrases and class labels. After a model is trained on one dataset, Liu et al. (2019a) *innoculate* models to undesirable behaviours by continuing training on a small number of instances targeting the bias in a process referred to as *fine-tuning*. The benefit of using fine-tuning is that

the models can be de-biased without the need to make architectural changes or retrain models allowing application of pre-existing systems, such as the models that are FEVER shared task models susceptible to adversarial attacks, discussed in Chapter 4.

Fine-tuning and instance weighting are multi-objective optimisation problems: parameterising the model to accurately classify instances from both the original dataset as well as a different evaluation set where the biases are removed. For fine-tuning specifically, the trade-off in accuracy on the original task in favour of the fine-tuning task is a form of catastrophic forgetting (French, 1999), as parameters for the original task are overridden during fine-tuning. In previous work on domain adaptation for machine translation (Thompson et al., 2019, Saunders et al., 2019), regularising fine-tuning with Elastic Weight Consolidation (Kirkpatrick et al., 2017, EWC) minimises catastrophic forgetting by penalising weight updates to parameters crucial to modelling the original dataset. Extending this line of research further, this chapter evaluates whether it is beneficial to use EWC to mitigate biases present in the sentence-pair classification models used in tasks such as claim verification and natural language inference.

This chapter investigates bias mitigation using fine-tuning and the techniques proposed by Mahabadi et al. (2020) for popular architectures for text-pair classification tasks. In contrast to Schuster et al. (2019), which assumes the bias is captured as the presence certain of n-grams which have high mutual information with the label, fine-tuning does not make any *a priori* assumptions about the structure of the bias. Models for FEVER and MultiNLI (Williams et al., 2018) are evaluated by considering fact verification instances with counterfactual evidence (Schuster et al., 2019) and the NLI lexical bias stress-test datasets released by Naik et al. (2018), respectively. On all experiments on the FEVER dataset, fine-tuning with counterfactual evidence mitigated hypothesis-only bias, increasing the absolute accuracy of a BERT-based model by approximately 10%. However, without EWC, accuracy on the original dataset was reduced from 87% to 79%, whereas with EWC, catastrophic forgetting was mitigated, and accuracy was 82%. In all experiments with EWC, the original task accuracy was significantly higher than

fine-tuning without regularisation and fine-tuning with L2 regularisation. These gains were attained while maintaining similar performance on the fine-tuning data, indicating that equivalent gains in accuracy can be made with less forgetting of the original dataset. Similar patterns were observed when fine-tuning MultiNLI models with the stress-test data (Naik et al., 2018). Additionally, the experimental results show that fine-tuning methods can be combined with instance-weighting during training to mitigate hypothesis-only bias. Fact verification models trained with POE and DFL (Mahabadi et al., 2020) to mitigate hypothesis-only bias can be further fine-tuned, yielding improvements on both the original task and the debiased symmetric dataset.

5.2 Quantifying the effect of hypothesis-only bias in fact verification datasets

Recent datasets for fake news detection and fact verification, surveyed in Chapter 2, such as LIAR-Plus (Alhindi et al., 2018) and MultiFC (Augenstein et al., 2019), have the same task signature as FEVER: a text-pair classification with a claim and supporting or refuting information. However, in these cases, this rationale that is provided with the claim is not necessarily evidential. All these tasks potentially exhibit a hypothesis-only bias where information from the claim can be used to predict the label without considering the second sentence in the text-pair.

For FEVER, this bias is introduced through the synthetic generation of the claims and can be more problematic than the biases that occur in datasets with naturally occurring claims as it does not reflect the biases present in the real world. In Liar and MultiFC, on the other hand, the claims arise from real-world events, and the biases in the data reflect political viewpoints and the choices made by fact-checkers when prioritising their work rather than the imagination used by the annotators when generating mutations as in FEVER. Using a model trained to predict claim veracity in both claim-only and text-pair setups provides some insights into the interaction between the claim and

its supporting information. A RoBERTa model is trained as a baseline using widely accepted hyperparameter choices for the tasks (listed in Appendix A.7) with results on the development sets presented in Table 5.1.

For both MutiFC and LIAR, the claim-only models, trained with only one of the two text inputs, outperforms the text-pair setup. In contrast, as FEVER is the only task that was designed with claims that require the use of evidence for verification, the text-pair accuracy is higher than the claim-only accuracy indicating that the evidence is helpful to the model rather than acting as a distraction. Even though the FEVER classifiers attain higher accuracy considering the interaction between a claim and evidence, there are still many instances (in both the binary setting, where instances labelled NotEnoughInfo are discarded, and the full task setting) where the model predicts the correct label without evidence. In practice, this may mean that as different evidence is retrieved or the evidence changes, the model may not be sensitive to this change, using patterns present in the claim alone for predicting the veracity label.

The bias analysis of Schuster et al. (2019) only considers the binary task setting and does not consider the influence of the instances labelled as NotEnoughInfo, which, as indicated by the results in Table 5.1, reduces the dependency on the claim alone. In the following sections, mitigating the claim-only bias in FEVER, experiments will be performed in the full task setting with instances from all classes.

Dataset	**Accuracy (%)**	
	Claim Only	**Text Pair**
Liar-Plus	28.74	20.48
Liar-Plus (binary)	72.59	70.48
MultiFC	46.02	44.83
FEVER	61.50	88.93
FEVER (2-way)	79.09	92.24

Table 5.1: Validation accuracy for claim-only vs sentence pair classification for fact verification datasets trained on RoBERTa. For 2-way FEVER instances labelled NotEnoughInfo are discarded. For binary Liar-Plus, all positive labels are mapped to true and all negative labels are mapped to false with neutral instances discarded.

5.3 Method

5.3.1 Minimising catastrophic forgetting

To ameliorate catastrophic forgetting when fine-tuning, where model parameters over-adjust to the instances targeting the bias, one can regularise the parameter updates so that they do not deviate too much from the original training, similar to the intuition behind multi-task training approaches (Ruder, 2017). Elastic Weight Consolidation (Kirkpatrick et al., 2017, EWC) penalises parameter updates according to the model's sensitivity to changes in these parameters. The model sensitivity is estimated by the Fisher information matrix, which describes the model's expected sensitivity to a change in parameters, and near the (local) minimum of the loss function used for training is equivalent to the second-order derivative:

$$F = \mathbb{E}_{(x,y)\sim\mathcal{D}_{original}}[\nabla^2 \log p(y|x;\theta)] \tag{5.1}$$

When fine-tuning with EWC (referred to as FT+EWC), the Fisher information is used to elastically scale the cost of updating parameters θ_i from the original value θ_i^*, controlled by the λ hyperparameter, as follows:

$$\mathcal{L}(\theta) = \mathcal{L}_{FT}(\theta) + \sum_i \frac{\lambda}{2} F_{i,i}(\theta_i - \theta_i^*)^2 \tag{5.2}$$

For efficiency, the *empirical Fisher* (Martens, 2020) estimate is often used: diagonal elements are approximated through squaring first-order gradients from a sample of instances, recomputed before each epoch. If the Fisher information is not used (i.e. $F_{i,i} = 1$), Equation (5.2) is equivalent to L2 regularisation (referred to as FT+L2).

5.3.2 Combining fine-tuning with instance weighting

This chapter will also evaluate whether fine-tuning can also be applied to models trained with instance weighting. A general text-pair classifier trained for claim verification

encodes the claim and evidence $f(g(c), g(e))$ and predicts the veracity. When fine-tuning, the biases captured in the parameters classifier f and encoder g, are updated to mitigate the bias. In contrast, the instance weighting method from Mahabadi et al. (2020) uses a second *bias branch* of the model with the same encoder, $h(g(c))$, to weight instances at training time to mitigate biases before being discarded at test time. Despite mitigating some biases, the parameters in f and g could be updated to further mitigate the remaining biases after training with additional fine-tuning.

Mahabadi et al. (2020) consider two loss functions for weighting instances, outlined in Equation (5.3) and Equation (5.4) which simultaneously[1] train the claim veracity classifier with the bias-branch model: Product of Experts (POE), which sums the log-probabilities of the predictions from the text-pair and bias branch models, and Debaised focal loss (DFL), which multiplies the log probabilities of the text-pair model by compliment probabilities of the bias branch model. The intuition is that a lower loss will be assigned to instances that the bias branch model confidently predicts the class of, reducing the instance's impact during training.

$$\mathcal{L}_{POE}(\theta_f; \theta_g; \theta_h) = -\sum_{i=1}^{N} \log \sigma\Big(\log P(y_i|g(x_i); \theta_g; \theta_h)) + \log P(y_i|x_i; \theta_f; \theta_g)\Big) \tag{5.3}$$

$$\mathcal{L}_{DFL}(\theta_f; \theta_g; \theta_h) = -\sum_{i=1}^{N} (1 - P(y_i|g(x_i); \theta_g; \theta_h))^{\gamma} \log P(y_i|x_i; \theta_f; \theta_g) \tag{5.4}$$

5.4 Experimental setup

The experiments in this chapter first evaluate the application of EWC to minimise catastrophic forgetting when mitigating model biases for fact verification. A comparison of accuracy on the original test set as well as the fine-tuning test set (hereafter referred

[1]The bias branch model is independently trained with supervision to predict the correct claim label with cross-entropy loss at the same time as the text-pair classifier. However, when the instance weights from the bias branch model are used to scale the cross-entropy loss term for the text-pair classifier, the loss from the text-pair classifier is not back-propagated to the bias branch model.

to as FT-test) will be made for an untreated model (original), fine-tuning (FT) with instances from the fine-tuning training set (hereafter referred to as FT-train), regularised fine-tuning with FT+EWC and FT+L2, as well as merging instances from FT-train when training on the original task (Merged). Each model is first trained using the original dataset and splits from the respective task before fine-tuning, using the AllenNLP implementations (Gardner et al., 2018) with default hyperparameters and tokenised with SpaCy or pre-trained transformer tokenisers. Five random initialisations of each model are trained, and the mean accuracy, standard deviation, and p-value with an unpaired t-test will be reported.

For fine-tuning, the learning rate, regularisation strength λ, and the number of epochs are selected through 5-fold cross-validation on the FT-train data, selecting the model with the highest FT-train accuracy. 30 hyperparameter choices were evaluated with grid search over 10 choices for regularisation strength between 10^6 and 10^8 and 3 choices of learning rates in $\{2 \cdot 10^{-6}, 4 \cdot 10^{-6}, 6 \cdot 10^{-6}\}$ for transformer models and $\{2 \cdot 10^{-4}, 4 \cdot 10^{-4}, 6 \cdot 10^{-4}\}$ for ESIM models. The models underwent fine-tuning for a maximum of 8 epochs. For the transformer-based models, the highest cross validation accuracy on the FT-train dataset was achieved with $LR = 4 \cdot 10^{-6}$, $\lambda = 10^7$ and 6 epochs. For the ESIM-based models, the highest FT-train accuracy was achieved with $LR = 2 \cdot 10^{-4}$, $\lambda = 10^7$ and 5 epochs. Full hyperparameter choices are listed in Appendix A.7.

Mitigating hypothesis-only bias in fact verification: In this setting, oracle evidence will be used, and the models will be trained with negative instances from retrieved Wikipedia articles for NEI claims as described in Chapter 3. Hypothesis-only bias will be evaluated using the dataset of instances with symmetric counterfactual evidence released by Schuster et al. (2019). While the authors evaluated this in the context of binary veracity prediction (where claims are supported or refuted), the presence of the NEI class changes the model behaviour and it is imperative to explore model behaviour in the context it will be used in at run-time.

The dataset contains 1420 instances, approximately 1% the size of FEVER, split into two equal-sized development and test partitions. The availability of this counterfactual data means that it is possible to experiment with fine-tuning as a mitigation strategy, using the published development and test data as FT-train and FT-test, respectively. These instances can be used to mitigate the hypothesis-only bias in a model as the counterfactual evidence reduces the mutual information between n-grams in the claims and the instance labels. Following Schuster et al. (2019)'s evaluation, two ESIM (Chen et al., 2017b) variants (one with GloVe embeddings (Pennington et al., 2014) and one with ELMo embeddings (Peters et al., 2018)), and a BERT-based (Devlin et al., 2019) transformer model will be trained and and evaluated. Additionally, the more recent RoBERTa model (Liu et al., 2019b), will also be compared as it has been shown to be more robust to adversarial testing (Bartolo et al., 2020).

Mitigating model limitations in NLI stress tests: To evaluate the benefits of EWC in other domains, beyond FEVER, model limitations are mitigated by fine-tuning with NLI stress tests. The MultiNLI task (Williams et al., 2018) considers the relation between two texts, where a classifier is trained to predict whether a hypothesis is entailed by a premise. Naik et al. (2018) identify limitations of models trained on this dataset where six phenomena: *presence of antonyms*, *numerical reasoning*, *word overlap*, *negation*, *length mismatch*, and presence of *spelling errors* were evaluated with 'stress tests'. When these stress test datasets were used to fine-tune models to inoculate models against these limitations (Liu et al., 2019a), catastrophic forgetting was observed when mitigating against the presence of antonyms.

Following the experimental setup from Liu et al. (2019a), this chapter will attempt to mitigate biases through fine-tuning an ESIM (Chen et al., 2017b) and a Decomposable Attention (Parikh et al., 2016) model on stress-test data. In addition, however, this chapter will compare the effect of regularisation with EWC to evaluate whether catastrophic forgetting can be mitigated. To this end, HANS (McCoy et al., 2019), a related evaluation of challenging instances for natural language inference, will not be evaluated as the

Model	FEVER Dataset (Original Task) Accuracy (%)				
	Original	**Merged**	**FineTune**	**FT+L2**	**FT+EWC**
ESIM+GloVe	79.94 ± 0.4	79.57 ± 0.4	70.78 ± 1.1	$73.29 \pm 0.4^{*}$	$74.64 \pm 0.7^{*\dagger}$
ESIM+ELMo	80.15 ± 0.2	80.33 ± 0.8	76.45 ± 0.8	$73.72 \pm 0.6^{*}$	$78.09 \pm 0.4^{*\dagger}$
BERT Base	86.88 ± 0.5	86.87 ± 0.5	78.82 ± 0.9	$79.90 \pm 1.4^{*}$	$82.23 \pm 1.1^{*\dagger}$
RoBERTa Base	88.12 ± 0.3	88.11 ± 0.1	82.51 ± 1.5	$83.14 \pm 1.4^{*}$	$85.12 \pm 1.1^{*\dagger}$
	Symmetric Dataset (Fine-tuning Task) Accuracy (%)				
ESIM+GloVe	68.37 ± 1.0	69.35 ± 0.5	$74.21 \pm 1.3^{\#}$	$73.34 \pm 1.2^{\#\diamondsuit}$	$73.20 \pm 1.4^{\#\diamondsuit\heartsuit}$
ESIM+ELMo	64.04 ± 0.7	$66.46 \pm 1.3^{\#}$	$68.68 \pm 0.7^{\#}$	$70.31 \pm 0.5^{\#}$	$69.16 \pm 0.7^{\#}$
BERT Base	74.77 ± 1.4	$79.24 \pm 0.7^{\#}$	$87.07 \pm 0.6^{\#}$	$86.66 \pm 0.4^{\#\diamondsuit}$	$85.11 \pm 0.4^{\#}$
RoBERTa Base	78.34 ± 0.2	$87.03 \pm 2.3^{\#}$	$91.01 \pm 0.6^{\#}$	$90.98 \pm 0.5^{\#\diamondsuit}$	$89.63 \pm 1.3^{\#\diamondsuit\heartsuit}$

Table 5.2: Bias mitigation for FEVER classifiers comparing no treatment (original), against merging from instances from the FT-train with the original task training dataset (Merged) and FineTuning (with EWC and L2). Improvements $p < 0.05$ are marked with the following symbols: $*$ against FT, † against FT+L2, # against original. Deteriorations $p > 0.05$ on the symmetric dataset are marked with $\diamondsuit$ against FT and $\heartsuit$ against FT+L2

authors of this dataset found that high accuracies can be attained for these challenging instances when fine-tuning without observing catastrophic forgetting. Each stress-test dataset contains a small number of procedurally generated instances (between 1500-9800) that specifically target one of these phenomena. The evaluation will compare FT and FT+EWC using the same methodology as Liu et al. (2019a), controlling the number of instances sampled from FT-train (between 10-1000) and reporting the change in accuracy on the FT-test and original FT-test test sets.

5.5 Results

5.5.1 Fact verification

Fine-tuning the models, rather than merging datasets, yielded the greatest improvements in accuracy on FT-test. All improvements from the untreated model were significant ($p < 0.05$, denoted #). Without L2 or EWC, catastrophic forgetting occurs due to the shift in label distribution between the FEVER and FT-train dataset, which only contains two of the original three label classes.

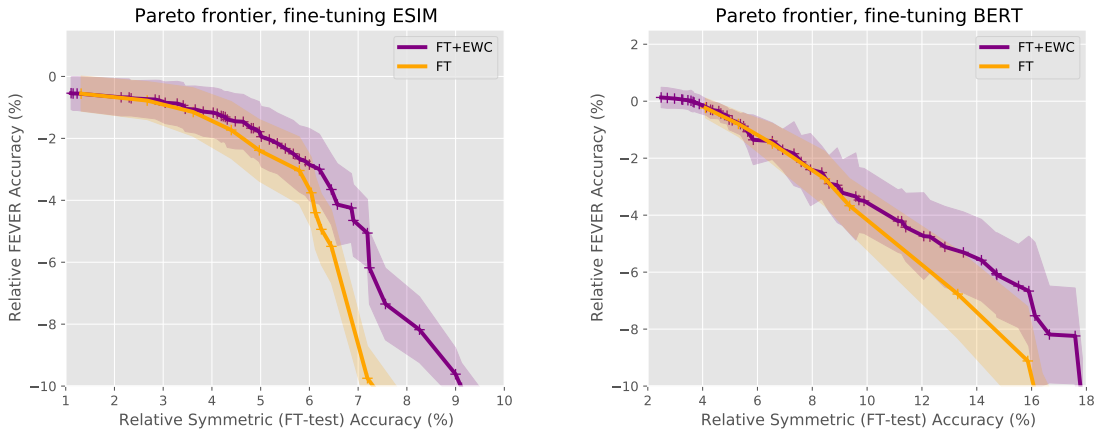


Figure 5.2: Pareto frontiers of fine-tuning ESIM and BERT on the symmetric data with and without EWC. Each point represents one hyper-parameter combination.

Both L2 and EWC reduced catastrophic forgetting. Improvements on the original task are significant ($p < 0.05$, denoted $*$) compared to FT. However, EWC regularisation retained more of the original task accuracy than L2 for all models; this was also significant (denoted †). In all cases, there is a trade-off between original and fine-tuning task accuracies and using Elastic Weight Consolidation Pareto dominates solutions without regularisation, illustrated for ESIM and BERT in Figure 5.2, which was plotted by training multiple models with the Cartesian product of all hyperparameter choices, listed in Appendix A.7.5. With regularisation, the FT-test accuracy was higher than FT+L2 and FT+EWC (except for ESIM+ELMo). However, the deterioration from the FT-test accuracy without regularising when using L2 and EWC was not significant ($p > 0.05$, denoted $\diamondsuit$). Furthermore, for RoBERTa, the highest performing model, the deterioration of using FT+EWC against FT+L2 was also not significant (denoted $\heartsuit$).

Training a model with a merged dataset of FEVER and FT-train yielded modest improvements on the FT-test without harming the original FEVER task accuracy. This can be attributed to the impact of these 700 instances being diluted by the large number of training instances in FEVER (FT-train is <1% the size of FEVER).

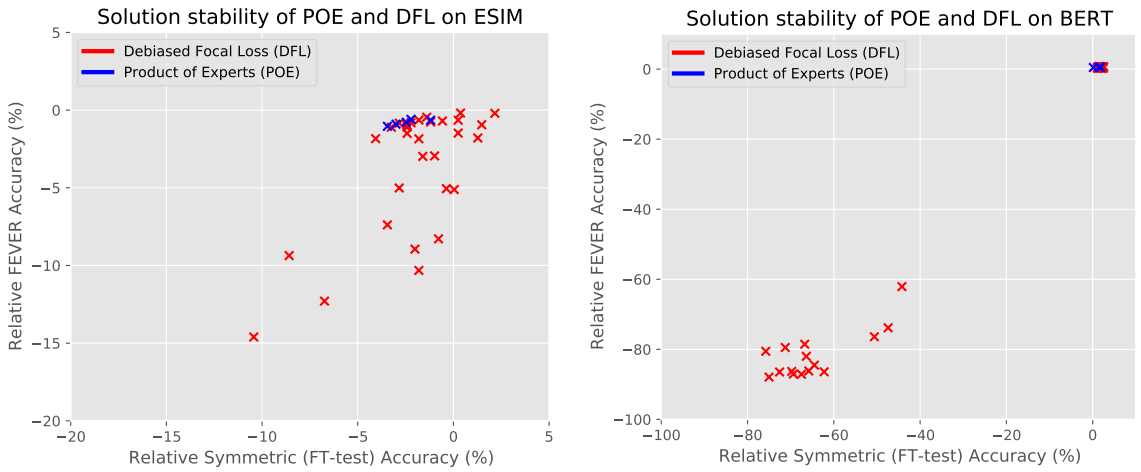


Figure 5.3: Solution stability of ESIM and BERT models trained with instance weighting using POE and DFL. Each point represents on hyper-parameter choice for β and γ.

Instance weighting

The Product of Experts and Debiased Focal Loss instance weighting methods were applied when training an ESIM and a BERT classifier. For both approaches, the rate at which the bias only model is trained may be modulated through hyperparameter β, which scales the cross-entropy loss of the bias branch model. The Debiased Focal Loss is also modulated through hyperparameter γ, which controls the strength of the bias branch model in the loss function for the text-pair model, as listed in Equation (5.4).

