eman ta zabal zazu

Universidad del País Vasco

Euskal Herriko Unibertsitatea

university of groningen

A Semantics-Aware Approach to Automated Claim Verification

Author: Blanca Calvo Figueras

Advisors:

Dr. Rodrigo Agerri, Dr. Montse Cuadros & Prof. Malvina Nissim

Erasmus Mundus Language and Communication Technologies

Final Thesis

August 2021

Departments: Computer Systems and Languages, University of the Basque Country; Faculty of Arts, University of Groningen.

Abstract

The influence of fake news in the perception of reality has become a mainstream topic in the last years due to the fast propagation of miss-leading information, which has been enhanced by social media. To contribute to the fight against misinformation, researchers have proposed to develop automated solutions. The task of automated claim verification consists in assessing the truthfulness of a claim by finding evidence about its veracity. Datasets with synthetic claims have been developed to train models that perform this task. However, naturally-occurring claims are usually semantically more complex than synthetic claims. In this work, we test if the use of explicit semantic structures can help with the task of claim verification. We integrate Semantic Role Labels and Open Information Extraction structures to a BERT model, showing some improvement on the performance of the task. Additionally, we perform some explainability tests which show that the semantically-enriched model is better at handling complex cases, such as sentences in passive form or with multiple propositions.

Contents

List of Figures

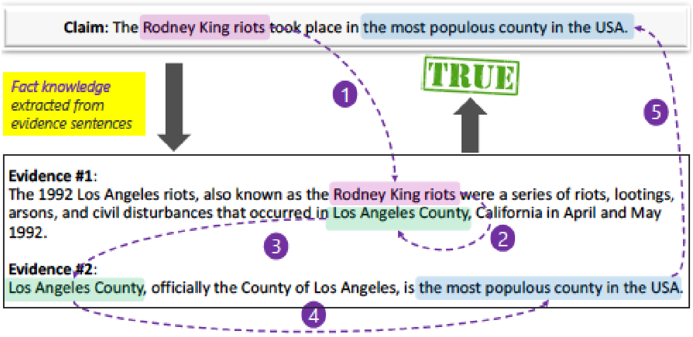
List of Tables

1 Introduction

The spread of fake news can influence the view that people have on reality (Zubiaga et al., 2018), which is why fact-checkers have been fighting misinformation by assessing the veracity of factual content. However, the ever faster spread of information requires for a more automated solution (Zhou and Zafarani, 2020; Oshikawa et al., 2020). Thus, researchers in Natural Language Processing (NLP) have proposed the task of *automated claim verification*: given a claim, a model should be able to look for evidence in order to infer whether it can be supported, refuted, or the information is just not available (Thorne et al., 2018). In general, it is considered that the inference part of this task requires reasoning over sentences with complex semantics (Thorne et al., 2019).

The task of claim verification has been approximated by creating datasets which include a set of claims and a ground-truth knowledge database. Each claim is given a truth-label according to the information available in the database. The most common benchmark for claim verification is FEVER (Thorne et al., 2018), which has 185k claims synthetically generated from a Wikipedia database. In this task, systems should find the articles where the information is located and then select the sentences that are relevant, which we call evidences. Given the right evidences, the inference model should be able to reason whether the claim can be supported or not.

Let us take, for instance, the claim *The Rodney King riots took place in the most populous county in the USA*. In Figure 1, the system has already found two evidences that contain information regarding this claim: one about the *The Rodney King Riots* and one about *Los Angeles County*. Now, the inference model should be able to understand that the entity *Rodney King riots* in the claim is also mentioned in Evidence #1, and that according to this evidence it happened in the place *Los Angeles County*. It should then understand that *Los Angeles County* is the same entity mentioned in Evidence #2, which does happen to be *the most populous county in the USA*. With all the previous information, the model should conclude that the claim can be supported. This is not a trivial task.



Claim: The Rodney King riots took place in the most populous county in the USA.

Evidence #1:
The 1992 Los Angeles riots, also known as the Rodney King riots were a series of riots, lootings, arsons, and civil disturbances that occurred in Los Angeles County, California in April and May 1992.

Evidence #2:
Los Angeles County, officially the County of Los Angeles, is the most populous county in the USA.

Figure 1: Example of the reasoning needed in FEVER from Zhong et al. (2020)

While the FEVER dataset has been used to develop many models, concerns have been raised that the FEVER dataset does not account for the complexity of naturally-occurring claims. Previous work pointed out that the sentences that human fact-checkers encounter are usually semantically more complex, and require more temporal and numerical reasoning (Thorne et al., 2019). This suggests that the next steps towards automated claim verification should focus on developing systems that are able to account for semantic complexity, instead of relying on shallow linguistic cues.

Indeed, recent work on the FEVER dataset has focused on improving the reasoning process by structuring the evidences as graphs and integrating semantic information (Zhou et al., 2019; Zhong et al., 2020). Taking those experiments as a starting point, we want to improve the inference part of a claim verification system so that it can reason better through semantically-complex sentences. In this work, we propose using explicit semantic information to train a model that is able to infer whether a claim is truthful or not.

Thus, the main objective of this work is to evaluate the effects of incorporating semantic knowledge in the inference module of automated claim verification. To this goal, we set several sub-objectives:

1. To annotate a portion of three claim verification datasets to confirm the semantic complexity of naturally-occurring claims.

2. To develop a strong baseline for fact-checking based on recent deep learning architectures and large pre-trained language models.