Figure 5.3 reports the relative change in accuracy on both the original FEVER task as well as the (unseen) FT-test set of symmetric counterfactual instances. Each point is the result of model training on FEVER, performing a grid-search over the β and γ hyperparameter choices offered in Mahabadi et al. (2020), listed in Appendix A.7.6. On the ESIM model (left), even though sweeping β parameter resulted in models that had stable accuracy on both tasks, no configuration of POE parameters offered improvements to the FT-test accuracy. In contrast, with DFL, sweeping the β and γ parameters caused large changes in accuracy, resulting in some solutions that improved the FT-test accuracy with negligible impacts on the original FEVER task. Similarly, for the BERT model (right), some solutions using DFL resulted in models that failed to converge when

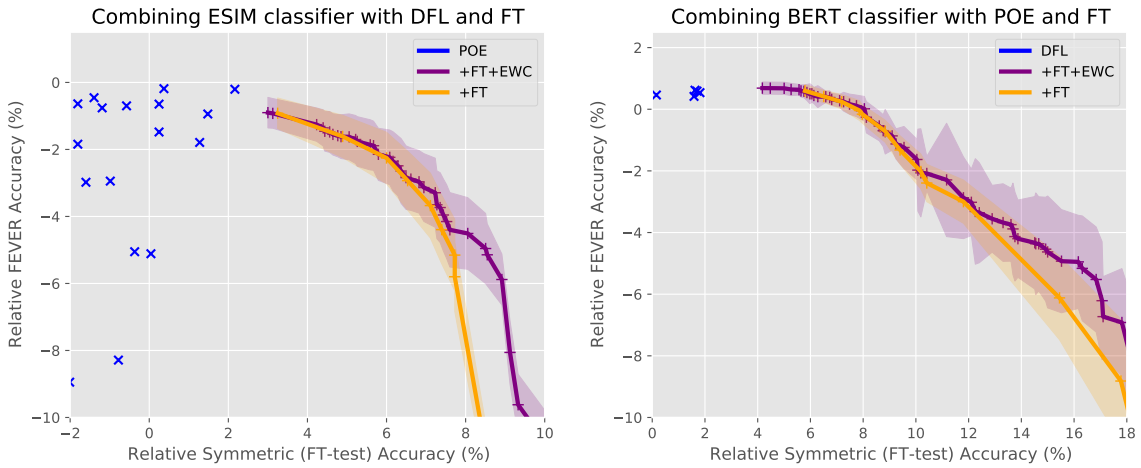


Figure 5.4: Pareto frontiers of fine-tuning ESIM and BERT with and without EWC regularisation, starting from a model trained with instance weighting.

sweeping the hyperparameters. In contrast, using POE in combination with the BERT model yielded negligible differences in accuracy to the DFL model for the best performing combinations of hyperparameters with much lower variance.

Combining FT and instance weighting

The trade-off between accuracy on the original and FT-test datasets is improved by combining both fine-tuning and instance weighting, visualised in Figure 5.4. The Pareto optimal model trained with POE or DFL is further fine-tuned with the symmetric data. For ESIM, DFL is used as this was the only approach that yielded improvements. For BERT, POE is used as this gave comparable improvements to DFL, but with much lower variance and one fewer hyperparameter. The Pareto frontier of fine-tuning is plotted using the same method as generating the plot from Figure 5.2. This further demonstrates that FT+EWC Pareto dominates FT for both ESIM and BERT, even when the starting point is a model trained with instance weighting. The combination of both techniques outperforms either of the bias modelling techniques in isolation.

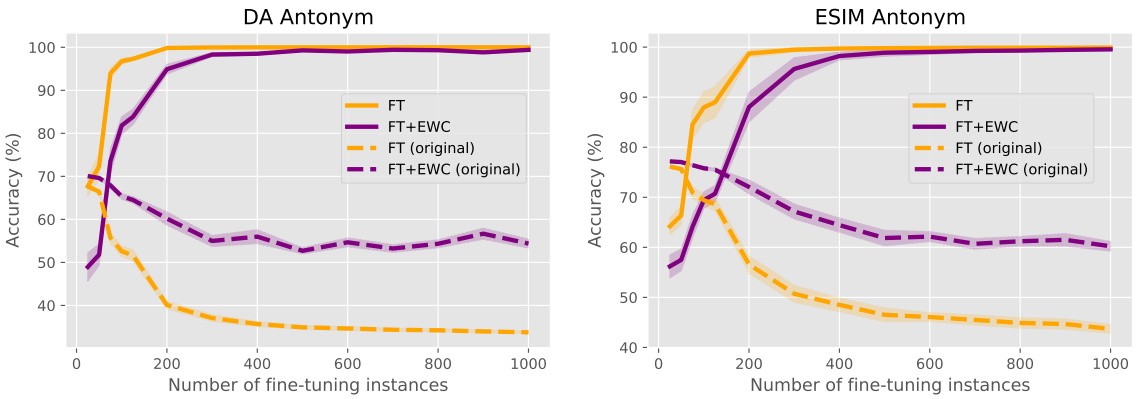


Figure 5.5: Fine-tuning a decomposable attention model trained MultiNLI models with two stress-test tasks from Naik et al. (2018)

5.5.2 Natural Language Inference (NLI)

In a separate experiment, EWC is applied to a different domain to inoculate biases in natural language inference using the stress test challenge datasets released by Naik et al. (2018). Liu et al. (2019a) observed that catastrophic forgetting was observed when fine-tuning using the *Antonym* challenge dataset. This was exacerbated by increasing the number of FT-training samples from 10 to 1000 which increased the accuracy for the stress-test task at the expense of reducing the accuracy on MultiNLI. Catastrophic forgetting was not observed by Liu et al. (2019a) on any other challenge data. However, to evaluate the impact of EWC, the same experiment is conducted on the *Numerical Reasoning* challenge for comparison.

Antonym challenge (Figure 5.5): Both the Decomposable Attention and ESIM model were sensitive to fine-tuning, attaining near-perfect accuracy on the FT-test data. The antonym stress-test only contains instances labelled *contradiction*: a change in label distribution that causes catastrophic forgetting. Replicating Liu et al. (2019a), without EWC, accuracy on MultiNLI fell to just above chance levels as the model learned only to predict contradiction (yellow dashed line). In contrast, by using an appropriate EWC regularisation penalty, the model still attains near-perfect accuracy with a smaller reduction in accuracy on the original MultiNLI task (purple dashed line).

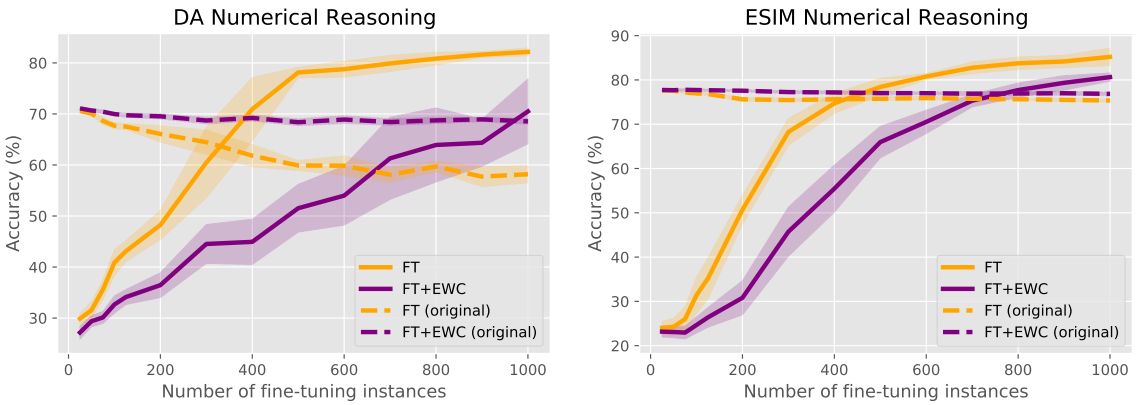


Figure 5.6: Fine-tuning an ESIM model trained MultiNLI models with two stress-test tasks from Naik et al. (2018)

Numerical reasoning challenge (Figure 5.6): Both the DA and ESIM models were sensitive to fine-tuning to introduce new numerical reasoning behaviours to the model. As the difference in label distribution in the inoculation dataset was less severe than the Antonym dataset, the reduction in accuracy on the original task was minimal for the ESIM model. FT+EWC minimised forgetting at the expense of reducing sample efficiency For the DA model, where catastrophic forgetting was observed, this was mitigated, again, through application of EWC.

5.6 Findings

Fine-tuning can mitigate model bias but has the risk that the model catastrophically forgets the data it was originally trained on. This was observed for both claim verification and natural language inference tasks, where the label distributions differed between the original task and the fine-tuning datasets. Incorporating elastic weight consolidation (EWC) when fine-tuning minimises catastrophic forgetting, yielding higher accuracy on the original task, and this holds for both the NLI stress-tests and debiasing fact-verification systems (Schuster et al., 2019).

Fine-tuning uses data to encode specific model biases and does not require alterations to a model architecture. In contrast, instance-weighting can also be used to target

biases; the bias for each instance needs to be quantified to scale the loss for instances during training. Schuster et al. (2019) generate these scalars through static analysis, and Mahabadi et al. (2020) generate these scalars by modifying the model architecture and co-training a bias branch model. While for some types of bias, such as hypothesis-only bias in text-pair classification, it is possible to quantify the degree of bias to weight instances, there are other types of bias, such as numerical reasoning or handling antonyms that are difficult to quantify in the same manner, and are more amenable to be targeted by fine-tuning. Where biases can be encoded by both fine-tuning and instance weighting and experimental results indicate that the combination of both techniques out-performs using either in isolation.

Chapter 6

Evidence-based factual error correction

This chapter presents an extension to the task of evidence-based claim verification where systems make potentially meaning-altering modifications to the claims so that they are better supported by evidence and was published under Thorne and Vlachos (2021b).

6.1 Introduction

This chapter proposes the task of *Factual Error Correction* as an explainable extension to claim verification. Rather than merely assigning a veracity label, possibly accompanied by evidence, the goal of error correction is to rewrite claims so that they are better supported by evidence. For example, in Figure 6.1, a claim that would be REFUTED by the evidence using a fact verification system is rewritten so that it becomes supported by evidence retrieved from Wikipedia. Similar to the FEVER task formulation, introduced in Chapter 3, the corrections are performed using evidence retrieved from Wikipedia.

With the potentially sensitive applications of predicting claim veracity, a number of recent works have focused on building explainable systems that augment the model output with a textual description of the reasoning process (Atanasova et al., 2020a,

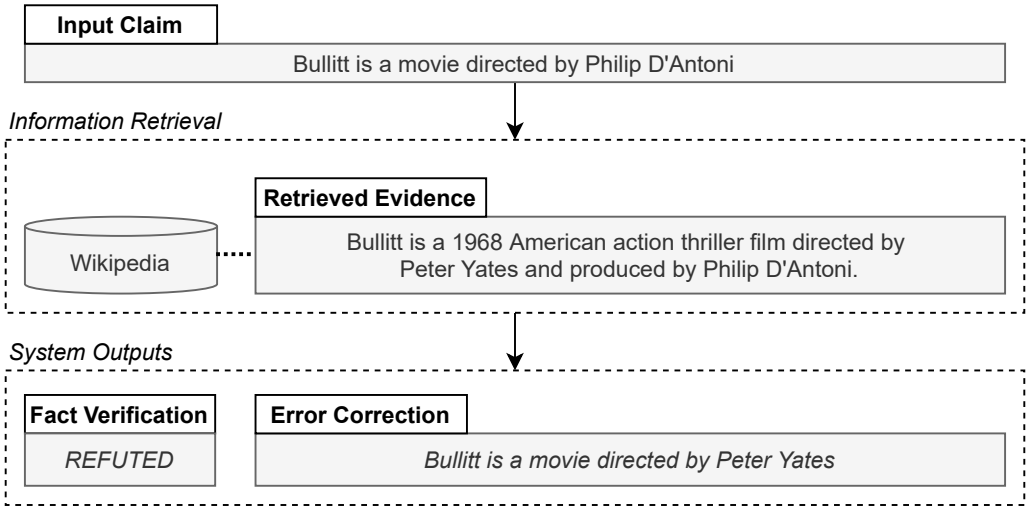


Figure 6.1: Factual Error Correction uses evidence to make corrections to claims, in contrast to fact verification, which instead classifies the veracity of the claim.

Stammbach and Ash, 2020, Kotonya and Toni, 2020). Exposing the evidence source and the decision-making process may help an end-user uncover subtle problems that cause automated systems to fail. Factual error correction acts as an additional mechanism to expose how a system predicts a veracity label. Furthermore, using retrieved evidence to continuously update news articles as facts change forms part of the vision outlined by Cohen et al. (2011) for automated newsrooms.

A challenge for factual error correction is the lack of datasets consisting of claims paired with reference corrections, which precludes training a model system with full supervision for the task and automated evaluation. However, with recent developments in fact-checking, there is an abundance of new datasets consisting of claims paired with evidence. To address this scarcity of reference corrections, this chapter makes use of distant supervision to incorporate retrieved evidence when generating corrections based on an approach for fact guided modification of Wikipedia articles (Shah et al., 2020). While Shah et al. (2020) assume that the claim and Wikipedia text are always incongruous and require a meaning-altering change, this chapter instead makes no assumptions over the veracity of the input given the evidence: it is applicable to claims that are supported or refuted by evidence. Additionally, following the same principles set out in FEVER

for retrieving evidence, the task proposal in this chapter further requires evidence to be retrieved from Wikipedia rather than requiring gold standard evidence to be explicitly provided as was the task model in Shah et al. (2020).

Although not specifically designed to constitute *corrections*, the intermediate annotations from the FEVER dataset construction, presented in Chapter 3, can be exploited for training and evaluating error correction systems. When constructing the FEVER dataset, sentences containing simple factoids from Wikipedia were first written by annotators before being rewritten with potentially meaning-altering mutations. These intermediate sentences may act as a reference for automated evaluation. One of the limitations, though, is that these intermediate sentences provide only one reference per claim. In practice, however, there may be many equally valid alternatives, resulting in artificially lower scores in automated evaluation, invalidating system comparisons. For example, in Figure 6.1, the claim could be corrected by adding a negation, replacing the entity name for the director, replacing the relation to 'produced by', or by replacing the entity name of the film to one that D'Antoni had directed. To capture these nuances in evaluation, systems will be evaluated and compared using human raters. Furthermore, the suitability of automated evaluation will be studied by reporting the correlation between the automated metrics and manual evaluation.

6.2 Related work

A number of related works offer methods to make corrections to sentences. However, their use of external information differs. The ways in which external information is incorporated can be placed on a continuum from only using the knowledge captured during language model pre-training to conditioning generation based on context. Key methods and approaches are briefly outlined:

Grammatical Error Correction (GEC) (Knight and Chander, 1994, Han et al., 2010, Ng et al., 2013, 2014, Yuan and Briscoe, 2016) is the task of making *meaning-preserving* changes to sentences so that grammatical errors are removed. No external information is

required as the sentence is undergoing a surface-level transformation where the (intended) semantic content should remain unchanged.

In contrast, the semantic content of sentences undergoing *factual* error correction will be altered, if needed, to better align the meaning with ground truth evidence. Shah et al. (2020) make meaning-altering updates to sentences in Wikipedia in a two-step process that does not require reference corrections in training: salient tokens are masked, and a corrector conditionally replaces the masks with ground truth evidence. In this approach, token salience is predicted by querying a model that is trained to perform fact verification for a claim against evidence. Cao et al. (2020) generate corrections as a post-editing step for outputs from abstractive summarisation so that they are consistent with the source text. Their approach uses a sequence-to-sequence model trained to restore artificially generated corruptions of a reference summary.

One potential source of knowledge is to use information stored in the parameters of large-scale pre-trained language models (Petroni et al., 2019). The language model can be used to recover tokens responsible for causing factual errors that are masked out as a variant of cloze-style evaluation (Taylor, 1953). While such approaches have been employed for fact verification (Lee et al., 2020), these approaches share the following limitations. Without explicit control (Nie et al., 2019a), the most likely decoded token may not be factually accurate or supported by the retrieved evidence, commonly referred to as a hallucination (Rohrbach et al., 2018, Zhou et al., 2021). Furthermore, *even if* the information stored within language model parameters could be reliably retrieved for factual error correction, facts change over time and the need to obtain information from up-to-date sources becomes greater as the state of the world diverges from the information captured within the model parameters. Recent language models augmented with a retrieval component such as REALM (Guu et al., 2020) and RAG (Lewis et al., 2020) could be applied. However, task-specific fine-tuning would still be required to condition the generation based on the factual error to mitigate hallucination.

6.3 Task definition

Let a claim c be the input sentence undergoing correction to yield the corrected claim c'. The correction requires incorporating knowledge from the retrieved evidence $E(c)$ such that c' is supported by this evidence, $E(c) \models c'$. The generated corrections must be intelligible, supported by evidence and correcting errors in the claim, as outlined:

R1 - intelligible Similar to other language generation tasks, the first requirement is that generated outputs are fluent and intelligible. They must be free of grammatical mistakes, and the meaning must be understandable without the aid of additional context or evidence so that their factual correctness can be assessed.

R2 - supported by evidence The generated correction must be supported by the retrieved evidence. This property follows from previous work on fact verification, and also requires models to condition generation on the retrieved evidence – penalising models that hallucinate (Holtzman et al., 2020).

R3 - error correction Specific to factual error correction, the corrections made by the system should be targeted to the errors present in the inputted claim. While this, in part, can be assessed by R2, the correction must be compared to the inputted claim to ensure the output is not introducing new unrelated information. For example, an erroneous claim: *France is in North America* could be supported by evidence if it were rewritten as *France is a republic*. However, the desired correction should remain on the topic of France's geography, such as *France is in Europe*.

6.4 Task decomposition

The choice of supervision for the factual error correction system influences the task decomposition. For example, with full supervision, the system can be constructed with an information retrieval module and a sequence-to-sequence module that conditionally

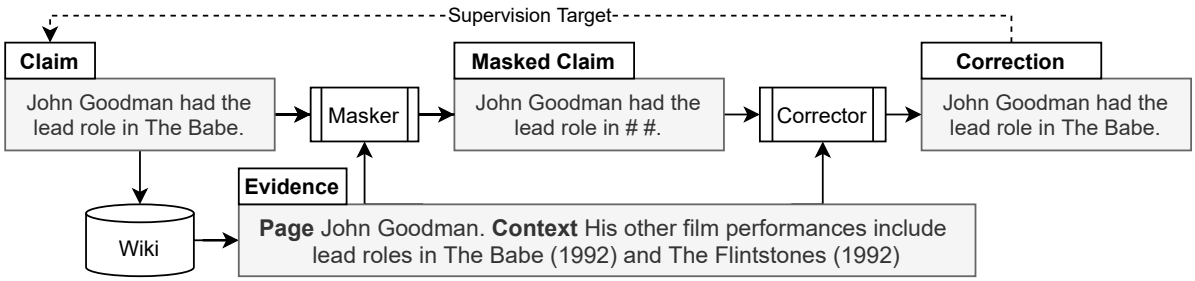


Figure 6.2: The corrector is trained to reconstruct masked claims, conditioned on retrieved evidence, indicated by the dashed arrow. At test time, the corrector is able to incorporate new facts from the evidence to generate corrections.

generates a correction given the claim and evidence. However, large datasets of claims paired with corrections are not available. Fact *verification* datasets, such as FEVER, ClimateFEVER and SciFact, are now an abundant resource and contain claims labelled with evidence but do not contain corrections. The absence of full supervision mandates the use of distant supervision when training systems, this chapter proposes a task decomposition that generates corrections by training models to reconstruct claims with masked tokens using retrieved evidence.

6.4.1 Distantly-supervised corrections

Test time Corrections are generated by a two-stage process, illustrated in Figure 6.2. Tokens from the claim, c, are first masked, yielding $\tilde{c}$, and then inputted to the corrector to return a correction $c' = Corr(\tilde{c}, E(c))$. The masker, $\tilde{c} = Mask(c, E(c))$, replaces a subset of tokens in the claim with a blank placeholder, conditioned on evidence $E(c)$. Its purpose is to remove tokens that are salient to the claim being supported or refuted by the evidence. Using the masked claim, $\tilde{c}$, the corrector replaces the blank placeholders with tokens conditionally generated using the retrieved evidence. To correct errors, evidence refuting a claim ($E(c) \nvDash c$) is used to condition the generation of a correction supported by it $E(c) \vDash c'$. This approach extends the protocol of Shah et al. (2020) by using multiple retrieved evidence sentences to correct claims, rather than using a single gold factoid to update a Wikipedia article.

Training the corrector Similar to masked language modelling, the training objective is to recover the input claim $c' = c = Corr(\tilde{c}, E(c))$ after masking $\tilde{c} = Mask(c, E(c))$. However, c' is generated conditioned on evidence $E(c)$. By training the model to regenerate the input claim, it is expected that the model will regenerate the input claim only if it was in complete agreement with the evidence (assuming the masking is correct) by incorporating information from the evidence. Otherwise, when the evidence is in disagreement with the claim, the generated correction will contain evidence correcting the masked claim, which enables systems to generate corrections satisfying requirements R2 (supported by evidence) and R3 (correcting the error), assuming correct masking.

Masker When applied to factual error correction, masking salient tokens from the claim acts as a proxy to which tokens need to be removed to correct an error. Parallels can be drawn between masking and generating token-level explanations (Camburu et al., 2018, Thorne et al., 2019a, Wiegreffe and Pinter, 2019). Common approaches to generating explanations are summarised in Section 6.5.2.

6.5 Model

6.5.1 Evidence retrieval

For evidence retrieval, GENRE (De Cao et al., 2020) and Dense Passage Retrieval (Karpukhin et al., 2020) are used as both exhibited high recall in Chapter 3 and have shown high recall for a number of language understanding tasks over Wikipedia (Petroni et al., 2021). At test-time, the claim is encoded using DPR, and the most similar passages from Wikipedia are returned using an inner-product search. The top-k passages returned by DPR are filtered, keeping only the passages from pages predicted by GENRE.

6.5.2 Token-level explanations as masks

At test time, the purpose of the masker is to selectively remove tokens that contribute to the factual errors within a claim. Different choices of maskers at training and test

time influence the quality of the corrections generated by the downstream corrector. The maskers evaluated in this chapter consider varying levels of access to model information and different run-time complexities. Both the black- and white-box methods, outlined below, require querying a veracity classifier, whereas the language model masker and heuristic string matching baseline do not.

Black-box masker This masker works by perturbing the input to a classifier that is trained to predict the veracity of a claim given evidence to highlight which tokens from the claim are important to the veracity classifier. Concretely, the explanations are generated using LIME (Ribeiro et al., 2016), a model diagnostic that trains a locally linear model to score the importance of input features (specifically, tokens in the claim) with respect to the predicted labels for every claim. The model under test is a BERT classifier where the evidence and the claim are concatenated in the input (specifically, the model trained with retrieved evidence negatively sampled for NEI claims from Section 3.5.5). This method is referred to as *black-box* because the model does not undergo modification, and no information about internal values or states is exposed when generating the explanation, which is used to mask the claim.

White-box masker In contrast, to obtain *white-box* model explanations, the model has undergone modification to expose internal information that aids in predicting whether a token should be masked or not. This chapter will evaluate the Neutrality Masker from Shah et al. (2020), which predicts which tokens are likely to cause a label flip from SUPPORTED or REFUTED to NOTENOUGHINFO when masked. This masker exposes the encoded input of an ESIM classifier (Chen et al., 2017b) and adds a linear classifier over the hidden states to predict per-token masking probability. At test time, masks can be generated through a single query to the model (unlike LIME in the black-box masker, which requires multiple queries to the model), but requires an additional step to train, using predictions from the classifier as a training signal.

Language model masker To verify whether it is possible to generate masks without the need for training a fact verification model, this approach uses a BERT pre-trained language model (Devlin et al., 2019) to measure the surprisal of tokens in the claim without task-specific fine-tuning. The intuition of this approach is to identify tokens that introduce misinformation, under the hypothesis that the world knowledge (Petroni et al., 2019) captured in pre-training would assign lower probabilities to tokens contradictory to the world state expressed in the text used to train the language model.

Baselines This considers two further simple baseline maskers that do not require a model: *random* masking of a subset of tokens and a *heuristic* method of masking tokens which are not present in both the claim and the retrieved evidence.

6.5.3 Corrections

Corrections are generated using an encoder-decoder transformer model that is trained to generate corrections from masked claims and evidence. The model is a pre-trained T5 transformer (Raffel et al., 2020) initialised with `T5-base` and is then fine-tuned with the distant supervision protocol described in Section 6.4.1. The masked claim and evidence are jointly encoded by concatenating these two texts in the input. This is compared against a baseline model from the related task of fact guided sentence modification (Shah et al., 2020) which uses the pointer generator network implementation from See et al. (2017). Unlike the transformer, which captures long-range dependencies between claim and evidence through self-attention (Vaswani et al., 2017), this baseline independently encodes the evidence and masked claim using LSTMs (Hochreiter and Schmidhuber, 1997) before decoding using a pointer-generator network (Vinyals et al., 2015).

In order to evaluate the impact of conditioning on evidence, this chapter compares using a language model without fine-tuning or conditioning to decode masked tokens, similar to the Language Models as Knowledge Bases approach introduced by Petroni et al. (2019). This would consider correcting claims using the implicit knowledge stored within the model parameters rather than using external evidence.

6.6 Data

Systems will be trained and evaluated using the FEVER dataset introduced in Chapter 3. To comprehensively evaluate the generated corrections, manual evaluation is required. However, this is expensive and not suitable for system development and hyperparameter optimisation. While FEVER, and other fact verification datasets do not contain reference corrections, the claim mutation process from the FEVER dataset construction, documented in Section 3.2 provides a *silver* standard which could be used for automated evaluation and for training a supervised system, indicating an upper-bound that the distantly-supervised systems can be compared against. The initially extracted *unmodified* facts will serve as the reference correction for the *mutated* claims.

The class balance and size of this corrections dataset is reported in Table 6.1. The training and test splits are disjoint by the entity used to generate the claim. The additional hidden shared task test set was not used. Furthermore, the claims labelled as NotEnoughInfo will not be used to train or test the error correction systems in this chapter as there is no labelled evidence from which to make corrections. The unused NotEnoughInfo instances without evidence total 25841 pairs of claims and corrections (21934 training, 1870 development and 2037 test).

Label	**Instance Count**		
	Train	**Development**	**Test**
Supports	37961	1477	1593
Refutes	20075	2091	2289
Total	58036	3568	3891

Table 6.1: Instance counts by class and dataset partitions

6.7 Evaluation

While it's convenient to use an automatic metric during development, metrics that compute token overlap against a single reference sentence cannot capture the nuances

required to assess the veracity of the generated corrections against evidence. Thus, the primary evaluation of systems in this chapter will be a manual assessment of whether the generated corrections meet the task requirements. Human raters are asked three questions about system outputs to assess whether the corrections meet the requirements of intelligibility, being supported by evidence, and correcting the error outlined in Section 6.3. The raters were four members of the lab who were familiar with fact verification and the FEVER dataset but had no prior exposure to the error correction task. Responses were calibrated with a pilot study using instances sampled from the development set.

For the first two requirements, the annotators were asked two binary questions, first labelling whether the claim was grammatical and intelligible, and then subsequently labelling whether the claim was supported by the retrieved evidence. For the third requirement of the correction being related to the error, the question shown to the annotator has three answer choices: (1) the information content with respect to the evidence improved, (2) information unrelated to the claim was added (i.e. the claim was ignored), (3) no correction was needed (i.e. the claim was already supported by evidence). The raters were shown each question in this sequence without knowledge of which system generated the correction. Negative answers to any question automatically assigned negative answers to subsequent ones (prescribing that an unintelligible sentence could not contain a fact supported by evidence or introduce a correction). 20% of the annotation tasks are assigned to two raters to measure inter-annotator agreement.

Popular automated evaluation measures (SARI, BLEU and ROUGE) are compared to manual evaluation using Pearson correlation. SARI (Xu et al., 2016) is a metric used for sentence simplification and considers n-grams retained from the source as well added or deleted n-grams through comparison against a reference sentence. BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) indicate precision and recall, respectively, of n-grams between the claim and correction.

6.8 Implementation

T5 Corrector The corrector model is initialised with the pre-trained `T5-base` checkpoint released by HuggingFace (Wolf et al., 2020). The learning rate was selected by optimising the overall SARI score on instances from the development split of the FEVER dataset. The search space for learning rate was $\{10^{-5}, 5 \cdot 10^{-5}, 10^{4}, 5 \cdot 10^{-4}\}$ and a maximum of 4 epochs. The optimal learning rate of $5 \cdot 10^{-5}$ was selected, and the weights from the epoch with the maximum SARI score were used at test time.

Fully supervised ceiling To estimate the ceiling performance of a factual error correction system assuming a reasonable amount of training data is available, an encoder-decoder model is trained with full supervision that other methods can be compared against. A T5-base model, initialised from the same `T5-base` checkpoint, is fine-tuned using the intermediate data from FEVER to supervise using the same hyperparameter choices as the T5 Corrector.

Evidence retrieval Both the DPR (Karpukhin et al., 2020) and GENRE (De Cao et al., 2020) implementations use the code and weights released by the respective authors. For DPR, an index of the FEVER Wikipedia pages was constructed, chunked into passages of 50 tokens. Selecting the top two matching passages resulted in the highest scores on the downstream corrections; SARI was lower when using 1 or 3 passages.

Maskers For the white-box masker, the implementation provided by Shah et al. (2020) was trained, retaining the hyperparameters choices from the authors. For the black-box masker, the LIME implementation from Ribeiro et al. (2016) was used to probe the BERT classifier trained with negative instances for the NEI class sampled from retrieved evidence from Section 3.5.5. For the language model and random baseline maskers, where the number of masks was tunable, 50% of the tokens were masked, similar to the proportion of tokens masked by the black- and white-box maskers.

System	Evidence	Training Masks	Test Masks	Aggregated Score (%)		
				Intel.	**Supp.**	**Corr.**
T5 Supervised	Gold	-	-	98.9	88.9	68.9
T5 Supervised	Retrieved	-	-	97.7	64.7	48.9
Mask + T5 Corr	Retrieved	Random	Heuristic	89.3	**57.9**	**40.0**
Mask + T5 Corr	Retrieved	Heuristic	Heuristic	90.0	38.0	20.0
Mask + T5 Corr	Retrieved	Random	Black-box	**93.1**	42.2	24.0
Mask + T5 Corr	Retrieved	Black-box	Black-box	91.4	37.0	19.8
Mask + T5 Corr	Retrieved	White-box	White-box	90.6	41.7	23.9
BERT LM	-	-	Heuristic	48.0	20.7	15.0
BERT LM	-	-	Black-box	30.1	4.9	3.4
Shah et al. (2020)	Gold	White-box	White-box	32.2	10.7	5.0

Table 6.2: Aggregated scores from human evaluation considering intelligibility, whether generated instances were supported by evidence and errors corrected.

Language model corrector Tokens are greedily decoded using a `bert-base-cased` masked language model using the HuggingFace implementation. In contrast to Petroni et al. (2019), who only decode single masks, greedy decoding allows multiple masks to be replaced with tokens. Using a `t5-base` auto-regressive language model without fine-tuning resulted in entirely nonsense outputs, and could not be evaluated.

Comparison to previous work To compare against previous work for fact guided sentence modification, the dual-encoder pointer network implementation from Shah et al. (2020) is trained on the error correction dataset. The original hyperparameter choices for the network were initially used and did not result satisfactory corrections. Changing these within ranges suggested by the authors yielded no improvement.

6.9 Results

The aggregated results of the manual evaluation, assessing the requirements that corrections are intelligible, supported by evidence, and improve the factuality of the claim, as listed in Section 6.3, are reported in Table 6.2. The evaluation considers a random sample

of 200 instances per system. To measure inter-annotator agreement, 20% of instances were annotated by two annotators: the Cohen's κ (Cohen, 1960) coefficients for the three questions are 0.92, 0.92 and 0.86, respectively, indicating strong agreement.

The fully supervised models had the highest rate of satisfactory corrections that improved the factuality of the claim (requirement 3), indicating a performance ceiling for the distantly supervised models. Incorporating retrieved evidence in these supervised models (rather than gold) reduced the number of corrections supported by evidence from 88.9% to 64.7% and the number of satisfactory corrections from 68.9% to 48.9%. This highlights the challenges of incorporating the retrieved evidence (which is possibly noisy) when generating the corrections.

When using retrieved evidence, the white-box masker generated no masks for 41% of instances. Without any masked tokens in the claim, the T5 corrector copied the input directly to the output which fits the assumption that, for claims already supported by evidence, no correction is required. However, this reduced the number of corrections supported by evidence as errors were not always masked out.

Using a heuristic masker at test time, which removed tokens from the claim not present in the evidence, generated more claims meeting the supported and corrected requirements than masks generated by querying a fact verification model (both black-box and white-box). An analysis of the masker's influence on the corrections is provided in Section 6.9.1. When using the masker and corrector distant supervision strategy, different maskers could be used to train the corrector from the masker used at test time. Training the corrector with random masks yielded both a higher rate of satisfactory corrections and corrections supported by evidence when using either the black-box or heuristic masker at test time. Further evaluation of training with random masks is provided in Section 6.9.2.

The two baseline systems, Dual Encoder Masker and Corrector, based on Shah et al. (2020), and a pre-trained BERT language model, generated corrections that were intelligible or supported by evidence at a lower rate than the aforementioned models. These are further discussed in Sections 6.9.3 and 6.9.4, respectively.

Metric	Correlation (Pearson r)		
	Intelligible	**Supported**	**Corrected**
SARI Keep	**.87**	**.95**	**.93**
SARI Final	.78	.92	.91
SARI Delete	.72	.82	.91
SARI Add	.52	.84	.79
ROUGE2	.75	.90	.91
ROUGE1	.71	.87	.88
BLEU2	−.05	.32	.45
BLEU1	−.46	−.10	.05

Table 6.3: Both SARI and ROUGE automated scoring metrics have high correlation to manual evaluation.

The correlation between automated scoring metrics and the manual evaluation is reported in Table 6.3. The KEEP component of SARI, which measures the F1 of n-grams from the claim retained in the output, had the highest correlation with all three requirements. Overly aggressive maskers which remove too much content from the claim can result in unintelligible outputs or corrections unrelated to the claim. ROUGE2, which measures the recall of bigrams in the correction with respect to the reference, exhibited a reasonable correlation to the manual evaluation against the supported and corrected requirements. However it does not correlate as well with intelligibility. The ADD and DELETE components of SARI provide further information but do not correlate as strongly with human judgements. Having only one reference correction reduces the utility of precision-oriented metrics, like BLEU, as valid corrections can differ from the reference.

6.9.1 Choice of masker

When training the corrector with the same masker that is used at test time, both the heuristic and black-box maskers yielded comparable scores under human evaluation. Inspection of SARI breakdown in Table 6.4 indicates that more tokens were kept when using the heuristic masker (Keep=.651). In contrast, the black-box model was more aggressive in masking, resulting in less information from the claim being retained (Keep=.594). This measure correlated well with human judgements as more information retained gives

Masker	SARI Score			
	Keep	**Delete**	**Add**	**Final**
Black-box (Gold)	.630	.582	.088	.433
White-box (Gold)	.652	.559	.128	.447
Black-box (IR)	.594	.526	.090	.412
White-box (IR)	.628	.535	.107	.426
Heuristic (IR)	.651	.574	.041	.422
Masked LM	.538	.509	.062	.370
Random	.619	.475	.087	.390

Table 6.4: Extrinsic evaluation of maskers, varying the use of evidence when generating the masks, evaluated using the Masker + T5 Corrector system.

a richer context for generating the correction and prevents erasure of claims already (partially) supported by the evidence.

Both the black-box (LIME) and white-box (the masker from Shah et al. (2020)) methods require querying a veracity classifier to generate the masks. Using retrieved evidence for the veracity classifier had a negative impact on most components of the SARI score. For the black-box masker, using retrieved evidence reduced the number of masked tokens from an average of 4.7 per claim to 3.9. In comparison, the number of tokens masked by the white-box masker remained unchanged at 4.7 (approximately 50% of the number of tokens in the claim). Most notably, the white-box method of mask generation (row 4 in Table 6.4) did not generate masks for 41% of instances when using retrieved evidence, whereas all instances had at least one mask when using gold evidence — an artefact of the noise introduced by retrieval.

6.9.2 Corrector trained with random masks

Generating large quantities of masked training data through querying a model, such as with the black-box model explanation techniques, can be computationally expensive.[1] In contrast, random masks can be generated without querying a model, requiring fewer

[1]Generating black-box explanations over the training data with LIME took approximately one day using a server with a single Nvidia P100 GPU

Masker	SARI Score			
	Keep	**Delete**	**Add**	**Final**
Black-box (Gold)	.618	.622	.102	.447
White-box (Gold)	.640	.570	.114	.441
Black-box (IR)	.611	.543	.194	.419
White-box (IR)	.618	.590	.144	.452
Heuristic (IR)	.652	.627	.155	.478
Masked LM	.561	.529	.078	.389

Table 6.5: Using random masks at training resulted in higher scores when testing with different maskers

resources by several orders of magnitude. Furthermore, using a corrector trained on random masks resulted in higher quality outputs at test time when paired with the black-box and heuristic maskers, as training with random masks promoted good exploration of the task. In contrast, while the black-box and heuristic approaches worked well during testing, correctors *trained* on these maskers generated worse outputs due to the limited exploration of the task space. Additionally, generating training data using the black- and white-box methods requires making predictions using the model's training data which may result in different outcomes to making predictions on unseen test data.

6.9.3 Comparison to previous work

Previous work for fact-guided sentence modification of Wikipedia articles uses a dual encoder pointer network (Shah et al., 2020) to make corrections, reported in Table 6.6. The corrector tended to copy portions of the claim rather than correct it, resulting in a SARI Keep score of .452, which is lower than the T5 model using the same white-box masker (Table 6.4). Human evaluation considered these corrections mostly unintelligible, even when using gold evidence (Table 6.2). This was especially the case for rarer entities; hyperparameter tuning of the corrector's coverage ratio, as suggested in correspondence with the authors, did not yield improvements.

System	SARI Score			
	Keep	Delete	Add	Final
Dual Enc Pointer (Gold)	.452	.569	.039	.353
Dual Enc Pointer (IR)	.345	.481	.017	.281

Table 6.6: Results using a dual encoder pointer network (Shah et al., 2020) were low, despite the strong masker.

Masker	SARI Score			
	Keep	Delete	Add	Final
Masked LM	.360	.472	.019	.289
Heuristic (IR)	.629	.651	.034	.438
White-box (IR)	.232	.446	.005	.228
Black-box (IR)	.364	.003	.001	.122

Table 6.7: Correcting claims using a language model does not condition the generation on evidence.

6.9.4 Language models as correctors?

With the exception of the heuristic masker, using a pre-trained language model without fine-tuning to correct claims resulted in low SARI scores (Table 6.7). Without conditioning on the evidence, the correction is not related to the claim or supported by evidence to verify the claim, which is indicated by the low SARI Add scores, which consider the precision of the added tokens. As these maskers deleted most tokens, retaining only stop-words, decoding most likely tokens without a prompt or context tokens resulted in unintelligible outputs. For the heuristic masker, more content words were retained, yielding more intelligible outputs. However, even though the BERT language model was pre-trained on a corpus that included pages from the English language Wikipedia (Devlin et al., 2019), the generated corrections were not always supported by evidence, indicated in the human evaluation in Table 6.2.

6.10 Discussion

Beyond simply identifying errors, factual error correction adds an additional mechanism of communicating claim veracity, making the decision process of a fact verification system more transparent as poor corrections generated by a model expose limitations that would otherwise be hidden by classification. Factual error correction presents a number of challenges that are shared with related domains, including information retrieval, fact verification and abstractive summarisation. This chapter demonstrated that the task could be performed with distant supervision, meaning that a new mechanism for communicating the veracity of claims can be generated using existing datasets such as FEVER. This is in contrast to other explanation mechanisms for fact verification which require additional supervision for generating explanations (Camburu et al., 2018, Kotonya and Toni, 2020, Atanasova et al., 2020a). Despite this chapter demonstrating the feasibility of the task, there are a number of outstanding challenges, including how evidence is incorporated into the correction and how corrections are scored.

For evaluation, intermediate data from the FEVER task was re-purposed as reference corrections for automated scoring. While this had a high correlation with SARI and, to a lesser extent, ROUGE, this evaluation is only a proxy for the underlying task, evaluating whether systems can undo mutations introduced by human annotators. Furthermore, it is limited by the fact that there is only a single reference correction and the fact that there may be multiple equally valid corrections one can make for an erroneous claim, reducing the utility of these token-based metrics. Future work should consider how automated scoring can be better used to assess the quality of the generated corrections.

Chapter 7

Conclusions

7.1 Summary

This thesis considers the task of predicting the veracity of claims, an emerging task that has garnered significant attention in mainstream media due to the rise of misinformation being shared on social media. The case for systems to retrieve and use evidence, argued in Chapter 2, influenced the creation of the FEVER dataset, presented in Chapter 3. The novelty of FEVER is the requirement for systems to retrieve evidence as part of the reasoning process, allowing systems to reason over veracity for unseen topics. This task is evaluated by considering what evidence the systems retrieve, and whether the systems correctly predict the claim's veracity.

Given the potentially sensitive applications of fact verification systems, a critical consideration is their resilience to previously unobserved patterns in claims. Adversarial evaluation can expose model blind spots. However, since the methods for generating adversarial instances are typically automated, any evaluation must consider whether the adversaries are truly exposing model limitations or have inadvertently changed the meaning of the claim without considering the whether the label needs updating, an issue highlighted in Section 4.2. The two new metrics introduced in Chapter 4, attack

potency and system resilience, consider instance correctness and allow for fairer system comparisons. The most effective attacks under this scoring approach were the simplest; while manually constructed rules induced a modest amount of errors, the instances were more often correct than the automated methods, resulting in higher potency.

The instances constructed for the FEVER dataset exhibit a bias similar to the hypothesis-only bias present in the related task of natural language inference, which also used annotators to write sentences for their instances. Chapter 5 considers how biases in text-pair classification models can be mitigated in FEVER and the related MultiNLI dataset. The experimental findings in Chapter 5 demonstrate that models can be debiased by fine-tuning with instances that specifically target the model limitations. Furthermore, regularising the fine-tuning with Elastic Weight Consolidation (EWC) yields Pareto optimal models. While regularisers like EWC are used in domain adaptation tasks in machine translation, this thesis is the first to consider fine-tuning with EWC regularisation for debiasing text-pair classifiers.

One of the key features that separates FEVER from previous fact-checking datasets is the use of the retrieved evidence as a mechanism to communicate to an end-user how a veracity prediction was reached. Chapter 6 presents the final contribution of the thesis: an extension to fact-checking where the veracity of claims is communicated through the generation of factual error corrections, making the system's decision-making process more transparent than returning a label alone, argued in Section 2.2.5. These corrections are generated through a distant-supervision strategy, using retrieved evidence, and do not require additional training data.

7.2 Impact

Not only has FEVER and the fact verification methods presented in this thesis influenced this emerging discipline within NLP, but their influence also extends to other NLP tasks such as information retrieval, open-domain machine reasoning, and natural language inference. Since the completion of the FEVER shared task (Thorne et al., 2018b),

the leaderboard[1] has received submissions from 114 teams; many of these submissions achieved substantial improvements in evidence recall, label accuracy and FEVER Score compared to the published baseline (Thorne et al., 2018a). Several new architectures for text classification have been developed for and evaluated on FEVER, such as using graph attention to aggregate reasoning over multiple evidence sentences (Zhou et al., 2019a, Liu et al., 2020). In the second FEVER shared task (Thorne et al., 2019c), four participating teams developed a range of adversarial attacks, and subsequent methods have considered the requirement for the adversarial instances to be correctly labelled and well-formed (Atanasova et al., 2020b).

The evidence-based reasoning approach to fact verification have been further studied in numerous task settings. The NLP community has built derivative datasets for other domains and languages, including scientific claims (and claims relating to COVID-19) (Wadden et al., 2020), climate change information (Diggelmann et al., 2020), and Danish-language claims (Nørregaard and Derczynski, 2021). Evidence-based verification has further been evaluated in applications in the newsroom, where Miranda et al. (2019) used models trained on FEVER to verify political claims. Furthermore, FEVER has been used to build Wikipedia-based fact-checking APIs (Chernyavskiy et al., 2021), including part of the Wikimedia Foundation's tooling (Trokhymovych and Saez-Trumper, 2021).

FEVER has been incorporated into the Knowledge Intensive Language Tasks dataset (Petroni et al., 2021, KILT), a larger collection of benchmarks for knowledge-intensive NLP tasks, such as question answering and slot filling, that requires information to be retrieved from Wikipedia by the model in order to generate the system output. FEVER's inclusion in this benchmark, and the experimental findings that all tasks can be modelled with a unified architecture, demonstrates that the challenges present in evidence-based fact verification are common to these other NLP tasks, and that the data is valuable to researchers working on them.

[1] `https://competitions.codalab.org/competitions/18814`

7.3 Limitations

It is important to acknowledge that the complexity of the fact-checking conducted by journalists is, for the moment, beyond the abilities of the systems that can be developed due to the complex reasoning, assumptions and background knowledge. While this complexity is a target that can help stimulate progress in NLP and related fields, it should also calibrate our expectations and promises to society. The claims in FEVER are relatively simple, pertaining to a single property of an entity, and may not represent all the ways in which misinformation can be introduced in the real world. In contrast to previous work, where political claims are scraped from fact-checking agencies, the claims in FEVER were manually constructed to build and evaluate NLP models. This design choice balances the challenges in finding and annotating evidence against the claims being unambiguous and simple enough to reason over their veracity. However, FEVER may not be representative of the types of facts that end-users would wish to verify; this misinformation about topics that are under-represented in FEVER may not be accurately verified or corrected. While FEVER demonstrates the feasibility of computational fact verification, application to political or scientific domains would require additional data, investigations into what biases exist, and likely require the introduction and evaluation of new methods.

The type of reasoning performed in FEVER also assumes that trustworthy, unbiased and non-contradictory bodies of textual evidence are present and sufficient for predicting veracity. This limits the ability of systems only to perform simple inferences rather than the broad range of inferences that fact-checkers would perform. Fact-checkers use a wider range of evidence materials from documents that vary in purpose, complexity and format, and must reason about which sources are appropriate and deal with societal context, ambiguity, and incomplete information. While the decision to use the introductory sections of popular Wikipedia articles enabled FEVER's annotators to rapidly and accurately identify appropriate evidence, systems trained on the dataset may be limited in the reasoning that they can perform. In contrast to other types of documents, these

introductory sections are information-dense. Another aspect of fact-checking, reasoning with numerical information, such as counting, summation and understanding trends, cannot be modelled using textual evidence alone. Since neural networks for NLP typically model numerical information in the same way that text is modelled, treating numbers as tokens, their ability to reason about numerical properties and trends is limited (Andor et al., 2019). Even with explicit modelling to capture large numbers of numerical facts, such as question answering using Neural Databases (Thorne et al., 2021b), statements must be unambiguous to filter appropriate evidence and select the correct arithmetic operation. However, the ambiguity in how trends and numerical information are reported and discussed (for example, 'The number of deaths from cancer has increased' or 'Jamaica has won the most golds for running') makes interpretation difficult even for human fact-checkers, let alone automated systems.

The final limitation of FEVER is the assumption of verifiability of the claims. While claims were labelled as NotEnoughInfo when information to support or refute them could not be found in Wikipedia, the assumption is that with sufficient information, a supported or refuted label could eventually be assigned. As unverifiable and ambiguous claims were excluded from the FEVER dataset by the annotators, there is no mechanism for models trained FEVER to identify whether a claim is unsuitable or cannot be verified, limiting the application of models on real-world claims. This caveat can be mitigated, however, by systems retrieving and presenting a limited subset of the evidence to an end-user, which acts as a mechanism to communicate the background information and circumstances surrounding a claim without necessarily prescribing veracity.

7.4 Future work

This thesis demonstrated the feasibility of verifying and correcting general-domain textual claims in English, using evidence from introductory sections of Wikipedia pages. While this has helped stimulate research in automating fact-checking, there are many extensions to FEVER that need to be explored in future work to build tools that support journalist fact-checkers and lay-users.

User studies The models in this thesis are evaluated in isolation, considering the accuracy of the veracity prediction and recall of evidence; this thesis has not considered how a veracity classifier can be integrated into an online platform, nor how users interact with such a system in real-world applications. While FEVER has supported the development and evaluation of new NLP models, future work should consider how such models can be used to augment end-users' ability to find and reason with information. Furthermore, evidence-based verification should be evaluated in combination with other task formulations, such as claim matching (Hassan et al., 2017), live data monitoring (Cohen et al., 2011), data exploration tools (Hassan et al., 2014), or human-in-the-loop systems (Nguyen et al., 2020), that all support digital journalism and automated fact-checking.

Extending to other domains and languages Misinformation is an issue that affects speakers of other languages beyond English. Future work should consider developing models and tooling to support the verification of claims in non-English languages. Replicating the dataset construction method from FEVER to build large-scale datasets for other languages will duplicate effort for each language targeted, incur high cost, and be prone to errors and biases introduced by the annotators. While data *can* be collected for training and evaluation, any modelling efforts should ensure that existing fact verification resources, multilingual models, and distantly supervised pre-training can be best exploited to maximise model performance.

Extending to other modalities and evidence sources A natural extension is to consider verifying claims that are more representative of real-world information

needs and against a broader range of evidence sources such as tables and info-boxes, or even evidence sources beyond Wikipedia. I have recently made in-roads into these two challenges through recent collaborations, extending the work of this thesis. (1) I have advised the construction of a new Wikipedia-based claim verification dataset that considers structured and unstructured data from full Wikipedia pages (rather than the introductory sections) in FEVEROUS (Aly et al., 2021) and (2) I have also led a collaborative project that uses real-world search engine queries rewritten as claims to approximate the information that users wish to fact-check (Thorne et al., 2021a). However, future work should consider how other types of evidence, such as databases, scientific reports, images and videos, can be incorporated as sources of evidence to verify a wider range of claim types.

Verifying the evidence FEVER used a static snapshot of Wikipedia as the knowledge source when predicting the veracity of claims. This type of reasoning requires supporting or refuting facts to be present before predicting claim veracity. As emerging events require verification, the approach used by FEVER would not be able to verify claims without this information; other sources of information, such as eye-witness accounts, may have to serve as evidence. It is also important to consider verifying the evidence used, a factor that is not considered by the closed-world modelling assumption used in FEVER. Multiple sources of evidence may need to be considered to reason about facts on specialist topics or emerging trends, with varying degrees of trust. Even when trust is established, claims may contradict established norms, requiring a great deal of care to weigh up what assertions should be challenged. For example, misinformation regarding a link between vaccines and autism was propagated as a result of a flawed scientific study leading to invalid conclusions (Eggertson, 2010). Even though the article was incorrect, the implied trust in publication media resulted in disinformation citing this study and establishing credibility.

Modelling A common challenge for the claim verification and correction models in Chapters 3 and 6 was the need for models to operate over potentially noisy retrieved evidence. The way in which models were exposed to retrieved evidence during training influenced their performance when combined with an information retrieval system for unseen claims, indicated in Section 3.5.6. Future work should investigate how to best use *both* the annotated evidence and retrieved evidence during training to improve their resilience to this source of noise. Domain adaptation through fine-tuning may allow models first to be trained on the oracle evidence before being adapted to use retrieved evidence. Furthermore, the parameters for the neural models for evidence retrieval were in the modelling approaches were static and not updated given their effect on the claim veracity prediction part of the model pipeline. Additional research is required to understand how the models for evidence retrieval impact the downstream claim veracity classifier and how their parameters can be updated jointly during training.

The large number of submissions to the FEVER shared task has helped identify broad trends in modelling for evidence retrieval and claim veracity prediction. These trends and results were reported for the first shared task by (Thorne et al., 2018b): all but one team modelled FEVER as two separate sub-tasks without joint training. Since the shared task, new methods have been developed with models that are jointly trained to perform both sub-tasks.

Jointly modelling retrieval with language modelling has allowed systems to predict the veracity of claims without labelled data for supervision, achieving accuracies within a few percentage points of supervised systems (Lewis et al., 2020). In supervised training, jointly modelling sentence selection with veracity prediction, Nie et al. (2020a) provide a model with combined labels which is accurate and data-efficient: requiring less training data. Similarly, Kernel Graph Attention Networks (Liu et al., 2020) jointly consider the relevance of evidence and evidence

interact and provide a well-grounded framework for modelling the combination of multiple evidence passages.

Finally, future modelling approaches do not necessarily have to follow the prescribed shared task formulation. For example, another successful modelling approach considers the logical relations between spans within the claim and evidence. The mutations in FEVER were designed to resemble the entailment relations in Natural Logical Inference (Angeli and Manning, 2014). This property has been explored by Krishna et al. (2021), who attain a top-5 score on the shared task leaderboard by modelling claim-level veracity prediction by first predicting the logical relations between spans in the claim and evidence. Not only does this yield high accuracy, the relation between the claim and evidence also serves as a further mechanism to communicate how a decision about a claim's accuracy was reached. Innovations that continue to challenge our modelling assumptions and provide new ways to explain or inform an end-user about why a claim is supported or refuted by evidence are essential to grow and develop automated fact verification.

Bibliography

E. Agirre, D. Cer, M. Diab, A. Gonzalez-Agirre, and W. Guo. *SEM 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43, Atlanta, Georgia, USA, 2013. Association for Computational Linguistics. URL https://aclanthology.org/S13-1004.

T. Alhindi, S. Petridis, and S. Muresan. Where is your evidence: Improving fact-checking by justification modeling. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 85–90, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5513. URL https://aclanthology.org/W18-5513.

H. Allcott and M. Gentzkow. Social media and fake news in the 2016 election. Technical report, National Bureau of Economic Research, 2017.

R. Aly, Z. Guo, M. Schlichtkrull, J. Thorne, A. Vlachos, C. Christodoulopoulos, O. Cocarascu, and A. Mittal. FEVEROUS: Fact Extraction and VERification Over Unstructured and Structured information. (NeurIPS 2021), 2021. URL http://arxiv.org/abs/2106.05707.

D. Andor, L. He, K. Lee, and E. Pitler. Giving BERT a calculator: Finding operations and arguments with reading comprehension. 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 5947–5952, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1609. URL https://aclanthology.org/D19-1609.

G. Angeli and C. D. Manning. NaturalLI: Natural logic inference for common sense reasoning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 534–545, Doha, Qatar, 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1059. URL https://aclanthology.org/D14-1059.

P. Atanasova, J. G. Simonsen, C. Lioma, and I. Augenstein. Generating fact checking explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7352–7364, Online, 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.656. URL https://aclanthology.org/2020.acl-main.656.

P. Atanasova, D. Wright, and I. Augenstein. Generating label cohesive and well-formed adversarial claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3168–3177, Online, Nov. 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.256. URL https://aclanthology.org/2020.emnlp-main.256.

M. Aubakirova and M. Bansal. Interpreting neural networks to improve politeness comprehension. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2035–2041, Austin, Texas, 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1216. URL https://aclanthology.org/D16-1216.

I. Augenstein, T. Rocktäschel, A. Vlachos, and K. Bontcheva. Stance detection with bidirectional conditional encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 876–885, Austin, Texas, 2016a.

Association for Computational Linguistics. doi: 10.18653/v1/D16-1084. URL https://aclanthology.org/D16-1084.

I. Augenstein, A. Vlachos, and K. Bontcheva. USFD at SemEval-2016 task 6: Any-target stance detection on Twitter with autoencoders. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 389–393, San Diego, California, 2016b. Association for Computational Linguistics. doi: 10.18653/v1/S16-1063. URL https://aclanthology.org/S16-1063.

I. Augenstein, S. Ruder, and A. Søgaard. Multi-task learning of pairwise sequence classification tasks over disparate label spaces. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1896–1906, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1172. URL https://aclanthology.org/N18-1172.

I. Augenstein, C. Lioma, D. Wang, L. Chaves Lima, C. Hansen, C. Hansen, and J. G. Simonsen. MultiFC: A real-world multi-domain dataset for evidence-based fact checking of claims. 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 4685–4697, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1475. URL https://aclanthology.org/D19-1475.

I. Avcibas, S. Bayram, N. Memon, M. Ramkumar, and B. Sankur. A classifier design for detecting image manipulations. *Proceedings - International Conference on Image Processing, ICIP*, 4:2645–2648, 2004. ISSN 15224880. doi: 10.1109/ICIP.2004.1421647.

M. F. Babakar and W. Moy. The State of Automated Factchecking. Technical report, 2016. URL https://fullfact.org/media/uploads/full_fact-the_state_of_automated_factchecking_aug_2016.pdf.

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In Y. Bengio and Y. LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1409.0473.

M. Bartolo, A. Roberts, J. Welbl, S. Riedel, and P. Stenetorp. Beat the AI: Investigating Adversarial Human Annotations for Reading Comprehension. 2020. URL http://arxiv.org/abs/2002.00293.

H. Bast, B. Buchhold, and E. Haussmann. Relevance scores for triples from type-like relations. In R. Baeza-Yates, M. Lalmas, A. Moffat, and B. A. Ribeiro-Neto, editors, *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, pages 243–252. ACM, 2015. doi: 10.1145/2766462.2767734. URL https://doi.org/10.1145/2766462.2767734.

H. Bast, B. Buchhold, and E. Haussmann. Overview of the Triple Scoring Task at the WSDM Cup 2017. *WSDM Cup*, 2017.

Y. Belinkov and Y. Bisk. Synthetic and natural noise both break neural machine translation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=BJ8vJebC-.

E. M. Bender and A. Koller. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185–5198, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.463. URL https://aclanthology.org/2020.acl-main.463.

C. Bhagavatula, S. Feldman, R. Power, and W. Ammar. Content-based citation recommendation. In *Proceedings of the 2018 Conference of the North American*

Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 238–251, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1022. URL https://aclanthology.org/N18-1022.

K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. *SIGMOD 08 Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, 2008. ISSN 07308078. doi: 10.1145/1376616.1376746. URL http://doi.acm.org/10.1145/1376616.1376746.

S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://aclanthology.org/D15-1075.

S. Brin. Extracting Patterns and Relations from the World Wide Web. *The World Wide Web and Databases*, 53(9):172–183, 1998. ISSN 1098-6596. doi: 10.1017/CBO9781107415324.004. URL https://arxiv.org/abs/1011.1669v3.

C. Budak, D. Agrawal, and A. E. Abbadi. Limiting the spread of misinformation in social networks. In S. Srinivasan, K. Ramamritham, A. Kumar, M. P. Ravindra, E. Bertino, and R. Kumar, editors, *Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28 - April 1, 2011*, pages 665–674. ACM, 2011. doi: 10.1145/1963405.1963499. URL https://doi.org/10.1145/1963405.1963499.

R. Bunescu and M. Paşca. Using encyclopedic knowledge for named entity disambiguation. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, Trento, Italy, 2006. Association for Computational Linguistics. URL https://aclanthology.org/E06-1002.

F. Burlot and F. Yvon. Evaluating the morphological competence of machine translation systems. In *Proceedings of the Second Conference on Machine Translation*, pages 43–55, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4705. URL https://aclanthology.org/W17-4705.

O. Camburu, T. Rocktäschel, T. Lukasiewicz, and P. Blunsom. e-snli: Natural language inference with natural language explanations. In S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 9560–9572, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/4c7a167bb329bd92580a99ce422d6fa6-Abstract.html.

M. Cao, Y. Dong, J. Wu, and J. C. K. Cheung. Factual error correction for abstractive summarization models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6251–6258, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.506. URL https://aclanthology.org/2020.emnlp-main.506.

C. Castillo, M. Mendoza, and B. Poblete. Information credibility on Twitter. *Proceedings of the 20th International Conference Companion on World Wide Web, WWW 2011*, pages 675–684, 2011. doi: 10.1145/1963405.1963500.

C. Castillo, M. Mendoza, and B. Poblete. Predicting information credibility in time-sensitive social media. *Internet Research*, 23(5):560–588, 2013. ISSN 10662243. doi: 10.1108/IntR-05-2012-0095.

A. Chakraborty, B. Paranjape, S. Kakarla, and N. Ganguly. Stop clickbait: Detecting and preventing clickbaits in online news media. In R. Kumar, J. Caverlee, and H. Tong, editors, *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2016, San Francisco, CA, USA, August 18-21, 2016*,

pages 9–16. IEEE Computer Society, 2016. doi: 10.1109/ASONAM.2016.7752207. URL https://doi.org/10.1109/ASONAM.2016.7752207.

D. Chen, A. Fisch, J. Weston, and A. Bordes. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada, 2017a. Association for Computational Linguistics. doi: 10.18653/v1/P17-1171. URL https://aclanthology.org/P17-1171.

Q. Chen, X. Zhu, Z.-H. Ling, S. Wei, H. Jiang, and D. Inkpen. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668, Vancouver, Canada, 2017b. Association for Computational Linguistics. doi: 10.18653/v1/P17-1152. URL https://aclanthology.org/P17-1152.

S. Chen, D. Khashabi, W. Yin, C. Callison-Burch, and D. Roth. Seeing things from a different angle:discovering diverse perspectives about claims. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 542–557, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1053. URL https://aclanthology.org/N19-1053.

W. Chen, H. Wang, J. Chen, Y. Zhang, H. Wang, S. Li, X. Zhou, and W. Y. Wang. Tabfact: A large-scale dataset for table-based fact verification. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=rkeJRhNYDH.

Y. Chen, N. J. Conroy, and V. L. Rubin. Misleading Online Content. *Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection - WMDD '15*, (November): 15–19, 2015. doi: 10.1145/2823465.2823467. URL http://dl.acm.org/citation.cfm?doid=2823465.2823467.

A. Chernyavskiy, D. Ilvovsky, and P. Nakov. WhatTheWikiFact: Fact-Checking Claims Against Wikipedia. 2021. URL http://arxiv.org/abs/2105.00826.

S. Chesney, M. Liakata, M. Poesio, and M. Purver. Incongruent headlines: Yet another way to mislead your readers. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 56–61, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4210. URL https://aclanthology.org/W17-4210.

S. L. Christopher and H. A. Rahulnath. Review authenticity verification using supervised learning and reviewer personality traits. *Proceedings of IEEE International Conference on Emerging Technological Trends in Computing, Communications and Electrical Engineering, ICETT 2016*, pages 6–12, 2016. doi: 10.1109/ICETT.2016.7873647.

G. L. Ciampaglia, P. Shiralkar, L. M. Rocha, J. Bollen, F. Menczer, and A. Flammini. Computational fact checking from knowledge networks. *PLoS ONE*, 10(6):1–13, 2015. ISSN 19326203. doi: 10.1371/journal.pone.0128193.

C. Clark, K. Lee, M.-W. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300.

P. Clark. Elementary school science and math tests as a driver for AI: take the aristo challenge! In B. Bonet and S. Koenig, editors, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 4019–4021. AAAI Press, 2015. URL http://www.aaai.org/ocs/index.php/IAAI/IAAI15/paper/view/10003.

A. Cohan, W. Ammar, M. van Zuylen, and F. Cady. Structural scaffolds for citation intent classification in scientific publications. In *Proceedings of the 2019 Conference*

of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 3586–3596, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1361. URL https://aclanthology.org/N19-1361.

J. Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960. doi: 10.1177/001316446002000104. URL https://doi.org/10.1177/001316446002000104.

S. Cohen, C. Li, J. Yang, and C. Yu. Computational journalism: A call to arms to database researchers. In *CIDR 2011, Fifth Biennial Conference on Innovative Data Systems Research, Asilomar, CA, USA, January 9-12, 2011, Online Proceedings*, pages 148–151. www.cidrdb.org, 2011. URL http://cidrdb.org/cidr2011/Papers/CIDR11_Paper17.pdf.

G. Da San Martino, S. Yu, A. Barrón-Cedeño, R. Petrov, and P. Nakov. Fine-grained analysis of propaganda in news articles. In *EMNLP 2019*, number 4, 2019.

I. Dagan, O. Glickman, and B. Magnini. The PASCAL Recognising Textual Entailment Challenge. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3944 LNAI:177–190, 2006. ISSN 03029743. doi: 10.1007/11736790_9.

I. Dagan, B. Dolan, B. Magnini, and D. Roth. Recognizing textual entailment: Rational, evaluation and approaches. *Natural Language Engineering*, 15(4):i–xvii, 2009. ISSN 1351-3249. doi: 10.1017/S1351324909990209. URL http://www.journals.cambridge.org/abstract_S1351324909990209.

N. De Cao, G. Izacard, S. Riedel, and F. Petroni. Autoregressive Entity Retrieval. pages 1–18, 2020. URL http://arxiv.org/abs/2010.00904.

M.-C. de Marneffe, A. N. Rafferty, and C. D. Manning. Finding contradictions in text. In *Proceedings of ACL-08: HLT*, pages 1039–1047, Columbus, Ohio, 2008. Association for Computational Linguistics. URL https://aclanthology.org/P08-1118.

L. Derczynski and K. Bontcheva. Veracy in Digital Social Networks. *Pheme.eu*, 2015. URL `http://ceur-ws.org/Vol-1181/pros2014_paper_05.pdf`.

L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. Wong Sak Hoi, and A. Zubiaga. SemEval-2017 task 8: RumourEval: Determining rumour veracity and support for rumours. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 69–76, Vancouver, Canada, 2017. Association for Computational Linguistics. doi: 10.18653/v1/S17-2006. URL `https://aclanthology.org/S17-2006`.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL `https://aclanthology.org/N19-1423`.

N. Diakopoulos. Algorithmic Accountability. *Digital Journalism*, 0811(February 2015): 1–18, 2015. ISSN 2167-0811. doi: 10.1080/21670811.2014.976411. URL `http://www.tandfonline.com/doi/abs/10.1080/21670811.2014.976411`.

N. Diakopoulos. Accountability in Algorithmic Decision Making. *Communications of the Acm*, 59(2), 2016. ISSN 00010782. doi: 10.1145/2844110.

T. Diggelmann, J. Boyd-Graber, J. Bulian, M. Ciaramita, and M. Leippold. CLIMATE-FEVER: A Dataset for Verification of Real-World Climate Claims. pages 1–16, 2020. URL `http://arxiv.org/abs/2012.00614`.

G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, 2004. European Language Resources Association (ELRA). URL `http://www.lrec-conf.org/proceedings/lrec2004/pdf/5.pdf`.

X. L. Dong, E. Gabrilovich, K. Murphy, V. Dang, W. Horn, C. Lugaresi, S. Sun, and W. Zhang. Knowledge-Based Trust: Estimating the Trustworthiness of Web Sources. (Section 3), 2015. ISSN 21508097. doi: 10.14778/2777598.2777603. URL http://arxiv.org/abs/1502.03519.

F. Doshi-Velez and B. Kim. Towards A Rigorous Science of Interpretable Machine Learning. (Ml):1–13, 2017. URL http://arxiv.org/abs/1702.08608.

J. Ebrahimi, A. Rao, D. Lowd, and D. Dou. HotFlip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 31–36, Melbourne, Australia, 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2006. URL https://aclanthology.org/P18-2006.

L. Eggertson. Lancet retracts 12-year-old article linking autism to MMR vaccines. *CMAJ : Canadian Medical Association journal = journal de l'Association medicale canadienne*, 182(4):199–200, 2010. ISSN 14882329. doi: 10.1503/cmaj.109-3179.

S. Feng, R. Banerjee, and Y. Choi. Syntactic stylometry for deception detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 171–175, Jeju Island, Korea, 2012. Association for Computational Linguistics. URL https://aclanthology.org/P12-2034.

V. W. Feng and G. Hirst. Detecting deceptive opinions with profile compatibility. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 338–346, Nagoya, Japan, 2013. Asian Federation of Natural Language Processing. URL https://aclanthology.org/I13-1039.

W. Ferreira and A. Vlachos. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1163–1168, San Diego, California, 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1138. URL https://aclanthology.org/N16-1138.

J. L. Fleiss. Measuring Naminal Scale Agreement mong many Raters. *Psychological Bulletin*, 76(5):378–382, 1971.

R. M. French. Catastrophic forgetting in connectionists networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999.

A. J. Fridrich, B. D. Soukal, and A. J. Lukáš. Detection of copy-move forgery in digital images. In *in Proceedings of Digital Forensic Research Workshop*, pages 1163–1168, 2003. ISBN 9781509002870. doi: 10.1109/ICMLA.2015.137.

Full Fact. Report on the Facebook Third Party Fact Checking programme. 2019. URL `https://fullfact.org/media/uploads/tpfc-q1q2-2019.pdf`.

M. Gardner, J. Grus, M. Neumann, O. Tafjord, P. Dasigi, N. F. Liu, M. Peters, M. Schmitz, and L. Zettlemoyer. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia, 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-2501. URL `https://aclanthology.org/W18-2501`.

P. Gencheva, P. Nakov, L. Màrquez, A. Barrón-Cedeño, and I. Koychev. A context-aware approach for detecting worth-checking claims in political debates. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 267–276, Varna, Bulgaria, 2017. INCOMA Ltd. doi: 10.26615/978-954-452-049-6_037. URL `https://doi.org/10.26615/978-954-452-049-6_037`.

M. Geva, Y. Goldberg, and J. Berant. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. 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 1161–1166, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1107. URL `https://aclanthology.org/D19-1107`.

D. Giampiccolo, H. T. Dang, B. Magnini, I. Dagan, and B. Dolan. The Fourth FASCAL Recognizing Textual Entailment Challenge. *Proceedings of the First Text Analysis Conference*, 2008.

M. Glenski, T. Weninger, and S. Volkova. Identifying and understanding user reactions to deceptive and trusted social news sources. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 176–181, Melbourne, Australia, 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2029. URL `https://aclanthology.org/P18-2029`.

L. Graves. Understanding the Promise and Limits of Automated Fact-Checking. *Technical Report, Reuters Institute, University of Oxford*, (February):1–7, 2018. ISSN 0717-6163. doi: 10.1007/s13398-014-0173-7.2. URL `https://arxiv.org/abs/1011.1669v3`.

A. Guess, B. Nyhan, and J. Reifler. Selective Exposure to Misinformation: Evidence from the consumption of fake news during the 2016 U.S. presidential campaign. *European Research Council*, 9(682758):9, 2018.

A. M. Guess and B. A. Lyons. Misinformation, Disinformation, and Online Propaganda. *Social Media and Democracy*, (September):10–33, 2020. doi: 10.1017/9781108890960.003.

S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. Bowman, and N. A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2017. URL `https://aclanthology.org/N18-2017`.

K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-w. Chang. REALM : Retrieval-Augmented Language Model Pre-Training. 2020. URL `https://arxiv.org/abs/2002.08909v1`.

Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with TrustRank. *Proceedings of the Thirtieth international conference on Very large data bases*, 30:

576–587, 2004. doi: 10.3810/psm.2004.05.304. URL http://dl.acm.org/citation.cfm?id=1316689.1316740.

Z. Hai, P. Zhao, P. Cheng, P. Yang, X.-L. Li, and G. Li. Deceptive review spam detection via exploiting task relatedness and unlabeled data. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1817–1826, Austin, Texas, 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1187. URL https://aclanthology.org/D16-1187.

A. Halevy, C. Canton-Ferrer, H. Ma, U. Ozertem, P. Pantel, M. Saeidi, F. Silvestri, and V. Stoyanov. Preserving integrity in online social networks. *arXiv*, pages 1–32, 2020. ISSN 23318422.

N.-R. Han, J. Tetreault, S.-H. Lee, and J.-Y. Ha. Using an error-annotated learner corpus to develop an ESL/EFL error correction system. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, 2010. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2010/pdf/821_Paper.pdf.

A. Hanselowski, H. Zhang, Z. Li, D. Sorokin, B. Schiller, C. Schulz, and I. Gurevych. UKP-athene: Multi-sentence textual entailment for claim verification. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 103–108, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5516. URL https://aclanthology.org/W18-5516.

A. Hanselowski, C. Stab, C. Schulz, Z. Li, and I. Gurevych. A richly annotated corpus for different tasks in automated fact-checking. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 493–503, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/K19-1046. URL https://aclanthology.org/K19-1046.

Y. Hao, L. Dong, F. Wei, and K. Xu. Visualizing and understanding the effectiveness of BERT. 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 4143–4152, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1424. URL https://aclanthology.org/D19-1424.

N. Hassan, A. Sultana, Y. Wu, G. Zhang, C. Li, J. Yang, and C. Yu. Data in, fact out: Automated Monitoring of Facts by FactWatcher. *Proceedings of the VLDB Endowment*, 7(13):1557–1560, 2014. ISSN 21508097. doi: 10.14778/2733004.2733029. URL http://dl.acm.org/citation.cfm?doid=2733004.2733029.

N. Hassan, B. Adair, J. T. Hamilton, C. Li, M. Tremayne, J. Yang, and C. Yu. The Quest to Automate Fact-Checking. In *world*, 2015a.

N. Hassan, C. Li, and M. Tremayne. Detecting check-worthy factual claims in presidential debates. In J. Bailey, A. Moffat, C. C. Aggarwal, M. de Rijke, R. Kumar, V. Murdock, T. K. Sellis, and J. X. Yu, editors, *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*, pages 1835–1838. ACM, 2015b. doi: 10.1145/2806416.2806652. URL https://doi.org/10.1145/2806416.2806652.

N. Hassan, M. Tremayne, F. Arslan, and C. Li. Comparing Automated Factual Claim Detection Against Judgments of Journalism Organizations. *Computation+Journalism Symposium*, 2016.

N. Hassan, A. K. Nayak, V. Sable, C. Li, M. Tremayne, G. Zhang, F. Arslan, J. Caraballo, D. Jimenez, S. Gawsane, S. Hasan, M. Joseph, and A. Kulkarni. ClaimBuster: the first-ever end-to-end fact-checking system. *Proceedings of the VLDB Endowment*, 10(12):1945–1948, 2017. ISSN 21508097. doi: 10.14778/3137765.3137815. URL http://www.vldb.org/pvldb/vol10/p1945-li.pdf.

R. Hill. The Full Fact Report 2020 Fighting the causes and consequences of bad information. Technical report, 2020. URL https://fullfact.org/media/uploads/fullfactreport2020.pdf.

S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9 (8):1735–1780, 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735.

A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi. The curious case of neural text degeneration. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=rygGQyrFvH.

D. A. Hughes. Acute chloroquine poisoning: A comprehensive experimental toxicology assessment of the role of diazepam. *British Journal of Pharmacology*, 177(21):4975–4989, 2020. ISSN 14765381. doi: 10.1111/bph.15101.

K. Hyland. Metadiscourse. *The International Encyclopedia of Language and Social Interaction*, (April 2015):1–11, 2015. doi: 10.1002/9781118611463/wbielsi003. URL https://www.researchgate.net/profile/Ken_Hyland/publication/285591598_Metadiscourse/links/5661183e08ae418a7866a4da.pdf.

P. Isabelle, C. Cherry, and G. Foster. A challenge set approach to evaluating machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2486–2496, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1263. URL https://aclanthology.org/D17-1263.

M. Iyyer, J. Wieting, K. Gimpel, and L. Zettlemoyer. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1170. URL https://aclanthology.org/N18-1170.

R. Jia and P. Liang. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark, 2017. Asso-

ciation for Computational Linguistics. doi: 10.18653/v1/D17-1215. URL https://aclanthology.org/D17-1215.

G. S. Jowett and V. O' Donnell. What Is Propaganda, and How Does It Differ From Persuasion? In *Propaganda and Misinformation*, chapter 1. Sage Publishers, 2006.

G. Karadzhov, P. Nakov, L. Màrquez, A. Barrón-Cedeño, and I. Koychev. Fully automated fact checking using external sources. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 344–353, Varna, Bulgaria, 2017. INCOMA Ltd. doi: 10.26615/978-954-452-049-6_046. URL https://doi.org/10.26615/978-954-452-049-6_046.

V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550.

D. Kiela, H. Firooz, A. Mohan, V. Goswami, A. Singh, P. Ringshia, and D. Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/1b84c4cee2b8b3d823b30e2d604b1878-Abstract.html.

J. Kim and K. Choi. Unsupervised fact checking by counter-weighted positive and negative evidential paths in a knowledge graph. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1677–1686, Barcelona, Spain (Online), Dec. 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.147. URL https://aclanthology.org/2020.coling-main.147.

J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences of the United States of America*, 114(13):3521–3526, 2017. ISSN 10916490. doi: 10.1073/pnas.1611835114.

K. Knight and I. Chander. Automated postediting of documents. *Proceedings of the National Conference on Artificial Intelligence*, 1:779–784, 1994.

L. Konstantinovskiy, O. Price, M. Babakar, and A. Zubiaga. Towards automated factchecking: Developing an annotation schema and benchmark for consistent automated claim detection. *arXiv*, pages 1–15, 2018. ISSN 23318422.

N. Kotonya and F. Toni. Explainable automated fact-checking for public health claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7740–7754, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.623. URL `https://aclanthology.org/2020.emnlp-main.623`.

B. Kovach and T. Rosenstiel. *The elements of journalism: What newspeople should know and the public should expect*. Three Rivers Press (CA), 2014.

A. Krishna, S. Riedel, and A. Vlachos. Proofver: Natural logic theorem proving for fact verification. *CoRR*, abs/2108.11357, 2021. URL `https://arxiv.org/abs/2108.11357`.

R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. 2016. URL `https://arxiv.org/abs/1602.07332`.

P. Le and I. Titov. Improving entity linking by modeling latent relations between mentions. In *Proceedings of the 56th Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 1595–1604, Melbourne, Australia, 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1148. URL https://aclanthology.org/P18-1148.

N. Lee, B. Z. Li, S. Wang, W.-t. Yih, H. Ma, and M. Khabsa. Language models as fact checkers? In *Proceedings of the Third Workshop on Fact Extraction and VERification (FEVER)*, pages 36–41, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.fever-1.5. URL https://aclanthology.org/2020.fever-1.5.

H. J. Levesque. On our best behaviour. *IJCAI*, 2013. ISSN 0035-8797.

R. Levy, Y. Bilu, D. Hershcovich, E. Aharoni, and N. Slonim. Context dependent claim detection. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1489–1500, Dublin, Ireland, 2014. Dublin City University and Association for Computational Linguistics. URL https://aclanthology.org/C14-1141.

P. S. H. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Q. Li and W. Zhou. Connecting the dots between fact verification and fake news detection. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1820–1825, Barcelona, Spain (Online), 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.165. URL https://aclanthology.org/2020.coling-main.165.

X. Li. Improving Knowledge Base Population With Information Extraction. 2016.

C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

T. Linzen, E. Dupoux, and Y. Goldberg. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535, 2016. doi: 10.1162/tacl_a_00115. URL https://aclanthology.org/Q16-1037.

N. F. Liu, R. Schwartz, and N. A. Smith. Inoculation by fine-tuning: A method for analyzing challenge datasets. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2171–2179, Minneapolis, Minnesota, 2019a. Association for Computational Linguistics. doi: 10.18653/v1/N19-1225. URL https://aclanthology.org/N19-1225.

X. Liu, A. Nourbakhsh, Q. Li, R. Fang, and S. Shah. Real-time rumor debunking on twitter. In J. Bailey, A. Moffat, C. C. Aggarwal, M. de Rijke, R. Kumar, V. Murdock, T. K. Sellis, and J. X. Yu, editors, *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*, pages 1867–1870. ACM, 2015. doi: 10.1145/2806416.2806651. URL https://doi.org/10.1145/2806416.2806651.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. 2019b. URL http://arxiv.org/abs/1907.11692.

Z. Liu, C. Xiong, M. Sun, and Z. Liu. Fine-grained fact verification with kernel graph attention network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7342–7351, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.655. URL https://aclanthology.org/2020.acl-main.655.

Y. Long, Q. Lu, R. Xiang, M. Li, and C.-R. Huang. Fake news detection through multi-perspective speaker profiles. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 252–256, Taipei, Taiwan, 2017. Asian Federation of Natural Language Processing. URL https://aclanthology.org/I17-2043.

M. Lukasik, K. Bontcheva, T. Cohn, A. Zubiaga, M. Liakata, and R. Procter. Using Gaussian Processes for Rumour Stance Classification in Social Media. 2016. URL http://arxiv.org/abs/1609.01962.

R. K. Mahabadi, Y. Belinkov, and J. Henderson. End-to-end bias mitigation by modelling biases in corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8706–8716, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.769. URL https://aclanthology.org/2020.acl-main.769.

T. Mahler, W. Cheung, M. Elsner, D. King, M.-C. de Marneffe, C. Shain, S. Stevens-Guille, and M. White. Breaking NLP: Using morphosyntax, semantics, pragmatics and world knowledge to fool sentiment analysis systems. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 33–39, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5405. URL https://aclanthology.org/W17-5405.

C. Malon. Team papelo: Transformer networks at FEVER. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 109–113, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5517. URL https://aclanthology.org/W18-5517.

C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*,

pages 55–60, Baltimore, Maryland, 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-5010. URL https://aclanthology.org/P14-5010.

A. Mantzarlis. Will verification kill fact-checking?, 2015. URL https://www.poynter.org/news/will-verification-kill-fact-checking.

B. Martens, L. Aguiar, E. Gomez-Herrera, and F. Mueller-Langer. The digital transformation of news media and the rise of disinformation and fake news. Technical report, European Commission, Seville, Spain, 2018. URL https://ec.europa.eu/jrc/sites/jrcsh/files/jrc111529.pdf.

J. Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(1-76), 2020. URL http://arxiv.org/abs/1412.1193.

T. McCoy, E. Pavlick, and T. Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1334. URL https://aclanthology.org/P19-1334.

P. Mcnamee, H. Simpson, and H. T. Dang. Overview of the TAC 2009 Knowledge Base Population Track. (November), 2009.

M. Mendoza, B. Poblete, and C. Castillo. Twitter under crisis: Can we trust what we RT? *SOMA 2010 - Proceedings of the 1st Workshop on Social Media Analytics*, pages 71–79, 2010. doi: 10.1145/1964858.1964869.

R. Mihalcea and C. Strapparava. The lie detector: Explorations in the automatic recognition of deceptive language. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 309–312, Suntec, Singapore, 2009. Association for Computational Linguistics. URL https://aclanthology.org/P09-2078.

T. Mihaylova, P. Nakov, L. Màrquez, A. Barrón-Cedeño, M. Mohtarami, G. Karadzhov, and J. Glass. Fact checking in community forums. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

G. A. Miller. WordNet: A lexical database for English. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992. URL https://aclanthology.org/H92-1116.

S. Miranda, D. Nogueira, A. Mendes, A. Vlachos, A. Secker, R. Garrett, J. Mitchell, and Z. Marinho. Automated fact checking in the news room. In L. Liu, R. W. White, A. Mantrach, F. Silvestri, J. J. McAuley, R. Baeza-Yates, and L. Zia, editors, *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 3579–3583. ACM, 2019. doi: 10.1145/3308558.3314135. URL https://doi.org/10.1145/3308558.3314135.

M. Mitchell, S. Wu, A. Zaldivar, P. Barnes, L. Vasserman, B. Hutchinson, E. Spitzer, I. D. Raji, and T. Gebru. Model cards for model reporting. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 220–229, 2019.

Y. Mu and N. Aletras. Identifying Twitter users who repost unreliable news sources with linguistic information. *PeerJ Computer Science*, 6:1–18, 2020. ISSN 23765992. doi: 10.7717/peerj-cs.325.

A. Naik, A. Ravichander, N. Sadeh, C. Rose, and G. Neubig. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, Santa Fe, New Mexico, USA, 2018. Association for Computational Linguistics. URL https://aclanthology.org/C18-1198.

K. Nakamura, S. Levy, and W. Y. Wang. Fakeddit: A new multimodal benchmark dataset for fine-grained fake news detection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6149–6157, Marseille, France, May 2020.

European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.755.

N. Nakashole and T. M. Mitchell. Language-aware truth assessment of fact candidates. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1009–1019, Baltimore, Maryland, 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-1095. URL https://aclanthology.org/P14-1095.

P. Nakov. Can we spot the "fake news" before it was even written? *arXiv*, pages 1–14, 2020. ISSN 23318422.

H. T. Ng, S. M. Wu, Y. Wu, C. Hadiwinoto, and J. Tetreault. The CoNLL-2013 shared task on grammatical error correction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–12, Sofia, Bulgaria, 2013. Association for Computational Linguistics. URL https://aclanthology.org/W13-3601.

H. T. Ng, S. M. Wu, T. Briscoe, C. Hadiwinoto, R. H. Susanto, and C. Bryant. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-1701. URL https://aclanthology.org/W14-1701.

D. Nguyen. Comparing automatic and human evaluation of local explanations for text classification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1069–1078, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1097. URL https://aclanthology.org/N18-1097.

T. T. Nguyen, M. Weidlich, H. Yin, B. Zheng, Q. H. Nguyen, and Q. V. H. Nguyen. Fact-catch: Incremental pay-as-you-go fact checking with minimal user effort. In *Proceedings*

of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 2165–2168, 2020.

F. Nie, J.-G. Yao, J. Wang, R. Pan, and C.-Y. Lin. A simple recipe towards reducing hallucination in neural surface realisation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2673–2679, Florence, Italy, 2019a. Association for Computational Linguistics. doi: 10.18653/v1/P19-1256. URL https://aclanthology.org/P19-1256.

Y. Nie, H. Chen, and M. Bansal. Combining fact extraction and verification with neural semantic matching networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6859–6866. AAAI Press, 2019b. doi: 10.1609/aaai.v33i01.33016859. URL https://doi.org/10.1609/aaai.v33i01.33016859.

Y. Nie, L. Bauer, and M. Bansal. Simple compounded-label training for fact extraction and verification. In *Proceedings of the Third Workshop on Fact Extraction and VERification (FEVER)*, pages 1–7, Online, July 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.fever-1.1. URL https://aclanthology.org/2020.fever-1.1.

Y. Nie, X. Zhou, and M. Bansal. What can we learn from collective human opinions on natural language inference data? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9131–9143, Online, 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.734. URL https://aclanthology.org/2020.emnlp-main.734.

J. Nørregaard and L. Derczynski. DanFEVER: claim verification dataset for Danish. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*,

pages 422–428, Reykjavik, Iceland (Online), 2021. Linköping University Electronic Press, Sweden. URL https://aclanthology.org/2021.nodalida-main.47.

M. Ott, Y. Choi, C. Cardie, and J. T. Hancock. Finding deceptive opinion spam by any stretch of the imagination. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 309–319, Portland, Oregon, USA, 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1032.

M. Ott, C. Cardie, and J. T. Hancock. Negative deceptive opinion spam. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 497–501, Atlanta, Georgia, 2013. Association for Computational Linguistics. URL https://aclanthology.org/N13-1053.

R. M. Palau and M. F. Moens. Argumentation mining: The detection, classification and structure of arguments in text. *Proceedings of the International Conference on Artificial Intelligence and Law*, pages 98–107, 2009. doi: 10.1145/1568234.1568246.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040.

A. Parikh, O. Täckström, D. Das, and J. Uszkoreit. A decomposable attention model for natural language inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2249–2255, Austin, Texas, 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1244. URL https://aclanthology.org/D16-1244.

E. Pavlick and T. Kwiatkowski. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694, 2019. doi: 10.1162/tacl_a_00293. URL https://aclanthology.org/Q19-1043.

J. W. Pennebaker, R. J. Booth, and M. E. Francis. Operator's Manual: Linguistic Inquiry and Word Count - LIWC2007. pages 1–11, 2007. ISSN 0040-5736. doi: 10.4018/978-1-60960-741-8.ch012.

J. W. Pennebaker, R. L. Boyd, K. Jordan, and K. Blackburn. The Development and Psychometric Properties of LIWC2015. *Environment and Planning D: Society and Space*, 2015. ISSN 0263-7758. doi: 10.1068/d010163. URL https://repositories.lib.utexas.edu/bitstream/handle/2152/31333/LIWC2015_LanguageManual.pdf.

J. Pennington, R. Socher, and C. Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL https://aclanthology.org/D14-1162.

V. Pérez-Rosas and R. Mihalcea. Experiments in open domain deception detection. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1120–1125, Lisbon, Portugal, 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1133. URL https://aclanthology.org/D15-1133.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL https://aclanthology.org/N18-1202.

F. Petroni, T. Rocktäschel, S. Riedel, P. Lewis, A. Bakhtin, Y. Wu, and A. Miller. Language models as knowledge bases? 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 2463–2473, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1250. URL https://aclanthology.org/D19-1250.

F. Petroni, A. Piktus, A. Fan, P. Lewis, M. Yazdani, N. De Cao, J. Thorne, Y. Jernite, V. Karpukhin, J. Maillard, V. Plachouras, T. Rocktäschel, and S. Riedel. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.200. URL https://aclanthology.org/2021.naacl-main.200.

A. Poliak, J. Naradowsky, A. Haldar, R. Rudinger, and B. Van Durme. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-2023. URL https://aclanthology.org/S18-2023.

D. Pomerleau and D. Rao. Fake News Challenge. http://fakenewschallenge.org/, 2017.

A. C. Popescu and H. Farid. Exposing digital forgeries by detecting traces of resampling. *IEEE Transactions on Signal Processing*, 53(2):758–767, 2005. ISSN 10009825. doi: 10.1109/TSP.2004.839932.

M. Potthast, S. Köpsel, B. Stein, and M. Hagen. Clickbait Detection. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9626(1):810–817, 2016. ISSN 16113349. doi: 10.1007/978-3-319-30671-1_72.

R. Procter, F. Vis, and A. Voss. Reading the riots on Twitter: Methodological innovation for the analysis of big data. *International Journal of Social Research Methodology*, 16 (3):197–214, 2013. ISSN 13645579. doi: 10.1080/13645579.2013.774172.

V. Qazvinian, E. Rosengren, D. R. Radev, and Q. Mei. Rumor has it: Identifying misinformation in microblogs. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1589–1599, Edinburgh, Scotland, UK., 2011. Association for Computational Linguistics. URL https://aclanthology.org/D11-1147.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving Language Understanding by Generative Pre-Training. *arXiv*, pages 1–12, 2018. URL https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf.

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21:1–67, 2020. URL http://arxiv.org/abs/1910.10683.

P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. URL https://aclanthology.org/D16-1264.

H. Rashkin, E. Choi, J. Y. Jang, S. Volkova, and Y. Choi. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2931–2937, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1317. URL https://aclanthology.org/D17-1317.

M. Redi, B. Fetahu, J. T. Morgan, and D. Taraborelli. Citation needed: A taxonomy and algorithmic assessment of wikipedia's verifiability. In L. Liu, R. W. White, A. Mantrach, F. Silvestri, J. J. McAuley, R. Baeza-Yates, and L. Zia, editors, *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17,*

2019, pages 1567–1578. ACM, 2019. doi: 10.1145/3308558.3313618. URL https://doi.org/10.1145/3308558.3313618.

X. Ren, J. Liu, X. Yu, U. Khandelwal, Q. Gu, L. Wang, and J. Han. Cluscite: effective citation recommendation by information network-based clustering. In S. A. Macskassy, C. Perlich, J. Leskovec, W. Wang, and R. Ghani, editors, *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 821–830. ACM, 2014. doi: 10.1145/2623330.2623630. URL https://doi.org/10.1145/2623330.2623630.

P. S. Resnik. Selection and information: A class-based approach to lexical relationships, 1993.

M. T. Ribeiro, S. Singh, and C. Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In B. Krishnapuram, M. Shah, A. J. Smola, C. C. Aggarwal, D. Shen, and R. Rastogi, editors, *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1135–1144. ACM, 2016. doi: 10.1145/2939672.2939778. URL https://doi.org/10.1145/2939672.2939778.

M. T. Ribeiro, S. Singh, and C. Guestrin. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865, Melbourne, Australia, 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1079. URL https://aclanthology.org/P18-1079.

M. T. Ribeiro, C. Guestrin, and S. Singh. Are red roses red? evaluating consistency of question-answering models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6174–6184, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1621. URL https://aclanthology.org/P19-1621.

M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.442. URL https://aclanthology.org/2020.acl-main.442.

M. Richardson, R. Agrawal, and P. Domingos. Trust Management for the semantic web. In *LNCS-ISWC'03 Proceedings of the Second International Conference on Semantic Web Conference*, pages 351–368, Sanibel Island, FL, 2003. Springer-Verlag, Berlin.

B. Riedel, I. Augenstein, G. P. Spithourakis, and S. Riedel. A simple but tough-to-beat baseline for the Fake News Challenge stance detection task. pages 1–6, 2017. URL http://arxiv.org/abs/1707.03264.

S. E. Robertson and S. Walker. Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval. In *Sigir '94*, number January 1994, 1994. ISBN 9781447120995. doi: 10.1007/978-1-4471-2099-5.

T. Rocktäschel, E. Grefenstette, K. M. Hermann, T. Kociský, and P. Blunsom. Reasoning about entailment with neural attention. In Y. Bengio and Y. LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. URL http://arxiv.org/abs/1509.06664.

A. Rogers, O. Kovaleva, and A. Rumshisky. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866, 2020. doi: 10.1162/tacl_a_00349. URL https://aclanthology.org/2020.tacl-1.54.

A. Rohrbach, L. A. Hendricks, K. Burns, T. Darrell, and K. Saenko. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4035–4045, Brussels, Belgium, 2018. Association

for Computational Linguistics. doi: 10.18653/v1/D18-1437. URL https://aclanthology.org/D18-1437.

A. S. Ross, M. C. Hughes, and F. Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. In C. Sierra, editor, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 2662–2670. ijcai.org, 2017. doi: 10.24963/ijcai.2017/371. URL https://doi.org/10.24963/ijcai.2017/371.

V. L. Rubin and T. Lukoianova. Full-Text Citation Analysis : A New Method to Enhance. *JOURNAL OF THE ASSOCIATION FOR INFORMATION SCIENCE AND TECHNOLOGY*, 66(5):905–917, 2014. URL https://asistdl.onlinelibrary.wiley.com/doi/epdf/10.1002/asi.23216.

V. L. Rubin and T. Vashchilko. Identification of truth and deception in text: Application of vector space model to Rhetorical Structure Theory. In *Proceedings of the Workshop on Computational Approaches to Deception Detection*, pages 97–106, Avignon, France, 2012. Association for Computational Linguistics. URL https://aclanthology.org/W12-0415.

S. Ruder. An Overview of Multi-Task Learning in Deep Neural Networks. 2017. URL http://arxiv.org/abs/1706.05098.

W. Samek, T. Wiegand, and K.-R. Müller. Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models. *ITU Journal: ICT Discoveries, Special Issue*, (1), 2017. URL http://arxiv.org/abs/1708.08296.

D. Saunders, F. Stahlberg, A. de Gispert, and B. Byrne. Domain adaptive inference for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 222–228, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1022. URL https://aclanthology.org/P19-1022.

T. Schuster, D. Shah, Y. J. S. Yeo, D. Roberto Filizzola Ortiz, E. Santus, and R. Barzilay. Towards debiasing fact verification models. 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 3419–3425, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1341. URL https://aclanthology.org/D19-1341.

T. Schuster, R. Schuster, D. J. Shah, and R. Barzilay. The limitations of stylometry for detecting machine-generated fake news. *Computational Linguistics*, 46(2):499–510, 2020. doi: 10.1162/coli_a_00380. URL https://aclanthology.org/2020.cl-2.8.

T. Schuster, A. Fisch, and R. Barzilay. Get your vitamin C! robust fact verification with contrastive evidence. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 624–643, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.52. URL https://aclanthology.org/2021.naacl-main.52.

A. See, P. J. Liu, and C. D. Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada, 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1099. URL https://aclanthology.org/P17-1099.

D. J. Shah, T. Schuster, and R. Barzilay. Automatic Fact-guided Sentence Modification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. URL http://arxiv.org/abs/1909.13838.

I. Shapiro, C. Brin, I. Bédard-Brûlé, and K. Mychajlowycz. Verification as a strategic ritual how journalists retrospectively describe processes for ensuring accuracy. *Journalism Practice*, 7(6):657–673, 2013. ISSN 17512794. doi: 10.1080/17512786.2013.765638. URL http://dx.doi.org/10.1080/17512786.2013.765638.

B. Shi and T. Weninger. Discriminative predicate path mining for fact checking in knowledge graphs. *Knowledge-Based Systems*, 104:123–133, 2016. ISSN 09507051. doi: 10.1016/j.knosys.2016.04.015.

P. Shiralkar, A. Flammini, F. Menczer, and G. L. Ciampaglia. Finding streams in knowledge graphs to support fact checking. In *Proceedings - IEEE International Conference on Data Mining, ICDM*, volume 2017-Novem, pages 859–864, 2017. ISBN 9781538638347. doi: 10.1109/ICDM.2017.105.

K. Shu, L. Cui, S. Wang, D. Lee, and H. Liu. defend: Explainable fake news detection. In A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, and G. Karypis, editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 395–405. ACM, 2019. doi: 10.1145/3292500.3330935. URL `https://doi.org/10.1145/3292500.3330935`.

R. H. Shultz and R. Godson. *Dezinformatsia: Active measures in Soviet strategy*. Pergamon-Brassey's London, UK, 1984.

C. Silverman. Verification and Fact Checking. In C. Silverman, editor, *Verification Handbook: A Definitive Guide To Verifying Digital Content For Emergency Coverage*, chapter 26. European Journalism Centre (EJC), 2014.

K. Spärck Jones. A statistical interpretation of term specificity and its application in retrieval, 1972. ISSN 00220418.

K. Spärck Jones. *Synonymy and semantic classification*. Edinburgh University Press, 1986.

I. Staliūnaitė and B. Bonfil. Breaking sentiment analysis of movie reviews. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 61–64, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-5410. URL `https://aclanthology.org/W17-5410`.

D. Stammbach and E. Ash. e-FEVER: Explanations and Summaries for Automated Fact Checking. *Truth and Trust Online*, 2020. URL https://doi.org/10.3929/ethz-b-000453826.

S. Suntwal, M. Paul, R. Sharp, and M. Surdeanu. On the importance of delexicalization for fact verification. 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 3413–3418, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1340. URL https://aclanthology.org/D19-1340.

C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In Y. Bengio and Y. LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6199.

N. T. Tam, M. Weidlich, B. Zheng, H. Yin, N. Q. V. Hung, and B. Stantic. From anomaly detection to rumour detection using data streams of social platforms. *Proceedings of the VLDB Endowment*, 12(9):1016–1029, 2018. ISSN 21508097. doi: 10.14778/3329772.3329778.

W. L. Taylor. "Cloze Procedure": A New Tool for Measuring Readability. *Journalism Quarterly*, 30(4):415–433, 1953. ISSN 0022-5533. doi: 10.1177/107769905303000401.

B. Thompson, J. Gwinnup, H. Khayrallah, K. Duh, and P. Koehn. Overcoming catastrophic forgetting during domain adaptation of neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2062–2068, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1209. URL https://aclanthology.org/N19-1209.

J. Thorne and A. Vlachos. An extensible framework for verification of numerical claims. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 37–40, Valencia, Spain, 2017. Association for Computational Linguistics. URL https://aclanthology.org/E17-3010.

J. Thorne and A. Vlachos. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3346–3359, Santa Fe, New Mexico, USA, 2018. Association for Computational Linguistics. URL https://aclanthology.org/C18-1283.

J. Thorne and A. Vlachos. Adversarial attacks against Fact Extraction and VERification. 2019. URL http://arxiv.org/abs/1903.05543.

J. Thorne and A. Vlachos. Elastic weight consolidation for better bias inoculation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 957–964, Online, 2021a. Association for Computational Linguistics. URL https://aclanthology.org/2021.eacl-main.82.

J. Thorne and A. Vlachos. Evidence-based factual error correction. 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 3298–3309, Online, 2021b. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.256. URL https://aclanthology.org/2021.acl-long.256.

J. Thorne, M. Chen, G. Myrianthous, J. Pu, X. Wang, and A. Vlachos. Fake News Detection using Stacked Ensemble of Classifiers. In *Natural Language Processing meets Journalism workshop at EMNLP 2017*, pages 80–83, 2017. URL https://aclweb.org/anthology/W/W17/W17-4214.pdf.

J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference*

of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 809–819, New Orleans, Louisiana, 2018a. Association for Computational Linguistics. doi: 10.18653/v1/N18-1074. URL https://aclanthology.org/N18-1074.

J. Thorne, A. Vlachos, O. Cocarascu, C. Christodoulopoulos, and A. Mittal. The fact extraction and VERification (FEVER) shared task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 1–9, Brussels, Belgium, 2018b. Association for Computational Linguistics. doi: 10.18653/v1/W18-5501. URL https://aclanthology.org/W18-5501.

J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal. Generating token-level explanations for natural language inference. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 963–969, Minneapolis, Minnesota, 2019a. Association for Computational Linguistics. doi: 10.18653/v1/N19-1101. URL https://aclanthology.org/N19-1101.

J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal. Evaluating adversarial attacks against multiple fact verification systems. 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 2944–2953, Hong Kong, China, 2019b. Association for Computational Linguistics. doi: 10.18653/v1/D19-1292. URL https://aclanthology.org/D19-1292.

J. Thorne, A. Vlachos, O. Cocarascu, C. Christodoulopoulos, and A. Mittal. The Second Fact Extraction and VERification (FEVER2.0) Shared Task. In *Proceedings of the second workshop on Fact Extraction and VERification at EMNLP-IJCNLP2019*, Hong Kong, China, 2019c. Association for Computational Linguistics. doi: 10.18653/v1/w18-5501.

J. Thorne, M. Glockner, G. Vallejo, A. Vlachos, and I. Gurevych. Evidence-based Verification for Real World Information Needs. 2021a. URL http://arxiv.org/abs/2104.00640.

J. Thorne, M. Yazdani, M. Saeidi, F. Silvestri, S. Riedel, and A. Halevy. From natural language processing to neural databases. In *Proceedings of the VLDB Endowment*, volume 14, pages 1033–1039. VLDB Endowment, 2021b.

R. Torok. Symbiotic radicalisation strategies: Propaganda tools and neuro linguistic programming. In *8th Australian Security and Intelligence Conference*, pages 58–65, Perth, Western Australia, 2015. SRI Security Research Institute, Edith Cowan University,. doi: 10.4225/75/57a941a2d3350. URL http://ro.ecu.edu.au/asi/45.

M. Trokhymovych and D. Saez-Trumper. WikiCheck: An end-to-end open source Automatic Fact-Checking API based on Wikipedia. In *Conference on Information and Knowledge Management*. Association for Computing Machinery, 2021. URL http://arxiv.org/abs/2109.00835.

J. A. Tucker, A. Guess, P. Barberá, C. Vaccari, A. Siegel, S. Sanovich, D. Stukal, and B. Nyhan. Social media, political polarization, and political disinformation. *Hewlett Foundation*, (March):1–95, 2018. ISSN 1556-5068. URL http://www.infoanimales.com/tortugas-terrestre.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

O. Vinyals, M. Fortunato, and N. Jaitly. Pointer networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, pages 2692–2700, 2015.

A. Vlachos and S. Riedel. Fact checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 18–22, Baltimore, MD, USA, 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-2508. URL `https://aclanthology.org/W14-2508`.

A. Vlachos and S. Riedel. Identification and verification of simple claims about statistical properties. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2596–2601, Lisbon, Portugal, 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1312. URL `https://aclanthology.org/D15-1312`.

S. Vosoughi, D. Roy, and S. Aral. The spread of true and false news online. *Science*, 359(6380):1146–1151, 2018. ISSN 0036-8075. doi: 10.1126/science.aap9559. URL `http://science.sciencemag.org/content/359/6380/1146`.

D. Wadden, S. Lin, K. Lo, L. L. Wang, M. van Zuylen, A. Cohan, and H. Hajishirzi. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.609. URL `https://aclanthology.org/2020.emnlp-main.609`.

S. Wang, M. Yu, J. Jiang, W. Zhang, X. Guo, S. Chang, Z. Wang, T. Klinger, G. Tesauro, and M. Campbell. Evidence aggregation for answer re-ranking in open-domain question answering. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL `https://openreview.net/forum?id=rJl3yM-Ab`.

W. Y. Wang. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational*

Linguistics (Volume 2: Short Papers), pages 422–426, Vancouver, Canada, 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-2067. URL https://aclanthology.org/P17-2067.

Z. Waseem, J. Thorne, and J. Bingel. Bridging the Gaps: Multi Task Learning for Domain Transfer of Hate Speech Detection. In J. Golbeck, editor, *Online Harassment*, pages 29–55. Springer International Publishing, Cham, 2018. ISBN 978-3-319-78583-7. doi: 10.1007/978-3-319-78583-7_3. URL https://doi.org/10.1007/978-3-319-78583-7_3.

P. Wei, N. Xu, and W. Mao. Modeling conversation structure and temporal dynamics for jointly predicting rumor stance and veracity. 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 4787–4798, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1485. URL https://aclanthology.org/D19-1485.

S. Wiegreffe and Y. Pinter. Attention is not not explanation. 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 11–20, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1002. URL https://aclanthology.org/D19-1002.

A. Williams, N. Nangia, and S. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL https://aclanthology.org/N18-1101.

T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and*

Conference on Empirical Methods in Natural Language Processing, pages 347–354, Vancouver, British Columbia, Canada, 2005. Association for Computational Linguistics. URL https://aclanthology.org/H05-1044.

T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6.

W. Xu, C. Napoles, E. Pavlick, Q. Chen, and C. Callison-Burch. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415, 2016. doi: 10.1162/tacl_a_00107. URL https://aclanthology.org/Q16-1029.

T. Yoneda, J. Mitchell, J. Welbl, P. Stenetorp, and S. Riedel. UCL machine reading group: Four factor framework for fact finding (HexaF). In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 97–102, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5515. URL https://aclanthology.org/W18-5515.

P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. doi: 10.1162/tacl_a_00166. URL https://aclanthology.org/Q14-1006.

Z. Yuan and T. Briscoe. Grammatical error correction using neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–386, San

Diego, California, 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1042. URL https://aclanthology.org/N16-1042.

R. Zellers, Y. Bisk, R. Schwartz, and Y. Choi. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1009. URL https://aclanthology.org/D18-1009.

R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, and Y. Choi. Defending against neural fake news. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 9051–9062, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/3e9f0fc9b2f89e043bc6233994dfcf76-Abstract.html.

D. Y. Zhang, L. Shang, B. Geng, S. Lai, K. Li, H. Zhu, M. T. A. Amin, and D. Wang. Fauxbuster: A content-free fauxtography detector using social media comments. In N. Abe, H. Liu, C. Pu, X. Hu, N. K. Ahmed, M. Qiao, Y. Song, D. Kossmann, B. Liu, K. Lee, J. Tang, J. He, and J. S. Saltz, editors, *IEEE International Conference on Big Data, Big Data 2018, Seattle, WA, USA, December 10-13, 2018*, pages 891–900. IEEE, 2018. doi: 10.1109/BigData.2018.8622344. URL https://doi.org/10.1109/BigData.2018.8622344.

Q. Zhang, S. Zhang, J. Dong, J. Xiong, and X. Cheng. Automatic detection of rumor on social network. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9362:113–122, 2015. ISSN 16113349. doi: 10.1007/978-3-319-25207-0_10.

W. Zhang, Y. Deng, J. Ma, and W. Lam. AnswerFact: Fact checking in product question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural*

Language Processing (EMNLP), pages 2407–2417, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.188. URL https://aclanthology.org/2020.emnlp-main.188.

Z. Zhao, D. Dua, and S. Singh. Generating natural adversarial examples. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=H1BLjgZCb.

C. Zhou, G. Neubig, J. Gu, M. Diab, F. Guzmán, L. Zettlemoyer, and M. Ghazvininejad. Detecting hallucinated content in conditional neural sequence generation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1393–1404, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.120. URL https://aclanthology.org/2021.findings-acl.120.

J. Zhou, X. Han, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun. GEAR: Graph-based evidence aggregating and reasoning for fact verification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 892–901, Florence, Italy, 2019a. Association for Computational Linguistics. doi: 10.18653/v1/P19-1085. URL https://aclanthology.org/P19-1085.

K. Zhou, C. Shu, B. Li, and J. H. Lau. Early rumour detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1614–1623, Minneapolis, Minnesota, 2019b. Association for Computational Linguistics. doi: 10.18653/v1/N19-1163. URL https://aclanthology.org/N19-1163.

L. Zhou, J. K. Burgoon, J. F. Nunamaker, and D. Twitchell. Automating Linguistics-Based Cues for detecting deception in text-based asynchronous computer-mediated communication. *Group Decision and Negotiation*, 13(1):81–106, 2004. ISSN 09262644. doi: 10.1023/B:GRUP.0000011944.62889.6f. URL https://arxiv.org/abs/1112.2903v1.

X. Zhou and R. Zafarani. A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities. *ACM Computing Surveys*, 53(5), 2020. ISSN 15577341. doi: 10.1145/3395046.

D. Zlatkova, P. Nakov, and I. Koychev. Fact-checking meets fauxtography: Verifying claims about images. 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 2099–2108, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1216. URL `https://aclanthology.org/D19-1216`.

A. Zubiaga and H. Ji. Tweet, but verify: epistemic study of information verification on Twitter. *Social Network Analysis and Mining*, 4(1):1–12, 2014. ISSN 18695469. doi: 10.1007/s13278-014-0163-y.

A. Zubiaga, M. Liakata, and R. Procter. Exploiting Context for Rumour Detection in Social Media. In G. L. Ciampaglia, A. Mashhadi, and T. Yasseri, editors, *Social Informatics*, pages 109–123, Cham, 2017. Springer International Publishing. ISBN 978-3-319-67217-5.

A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, and R. Procter. Detection and resolution of rumours in social media: A survey. *ACM Computing Surveys (CSUR)*, 51(2), 2018. ISSN 23318422.

Appendix A

Supplementary materials

A.1 Full FEVER annotation guidelines

A.1.1 Task 1 definitions

Claim A claim is a single sentence expressing information (true or mutated) about a single aspect of one target entity. Using only the source sentence to generate claims will result in simple claims that are not challenging. But, allowing world knowledge to be incorporated is too unconstrained and will result in claims that cannot be evidenced by this dataset. This is addressed by introducing a dictionary that provides additional knowledge that can be used to increase the complexity of claims in a controlled manner.

Dictionary Additional world knowledge is given to the annotator in a dictionary. This allows for more complex claims to be generated in a structured manner by the annotator. This knowledge may be incorporated into claims or may be needed when labelling whether evidence supports or refutes claims.

Mutation True claims will be distorted or mutated as part of the claim generation workflow. This may be achieved by making the sentence negative, substituting words or

ideas or by making words more or less specific. The annotation system will select which type of mutation will be used.

Requirements:

- Claims must reference the target entity directly and avoid use of pronouns/nominals (e.g. he, she, it, the country)
- Claims must not use speculative/cautious/vague language (e.g. may be, might be, it is reported that)
- True claims should only be facts that can be deduced by information given in the source sentence and dictionary
- Minor variations over the entity name are acceptable: (e.g. Amazon River vs River Amazon)

Examples of true claims:

- Keanu Reeves has acted in a Shakespeare play
- The Assassin's Creed game franchise was launched in 2007
- Prince Hamlet is the Prince of Denmark
- In 2004, the coach of the Argentinian men's national field hockey team was Carlos Retegui

A.1.2 Task 1 (subtask 1) guidelines

The objective of this task is to generate true claims from this source sentence that was extracted from Wikipedia.

- Extract simple factoid claims about entity given the source sentence.
- Use the source sentence and dictionary as the basis for your claims.
- Reference any entity directly (i.e. pronouns and nominals should not be used)

- Minor variations of names are acceptable (e.g. John F Kennedy, JFK, President Kennedy).
- Avoid vague or cautions language (e.g. might be, may be, could be, is reported that)
- Correct capitalisation of entity names should be followed (India, not india).
- Sentences should end with a period.
- Numbers can be formatted in any appropriate English format (including as words for smaller quantities).
- Some of the extracted text might not be accurate. These are still valid candidates for summary. It is not your job to fact-check the information

World Knowledge

- Do not incorporate your own knowledge or beliefs.
- Additional world knowledge is given to the you in the form of a dictionary. Use this to make more complex claims (we prefer using this dictionary instead of your own knowledge because the information in this dictionary can be backed up from Wikipedia)
- If the source sentence is not suitable, leave the box blank to skip.
- If a dictionary entry is not suitable or uninformative, ignore it.

A.1.3 Task 1 (subtask 1) examples

See Tables A.1 and A.2 for examples from the real data.

Entity	India
Source Sentence	It shares land borders with Pakistan to the west; China, Nepal, and Bhutan to the northeast; and Myanmar (Burma) and Bangladesh to the east.
Dictionary	**Bhutan** Bhutan, officially the Kingdom of Bhutan, is a landlocked country in Asia, and it is the smallest state located entirely within the Himalaya mountain range. **China** China, officially the People's Republic of China (PRC), is a unitary sovereign state in East Asia and the world's most populous country, with a population of over 1.381 billion. **Pakistan** Pakistan, officially the Islamic Republic of Pakistan, is a federal parliamentary republic in South Asia on the crossroads of Central and Western Asia.
Claims	- One of the land borders that India shares is with the world's most populous country. (uses information from the dictionary entry for china) - India borders 6 countries. (summarises some of the information in the source sentence) - The Republic of India is situated between Pakistan and Burma. (deduced by Pakistan being West of India, and Burma being to the East)

Table A.1: Task 1 (subtask 1) example: India

Entity	Canada
Source Sentence	Canada is sparsely populated, the majority of its land territory being dominated by forest and tundra and the Rocky Mountains.
Dictionary	**Province of Canada** The United Province of Canada, or the Province of Canada, or the United Canadas was a British colony in North America from 1841 to 1867. **Rocky Mountains** The Rocky Mountains, commonly known as the Rockies, are a major mountain range in western North America. **tundra** In physical geography, tundra is a type of biome where the tree growth is hindered by low temperatures and short growing season.
Claims	- The terrain in Canada is mostly forest and tundra. - Parts of Canada are subject to low temperatures. - Canada is in North America. - In some areas of Canada, it is difficult for trees to grow.

Table A.2: Task 1 (subtask 1) example: Canada

A.1.4 Task 1 (substask 2) guidelines

The objective of this task is to generate modifications to claims. The modifications can be either true or false. You will be given specific instructions about the types of modifications to make.

- Use the original claims and dictionary as the basis for your modifications to facts about entity
- Reference any entity directly (i.e. pronouns and nominals should not be used)
- Avoid vague or cautions language (e.g. might be, may be, could be, is reported that)
- Correct capitalisation of entity names should be followed (India, not india).
- Sentences should end with a period.
- Numbers can be formatted in any appropriate English format (including as words for smaller quantities).
- Some of the extracted text might not be accurate. These are still valid candidates for summary. It is not your job to fact-check the information

Specific guidelines for this screen

- Aim to spend about up to 1 minute generating each claim.
- You are allowed to incorporate your own world knowledge in making these modifications and misinformation.
- All facts should reference any entity directly (i.e. pronouns and nominals should not be used).
- The mutations you produce should be objective (i.e. not subjective) and verifiable using information/knowledge that would be publicly available
- If it is not possible to generate facts or misinformation, leave the box blank.

There are six types of mutation the annotator will be asked to introduce. These will all be given on the same annotation page as all the claim modification types are related.

1. Rephrase the claim so that it has the same meaning
2. Negate the meaning of the claim
3. Substitute the verb and/or object in the claim to alternative from the same set of things
4. Substitute the verb and/or object in the claim to alternative from a different set of things
5. Make the claim more specific so that the new claim implies the original claim (by making the meaning more specific)
6. Make the claim more general so that the new claim can be implied by the original claim (by making the meaning less specific)

It may not always be possible to generate claims for each modification type. In this case, the box may be left blank.

A.1.5 Task 1 (subtask 2) examples

The following example illustrates how given a single source sentence, the following mutations could be made and why they are suitable. For the claim "*Barack Obama toured the UK.*", Figure A.1 shows the relations between objects, and Table A.3 contains examples for each type of mutation.

A.1.6 Task 2 guidelines

The purpose of this task is to identify evidence from a Wikipedia page that can be used to support or refute simple factoid sentences called claims. The claims are generated by humans (as part of the WF1 annotation workflow) from a Wikipedia page. Some claims

Type	Claim	Rationale
Rephrase	President Obama visited some places in the United Kingdom.	Rephrased. Same meaning.
Negate	Obama has never been to the UK before.	Obama could not have toured the UK if he has never been there.
Substitute Similar	Barack Obama visited France.	Both the UK and France are countries
Substitute Dissimilar	Barrack Obama attended the Whitehouse Correspondents Dinner.	In the claim, Barack Obama is visiting a country, whereas the dinner is a political event.
More specific	Barrack Obama made state visit to London.	London is in the UK. If Obama visited London, he must have visited the UK.
More general	Barrack Obama visited a country in the EU.	The UK is in the EU. If Obama visited the UK, he visited an EU country.

Table A.3: Example mutations

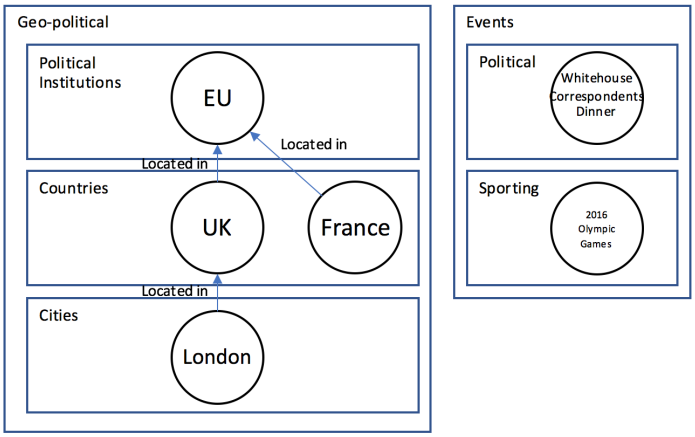


Figure A.1: Toy ontology to be used with the provided examples of similar and dissimilar mutations

are true. Some claims are fake. You must find the evidence from the page that supports or refutes the claim.

Other Wikipedia pages will also provide additional information that can serves as evidence. For each line, we will provide extracts from the linked pages in the dictionary column which appear when you "Expand" the sentence. The sentences from these linked pages that contain relevant supplementary information should be individually selected to record which information is used in justifying your decisions.

Step-by-step guide:

1. Read and understand the claim

2. Read the Wikipedia page and identify sentences that contain relevant information.

3. On identifying a relevant sentence, press the Expand button to highlight it. This will load the dictionary and the buttons to annotate it:

   (a) If the highlighted sentence contains enough information in a definitive statement to support or refute the claim, press the Supports or Refutes button to add your annotation. No information from the dictionary is needed in this case (this includes information from the main Wikipedia page). Then continue annotating from step 2.

   (b) If the highlighted sentence contains some information supporting or refuting the claim but also needs supporting information, this can be added from the dictionary.

      i. The hyperlinked sentences from the passage are automatically added to the dictionary

      ii. If a sentence from the main Wikipedia article is needed to provide supporting information. Click "Add Main Wikipedia Page" to add it to the dictionary.

iii. If the claim or sentence contains an entity that is not in the dictionary, then a custom page can be added by clicking "Add Custom Page". Use a search engine of your choice to find the page and then paste the Wikipedia URL into the box.

iv. Tick the sentences from the dictionary that provide the minimal amount of supporting information needed to form your decision. If there are multiple equally relevant entries (such as a list of movies), then just select the first. Once all required information is added, then press the Supports or Refutes button to add your annotation and continue from step 2.

(c) If the highlighted sentence and the dictionary do not contain enough information to support or refute the claim, press the Cancel button and continue from step 2 to identify more relevant sentences.

4. On reaching the end of the Wikipedia page. Press Submit if you could find information that supports or refutes the claim. If you could not find any supporting evidence, press Skip then select Not enough information

The objective is to find sentences that support or refute the claim.

You must apply common-sense reasoning to the evidence you read but avoid applying your own world-knowledge by basing your decisions on the information presented in the Wikipedia page and dictionary.

As a guide - you should ask yourself:

If I was given only the selected sentences, do I have stronger reason to believe claim is true (supported) or stronger reason to believe the claim is false (refuted). If I'm not certain, what additional information (dictionary) do I have to add to reach this conclusion.

A.1.7 Task 2 examples

What does it mean to Support or Refute a claim

The following count as valid justifications for marking an item as supported/refuted:

Sentence directly states information that supports/refutes the claim or states information that is synonymous/antonymous with information in the claim

Claim: Water occurs artificially
Refuted by: "It also occurs in nature as snow, glaciers ..."

Claim: Samuel L. Jackson was in the third movie in the Die Hard film series.
Supported by: "He is a highly prolific actor, having appeared in over 100 films, including Die Hard 3."

Sentence refutes the claim through negation or quantification

Claim: Schindler's List received no awards.
Refuted by: "It was the recipient of seven Academy Awards (out of twelve nominations), including Best Picture, Best Director..."

Sentence provides information about a different entity and only one entity is permitted (e.g. place of birth can only be one place).

Claim: David Schwimmer finished acting in Friends in 2005.
Refuted by: "After the series finale of Friends in 2004, Schwimmer was cast as the title character in the 2005 drama Duane Hopwood."

Sentence provides information that, in conjunction with other sentences from the dictionary, fulfils one of the above criteria.

Claim: John McCain is a conservative.
Refuted by: "He was the Republican nominee for the 2008 U.S. presidential election." AND "The Republican Party's current ideology is American conservatism, which contrasts with the Democrats' more progressive platform (also called modern liberalism)."

Adding primary Wikipedia page to dictionary In the case where the claim can be supported from multiple sentences from the main Wikipedia page, information the main Wikipedia page should be added to the dictionary to add supporting information. This is because each line that is annotated in the left column for the main Wikipedia page is stored independently.

Claim: George Washington was a soldier, born in 1732.

Wikipedia page: George Washington

> Sentence 1: George Washington was an American politician and soldier who served as the first President of the United States from 1789 to 1797 and was one of the Founding Fathers of the United States.
>
> Sentence 2: In 1775, the Second Continental Congress commissioned him as commander-in-chief of the Continental Army in the American Revolution.
>
> Sentence 3: The Gregorian calendar was adopted within the British Empire in 1752, and it renders a birth date of February 22, 1732.

Sentence 1 contains enough information to wholly support the claim without the need for any additional information.

Sentence 2 and 3 contain partial information that can be combined. Expand sentence 2 and click add main Wikipedia page to add the Wikipedia page to add George Washington to the dictionary. Sentence 3 can now be added to dictionary to support the claim.

The order of the sentences doesn't matter (selecting sentence 2+3 is the same as adding sentence 3+2) because we sort the sentences in document order. This means that you only need to annotate this once.

If you attempt to add the main Wikipedia page to the dictionary from sentence 3 having already used it for sentence 2, the system will warn you that you are making a duplicate annotation.

Adding custom pages

You may need to add a custom page from Wikipedia to the dictionary. This may happen in cases where the claim discusses an entity that was not in the original Wikipedia page

Claim: Colin Firth is a Gemini. In Original Page: "Colin Firth (born 10 September 1960)... " Requires Additional Information from Gemini: "Under the tropical zodiac, the sun transits this sign between May 21 and June 21."

Tense The difference in verb tenses that do not affect the meaning should be ignored.

Claim: Frank Sinatra is a musician Supported: He is one of the best-selling music artists of all time, having sold more than 150 million records worldwide.

Claim: Frank Sinatra is a musician Supported: Francis Albert Sinatra was an American singer

Skipping

There may be times where it is appropriate to skip the claim by pressing the Skip button:

- The claim cannot be verified using the information with the information provided:
  - If the claim could potentially be verified using other publicly available information. Select Not Enough Information
  - If the claim can't be verified using any publicly available information (because it's ambiguous, vague, personal or implausible) select The claim is ambiguous or contains personal information
  - If the claim doesn't meet the guidelines from WF1, select: The claim doesn't meet the WF1 guidelines
- The claim contains typographical errors, spelling mistakes, is ungrammatical or could be fixed with a very minor change

– Select The claim has a typo or grammatical error

Keep in mind that clicking Not Enough Information or The claim is ambiguous or contains personal information is still very useful feedback for the AI systems. They need examples of what a verifiable claim looks like, and negative examples are as useful (if not more so) than positive ones!

A.1.8 Task 2 additional guidelines

After conferring with myself and Christos, the annotators expanded the guidelines to include common case that were not explicitly covered in the guidelines:

1. Any claims involving "many", "several", "rarely", "barely" or other indeterminate count words are going to be ambiguous and can be flagged.

2. Same goes for "popular", "famous", and "successful (for people, for works we can assume commercial success)".

3. We cannot prove personhood for fictional characters like we can for real people (dogs and cats can be authors, actors, and citizens in fiction).

4. If a claim is "[Person] was in [film]", the only way to refute it would be (a) if they were born after it was released, or (b) their acting debut is mentioned and occurs a realistically long enough amount of time after the film's release (at least a few years).

5. A list of movies that someone was in or jobs a person held is not necessarily exclusive, we cannot refute someone being a lawyer because the first sentence of their wiki article says they were an actor.

6. A person is not their roles, if a claim is something like "Tom Cruise participated in a heist in Mission Impossible 3", we cannot prove it, because Ethan Hunt did that, not Tom Cruise.

7. Our workflow is time-insensitive, but if a claim tags something to a time period, we can treat it as such. "Neil Armstrong is an astronaut" can be supported, but "Neil Armstrong was an astronaut in 2013" can be refuted, because he was dead at the time.

8. If someone won 5 Academy Awards, they won 3 Academy Awards. Similarly, if they won an Academy Award, they were nominated for an award.

9. Multiple citizenships can exist.

10. If a claim says "[Person] was in [film] in 2009", then the film's release date can support it. If the claim is "[Person] acted in [film] in 2009", filming dates or release dates can prove it.

11. Flag anything related to death, large-scale recent disasters, and controversial religious or social statements.

A.2 FEVER annotation interface screenshots

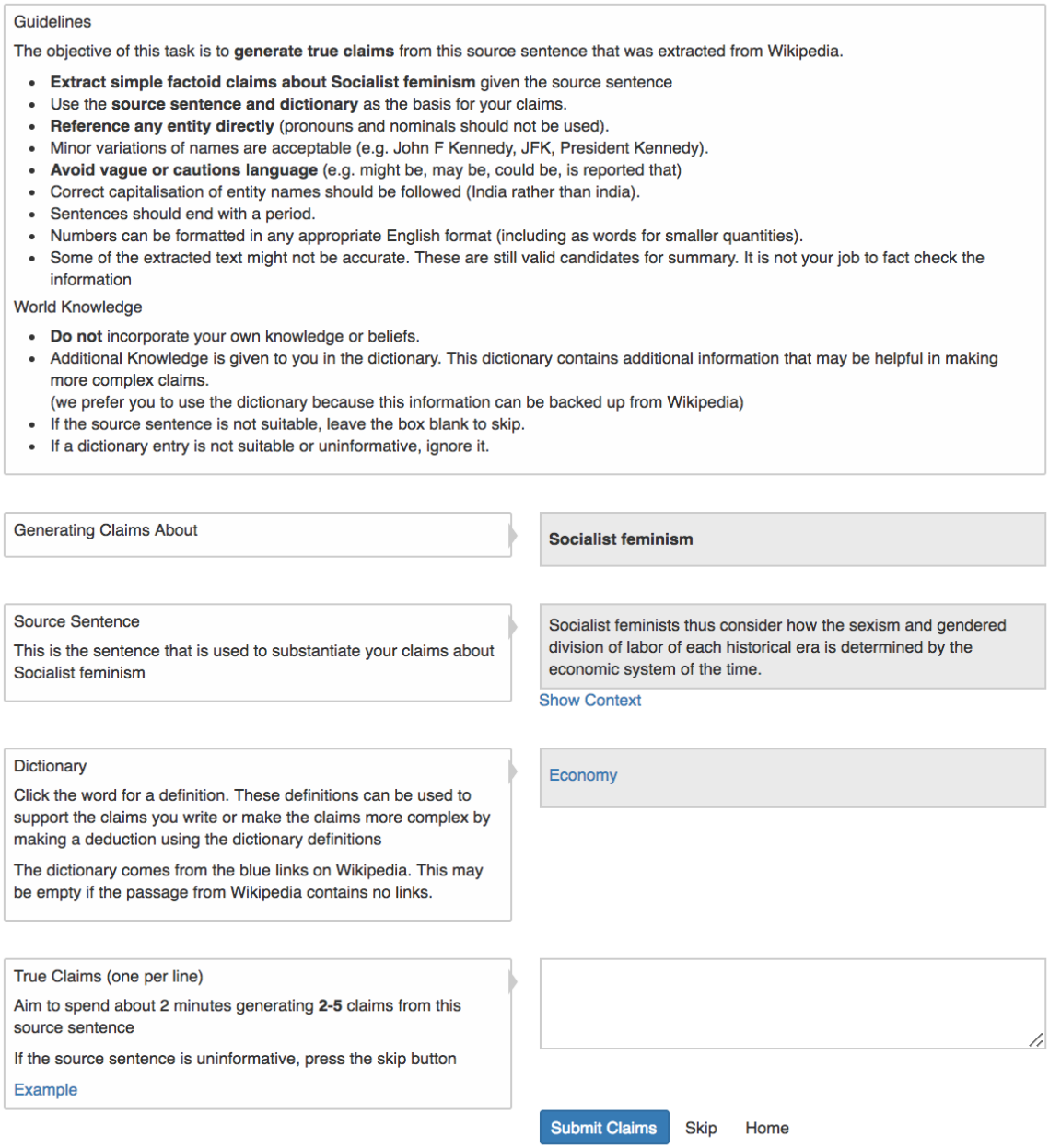


Figure A.2: Screenshot of the claim generation annotation task: Generating true claims by extracting facts from sentence sampled from Wikipedia.

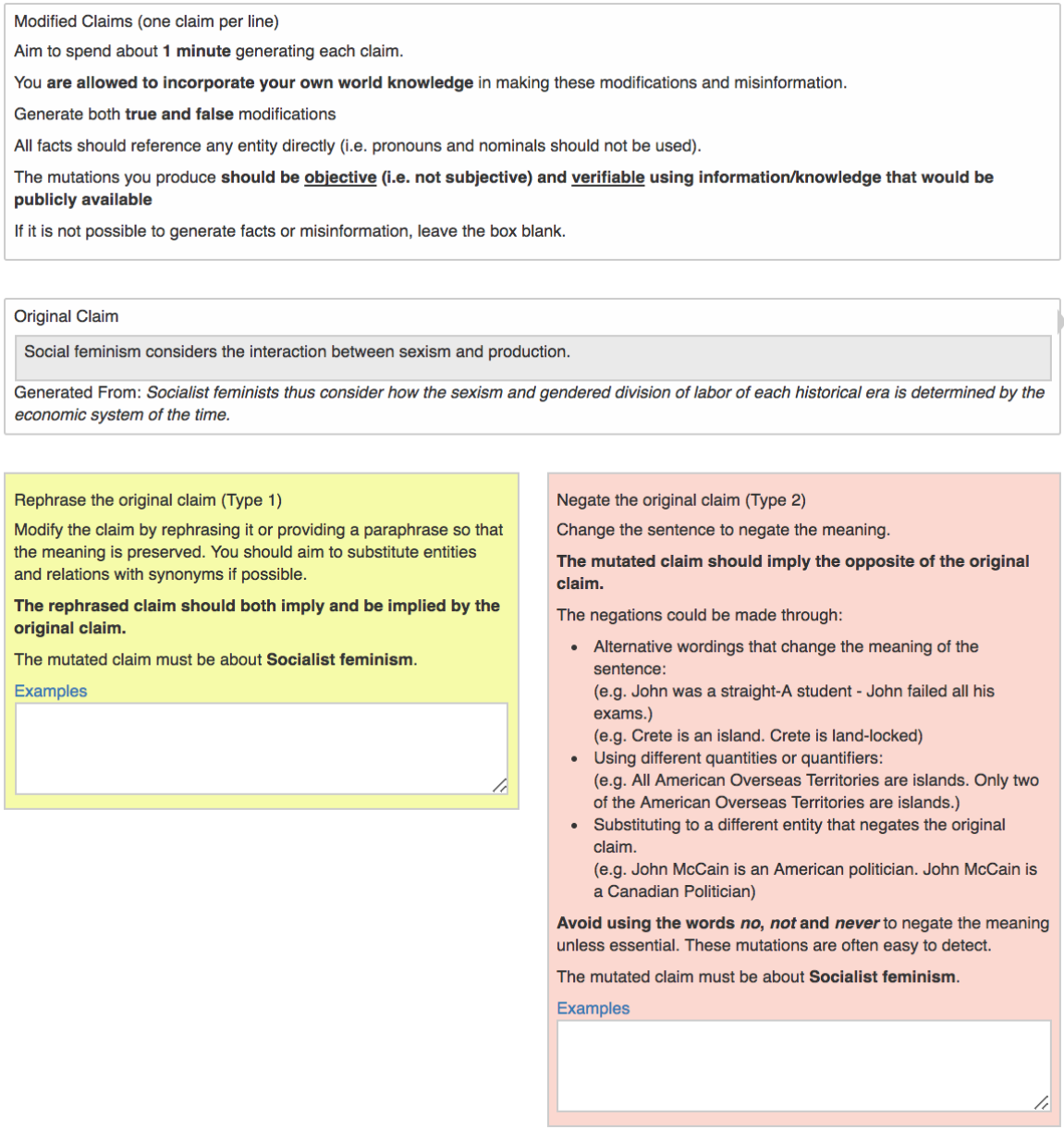


Figure A.3: Screenshot of the claim generation annotation task: Mutating claims by making meaning altering changes following 6 different prompts (1 of 2).

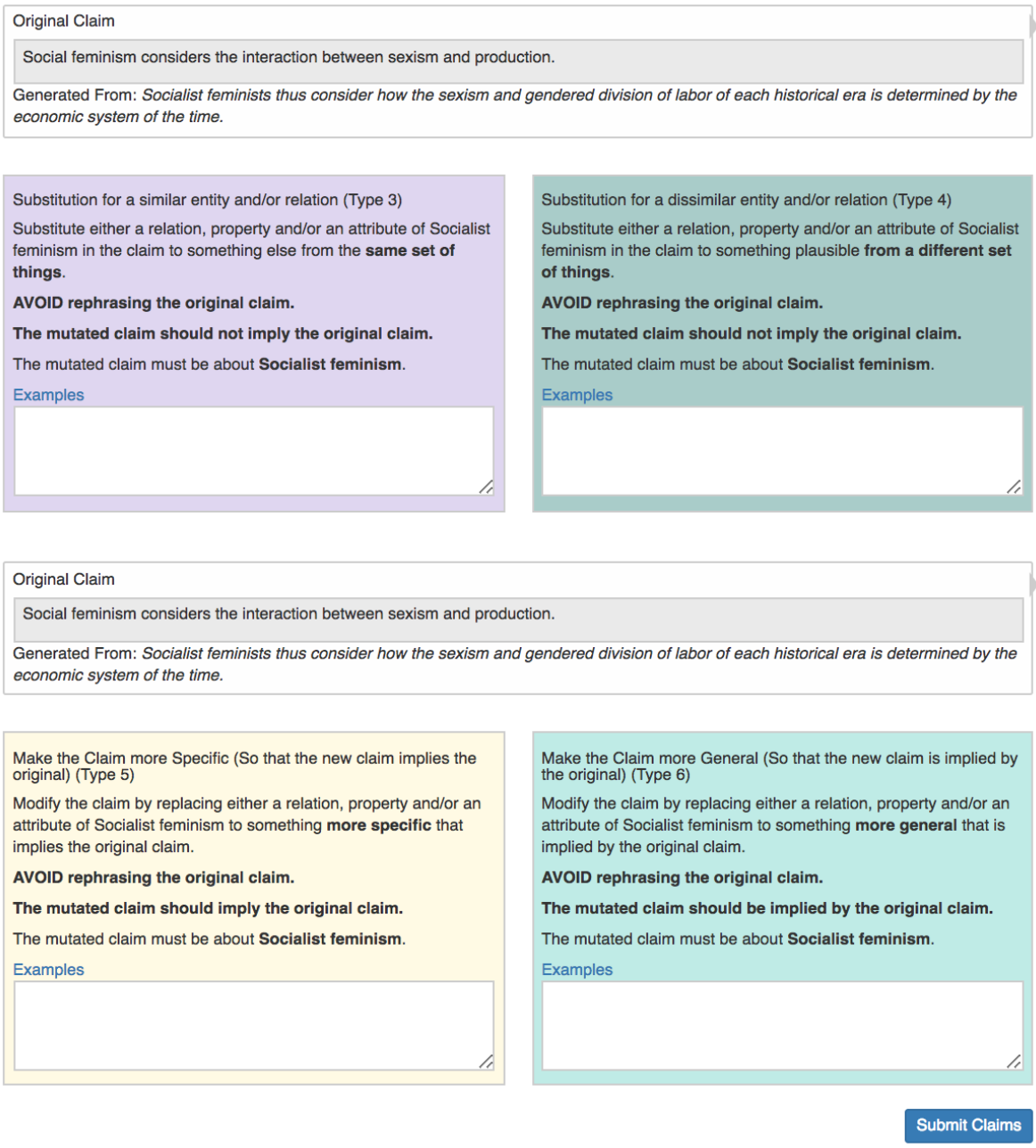


Figure A.4: Screenshot of the claim generation annotation task: Mutating claims by making meaning altering changes following 6 different prompts (2 of 2)

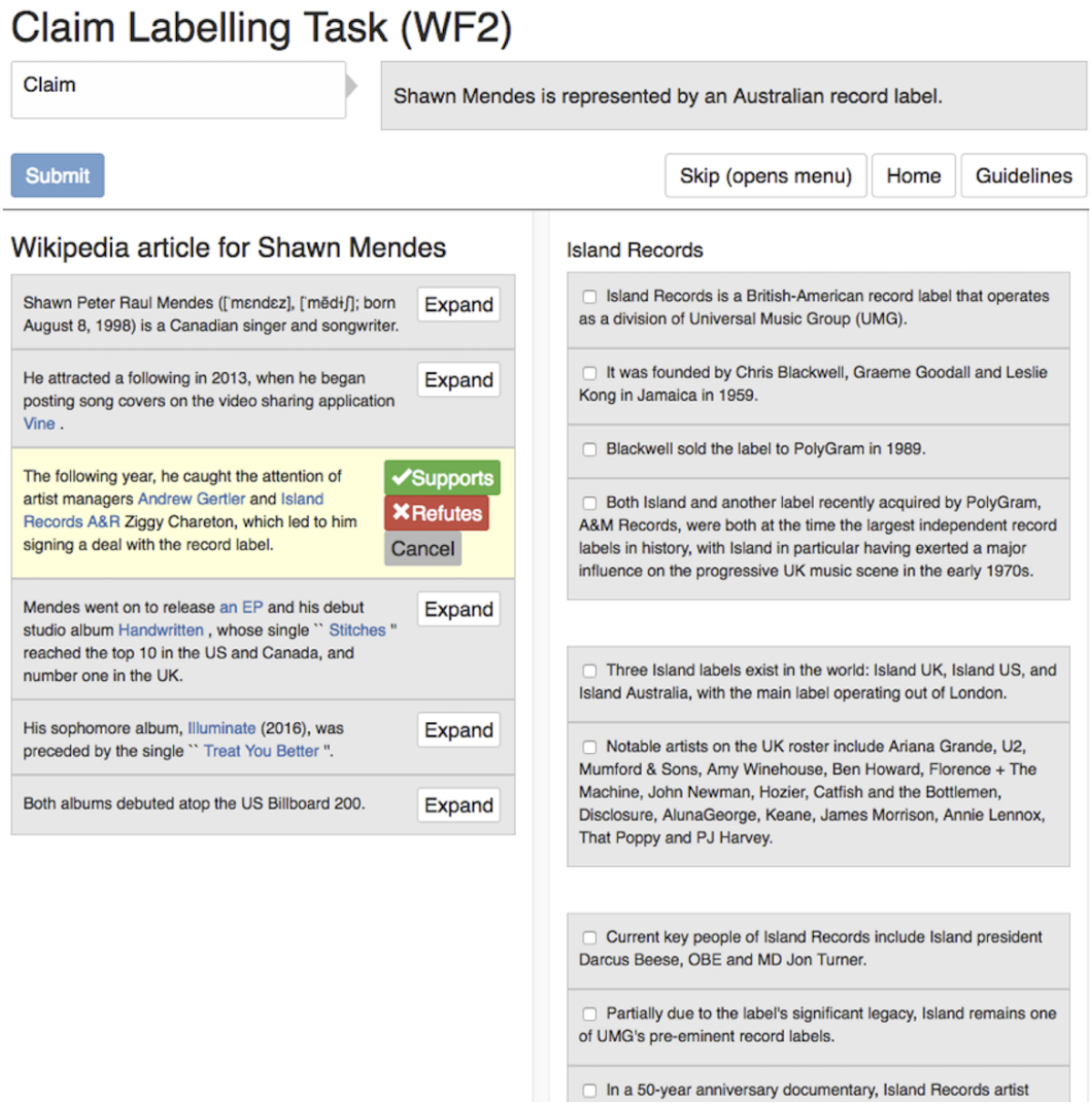


Figure A.5: Screenshot of the claim labelling annotation task: evidence from a Wikipedia page (left) is selected and combined with linked pages (right) to form labelled evidence groups for a given claim (top).

A.3 FEVER model hyperparameter choices

Widely accepted hyper-parameter choices are used for the DA, ESIM, BERT and RoBERTa models.

Decomposable attention

- Pre-trained embeddings: GloVe, 300d, frozen
- Encoder dimension: 200
- Dropout: 0.2 (probability of drop)
- Optimizer: Adagrad
- Gradient clipping: 5.0
- Batch Size: 64
- Learning Rate: 1e-2
- Learning Rate Schedule: none
- Number of Epochs: 140
- Early Stopping: Patience 20

ESIM

- Pre-trained embeddings: GloVe, 300d, trainable
- Encoder dimension: 300, bidirectional
- Dropout: 0.5
- Optimizer: Adam
- Gradient Norm: 10.0
- Batch Size: 64
- Learning Rate: 0.0004
- Learning Rate Schedule: reduce on plateau, patience 0, factor 0.5
- Number of Epochs: 75
- Early Stopping: Patience 5

BERT+RoBERTa

- Embedding dimension: 768
- Optimizer: AdamW
- Gradient Norm: 10.0
- Batch Size: 8
- Learning Rate: 0.0002
- Weight Decay: 0.1
- Learning Rate Schedule: slanted triangular, cut frac 0.06
- Number of Epochs: 5
- Early Stopping: Patience 0

TF-IDF

- Hash size: 24 bit
- NGram size: 2

BM25

- Hash size: 24 bit
- NGram size: 2
- b: 0.75
- k_1: 1.2

GENRE Using TRIE and model weights from `https://github.com/facebookresearch/genre`

A.4 FEVER common claim patterns

Bigram	Occurrences	Proportion (%)
is a	14254	12.98
in the	6868	6.25
of the	5239	4.77
is an	4007	3.65
in a	3560	3.24
film .	3289	3.00
was born	3173	2.89
was a	2877	2.62
an american	2534	2.31
is the	2488	2.27
a film	2305	2.10
was in	2068	1.88
born in	1987	1.81
the united	1671	1.52
was released	1654	1.51
film)	1571	1.43
united states	1536	1.40
actor .	1511	1.38
on the	1491	1.36
is in	1468	1.34
is not	1466	1.34
was the	1459	1.33
of a	1418	1.29
starred in	1336	1.22
has a	1282	1.17
Total	*72512*	*66.03*

Table A.4: Most common bigrams in the FEVER training set that were used to inspire the generation of the adversarial rules.

A.5 Adversarial rules

List of regular expressions matching claims and generators for claims. Replacements prefixed with parenthesis are conditionally applied for claims with the matching label, or matching named entity tags in the claim.

`(.+) is a (.+) :`

Label preserving modifications:

- There exists a $0 called $1.
- There exists a $0 that goes by the name of $1.
- There is a $0 called $1.
- (NON PERSON)There exists a $0, it goes by the name of $1.
- (PERSON)There is a $0, they are called $1.
- (NON PERSON)There is a $0, it is called $1.

Simple negations:

- $0 is not a $1.
- $0 is definitely not a $1.
- $0 is certainly not a $1.

Complex negations:

- There does not exist a $0 called $1.
- There does not exist a $0 that goes by the name of $1.
- There is not a $0 called $1.
- There is not a $0 that goes by the name of $1.

`(.+) is an (.+) :`

Label preserving modifications:

- There exists an $0 called $1.

- There exists an $0 that goes by the name of $1.
- There is an $0 called $1.
- (NON PERSON)There exists an $0, it goes by the name of $1.
- (PERSON)There is an $0, they are called $1.
- (NON PERSON)There is an $0, it is called $1.

Simple negations:

- $0 is not an $1.
- $0 is definitely not an $1.
- $0 is certainly not an $1.

Complex negations:

- There does not exist an $0 called $1.
- There does not exist an $0 that goes by the name of $1.
- There is not an $0 called $1.
- There is not an $0 that goes by the name of $1.

`(.+) is (?:a|an) (.+) :`

Complex negations:

- There exists no $0 called $1.

`(.+) was a (.+) :`

Label preserving modifications:

- There existed a $0 called $1.
- There existed a $0 that went by the name of $1.
- There was a $0 called $1.
- (NON PERSON)There existed a $0, it was called $1.
- (NON PERSON)There existed a $0, it went by the name of $1.
- (NON PERSON)There was a $0, it was called $1.

Simple negations:

- $0 wasn't a $1.
- $0 definitely was not a $1.
- $0 was certainly not a $1.

`(.+) was an (.+) :`

Label preserving modifications:

- There existed an $0 called $1.
- There existed an $0 that went by the name of $1.
- There was an $0 called $1.
- (NON PERSON)There existed an $0, it was called $1.
- (NON PERSON)There existed an $0, it went by the name of $1.
- (NON PERSON)There was an $0, it was called $1.

Simple negations:

- $0 wasn't an $1.
- $0 definitely was not an $1.
- $0 was certainly not an $1.

Complex negations:

- There did not exist an $0 called $1.
- There did not exist an $0 that goes by the name of $1.
- There was not an $0 called $1.
- There was not an $0 that went by the name of $1.

`(.+) was (?:a|an) (.+) :`

Complex negations:

- There existed no $0 called $1.

`(.+) (?:was|is)?  directed by (.+)  :`

Label preserving modifications:

- There is a movie called $0 which is directed by $1.
- $1 is the director of $0.
- $1 was the director of $0.
- There is a director, $0, who was involved in the production of $1.
- There is a person involved in the movie industry, $0, who was the director of $1.

Simple negations:

- $0 is not directed by $1.
- $0 isn't directed by $1.
- $0 is definitely not directed by $1.
- $0 is certainly not directed by $1.

Complex negations:

- There is a movie called $0 which is not directed by $1.
- There is a movie called $0 which wasn't directed by $1.
- There is a director, $0, who was not involved in the production of $1.
- There is a person involved in the movie industry, $0, who was not the director of $1.

`(.+) (?:starred|stars) in (.+)  :`

Label preserving modifications:

- There is a person, $0, that starred in $1.
- There is a person, $0, that took a leading acting role in $1.

Simple negations:

- $0 did not star in $1.
- $0 didn't star in $1.

Complex negations:

- There is a person, $0, that did not star in $1.
- There is a person, $0, that did not take a leading acting role in $1.
- There is a person, $0, that did not appear in $1.
- There is a person, $0, that had no role in $1.

`(.+) an American (.+) :`

Label preserving modifications:

- $0 $1 that originated from the United States.

Simple negations:

- (SUPPORTED)$0 a Canadian $1.
- (SUPPORTED)$0 a French $1.

Complex negations:

- $0 $1 that originated from outside the United States.

`(.+) (?:was|is) born (?:in|on)?  (.+) :`

Label preserving modifications:

- (PLACE,NOT TIME)There exists a place, $1, that is the birthplace of the person $0.
- (PLACE,NOT TIME)There exists a place, $1, that is where the person $0 started living.
- (TIME,NOT PLACE)$1 is the approximate time at which the person $0 was born.
- (TIME,NOT PLACE)$1 is the approximate time at which the person $0 started living.

Simple negations:

- $0 was not born on $1.

- (SUPPORTED)$0 was never born.

Complex negations:

- There exists a place, $1, that was not the birthplace of the person $0.
- There exists a place, $1, that is not where the person $0 started living.
- $0 was born in some other place than $1.
- $1 is some place other than where the person $0 was born.
- $1 is not the approximate time at which the person $0 was born.
- $1 is not the approximate time at which the person $0 started living.
- $0 was born at some other time than $1.
- $1 is some other time than when $0 was born.

`(.+) died (?:in|on) (.+) :`

Label preserving modifications:

- (PLACE,NOT TIME)There exists a place, $1, that is the place where the person $0 became deceased.
- (PLACE,NOT TIME)There exists a place, $1, that is the place where the person $0 died.
- (PLACE,NOT TIME)There exists a place, $1, that is the place where the person $0 took their final breath(TIME,NOT PLACE)$1 is the approximate time at which the person $0 became deceased.
- (TIME,NOT PLACE)$1 is the approximate time at which the person $0 died.
- (TIME,NOT PLACE)$1 is the approximate time at which the person $0 took their final breath.

Simple negations:

- $0 did not die on $1.
- (SUPPORTED)$0 never died.
- (SUPPORTED)$0 has not died.

Complex negations:

- $1 is somewhere other than the place where the person $0 became deceased.
- There exists a place, $1, that is not the place where the person $0 died.
- There exists a place, $1, that is not the place where the person $0 took their final breath.
- $1 is some other time than when the person $0 became deceased.
- $1 is not the when the person $0 died.
- $1 is not the approximate time at which the person $0 took their final breath.

A.6 Adversarial instance error coding procedure

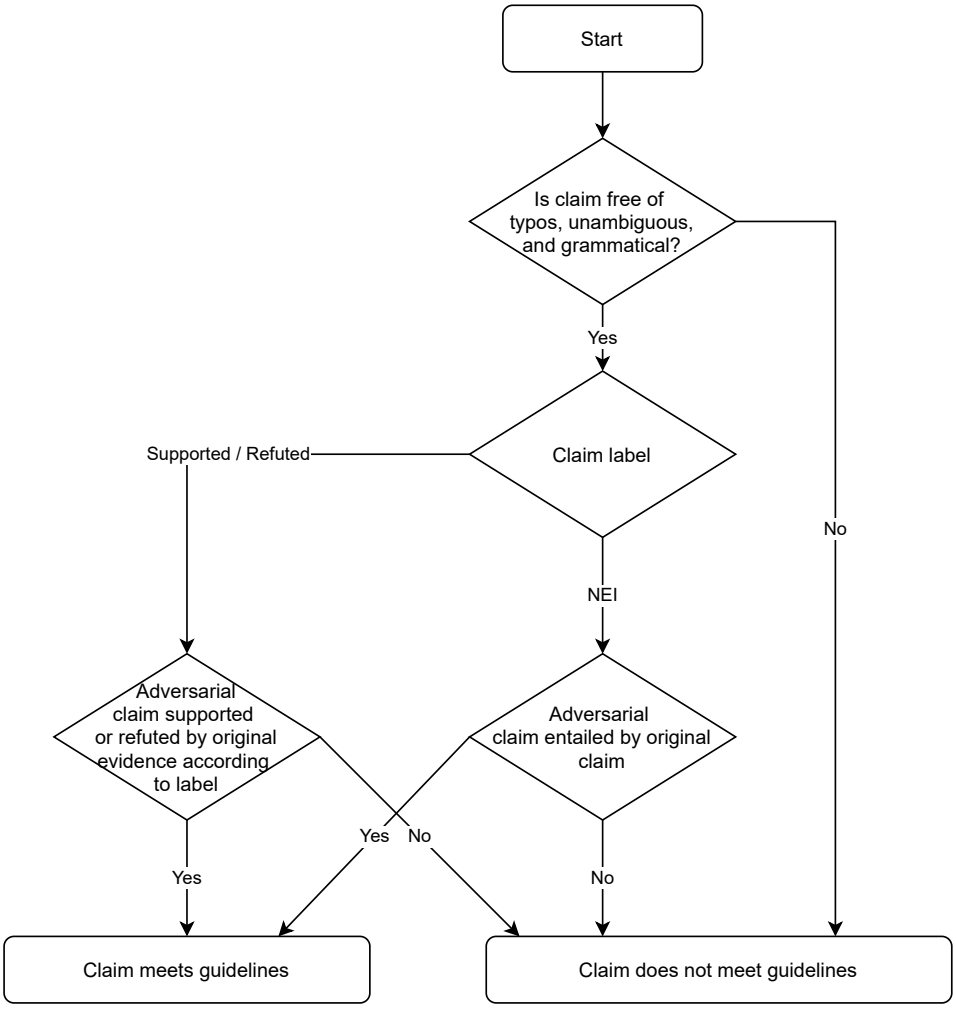


Figure A.6: Error coding procedure for adversarial claims. The definitions for supported/refuted, entailment and ambiguity use the same definitions as in Chapter 3.

A.7 Text-pair bias hyperparameters

A.7.1 Base models

For the base-models, the default hyperparemters in AllenNLP are used for the ESIM, BERT and RoBERTa models.

ESIM

- Embedding dimension: 300, bidirectional
- Dropout: 0.5
- Optimizer: Adam
- Gradient Norm: 10.0
- Batch Size: 64
- Learning Rate: 0.0004
- Learning Rate Schedule: reduce on plateau, patience 0, factor 0.5
- Number of Epochs: 75
- Early Stopping: Patience 10

BERT+RoBERTa

- Embedding dimension: 768
- Optimizer: AdamW
- Gradient Norm: 10.0
- Batch Size: 8
- Learning Rate: 0.0004
- Learning Rate Schedule: slanted triangular, cut frac 0.06
- Number of Epochs: 5
- Early Stopping: Patience 0

A.7.2 Fine-tuning without EWC

ESIM

- FT Learning Rate: 0.0002
- FT Epochs: 8

BERT

- FT Learning Rate: 0.000004
- FT Epochs: 6

RoBERTa

- FT Learning Rate: 0.000004
- FT Epochs: 7

A.7.3 Fine-tuning using EWC

ESIM

- FT Learning Rate: 0.0002
- FT Epochs: 5
- EWC: 10000000

BERT+RoBERTa

- FT Learning Rate: 0.000004
- FT Epochs: 6
- EWC: 10000000

A.7.4 Unsupervised bias mitigation

Hyperparameters: β controls the weight update for the hypothesis-only model and γ controls the modulation of hypothesis-only model in the loss function.

- POE (BERT) $\beta = 0.4$
- POE (ESIM) $\beta = 0.05$
- DFL (BERT) $\beta = 0.4, \gamma = 0.6$
- DFL (ESIM) $\beta = 0.05, \gamma = 2.0$

A.7.5 Search bounds for fine-tuning

Approximately 30 configurations were considered (Cartesian product of LR+EWC). The best performing system was selected through max accuracy on the FT cross validation dataset through 5 fold cross validation.

- EWC $\{10^6, 2 \cdot 10^6, 4 \cdot 10^6, 8 \cdot 10^6, 10^7, 2 \cdot 10^7, 4 \cdot 10^7, 6 \cdot 10^7, 8 \cdot 10^7, 10^8\}$
- Fine-tuning learning rate (ESIMs): $\{0.0002, 0.0004, 0.0006\}$
- Fine-tuning learning rate (Transformer): $\{0.000002, 0.000004, 0.000006\}$
- Epochs: Up to 10 epochs cross-validating on the FT-training dataset, selecting the highest performing.

For the Pareto frontiers, the average FT-test accuracy is reported from 5 random initialisations with the same hyperparameter choices.

A.7.6 Search bounds for instance weighting bias mitigation

The same range of values published by Mahabadi et al. (2020) were used. For DFL, a grid search over every pair of values was performed, totalling 30 trials.

$\gamma \in \{0.02, 0.05, 0.1, 0.6, 2.0, 4.0, 5.0\}$

$\beta \in \{0.05, 0.2, 0.4, 0.8, 1.0\}$

A.7.7 Stress test sizes

Following the evaluation of (Liu et al., 2019a), the number of instances sampled from the stress test (between 1500-9800) is varied. To plot Figures 5.5 and 5.6, the values used are: 10, 25, 50, 75, 100, 250, 400, 500, 600, 700, 800, 900, 1000 instances.