3. To incorporate different types of semantic information into the baseline.

4. To evaluate the adequacy of the various types of semantic information for automated fact-checking.

5. To assess the linguistic capabilities of the semantically enriched systems by performing tests that make the models more explainable.

The semantic information we use in this work is Semantic Role Labels (SRL, Palmer et al. (2005)) and Open Information Extraction (OpenIE, Etzioni et al. (2008)). In our experiments, these semantic structures are used as additional input to the BERT contextual word embeddings (Devlin et al., 2019). We integrate this information using the SemBERT architecture presented in Zhang et al. (2020a).

Our annotations show that semantic complexity is common in naturally-occurring claims. The main finding of this work is that semantic information does have a positive contribution to the task of automated claim verification. We encounter that, when comparing our semantically enriched model to the baseline, the new model is able to better understand sentences with multiple propositions or written in passive form. We also observe that BERT is already a strong baseline for this task, and that SRL provides more helpful information than OpenIE. All the code from our experiments is open and can be found on our Github respoitory.[1]

[1] https://github.com/BlancaCalvo/Claim-Verification-FakeNews

In the following sections we discuss the state-of-the-art of NLP approaches to fake news, and in particular to claim verification (Section 2), we describe the resources that are used in this work (Section 3), we explain the experiments that we performed (Section 4), and we evaluate the results of our best model testing its linguistic capabilities (Section 5). Finally, we conclude by summarising our observations and pointing at future work (Section 6).

2 Background

The challenge of automated claim verification involves multiple tasks of Natural Language Processing (NLP). In this section, we are going to review the existing literature regarding these different tasks. In the first place, we are going to focus on the concept of fake news and the NLP-based approaches to deal with this issue (Section 2.1). Next, we are going to introduce the task of claim verification (Section 2.2). Then, we are going to highlight the complexities of developing a system for claim verification, present the datasets that have been created to approximate the task, and the state-of-the-art systems (Section 2.3). Finally, we are going to introduce the current practices in language and knowledge representation and note some of its limitations, such as the lack of semantic structure and the issues with explainability (Section 2.4).

2.1 Fake News in NLP

In its broader sense, fake news is defined as a news article or message published through media that carries false information (Kshetri and Voas, 2017; Zhou and Zafarani, 2020). Taking this definition, fake news include disinformation (intentionally false information), misinformation (unintentionally false information) and satire (false information for humorous purposes).

The creation and distribution of false information is not a new phenomena. However, the raise of online social networks as the main media for information propagation has changed the nature of this issue (Hermida, 2010). The absence of control systems and fact-checking in social media has created a prolific environment for the spread of false information. This information arrives to a large number of users in a short time, thus greatly influencing the perception of real world events (Zubiaga et al., 2018). Studies have shown that fake news spread faster in social media than factual news (Vosoughi et al., 2018).

Evaluating the impact of the massive propagation of fake news has been a major research goal in controversial events, such as the US Presidential elections of 2016 (Allcott and Gentzkow, 2017; Grinberg et al., 2019), the Brexit referendum (Bastos and Mercea, 2019), or the COVID-19 pandemic (Alam et al., 2021). These studies have shown that fake news not only disseminate false information, but also promote panic, racism, xenophobia, fake cures, and mistrust in the authorities (Alam et al., 2021).

Fact-checking platforms have been doing great efforts to prevent the propagation of fake news, both through gathering professional fact-checkers and by using crowd-sourcing (Zhou and Zafarani, 2020). However, the current scale of distribution of fake news has put the focus on finding automated solutions to tackle the issue (Oshikawa et al., 2020). These approaches have come from Data Mining, Computational Social Science, Cultural Analytics, and Natural Language Processing (Su et al., 2020). In section 2.1.2, we give an overview of NLP approaches to fake news detection.

2.1.1 Types of Fake News

Deceptive news can get a lot of different shapes depending on their authenticity (if the facts reported are true or not), their intention (if they are created with the purpose of lying, misleading or entertaining its readers), and whether the information reported is news or not (Zhou and Zafarani, 2020). As follows, we define the different types of false information that are usually studied under the broader field of fake news.

- Disinformation. News that are intentionally false and are spread deliberately for some malicious purpose.
- Misinformation. News that contain unintentionally false information because they are created or distributed without a proper fact-checking process.
- Satire. News that are intentionally false and are created for humorous purposes.
- Clickbait. Consists in exaggerating information and under-delivering it. This is often done using controversial headlines that do not always agree with the content of the news article.
- Rumour. This is an unverified claim, which is made by users on social media platforms and can potentially spread beyond their private network.
- Biased-reporting. Consists in reporting news using only some of the facts to serve an agenda.

In the task of claim verification we mainly deal with the first two types of fake news: disinformation and misinformation.

2.1.2 NLP approaches to Fake News Detection

There are four main approaches that NLP researchers have used to automatically detect fake news: using style-based features as proxies to fake news (Zhou et al., 2020; Schuster et al., 2020); building knowledge-based systems to evaluate the factuality of claims (Thorne et al., 2018; Augenstein et al., 2019); observing common propagation patterns of deceptive news (Shao et al., 2020; Pastor-Galindo et al., 2020); and using source-based features that focus on the credibility of the publisher/spreader (Popat et al., 2018).

The great interest of the NLP community on the topic of fake news is evidenced by the multiple shared tasks proposed in the last years. These tasks have focused on different aspects of fake news, such as stance detection and hyperpartisan news detection (Hanselowski et al., 2018a; Mohammad et al., 2016), detecting fake news spreaders (Rangel et al., 2017, 2018, 2020; Wiegmann et al., 2019), evaluating the check-worthiness of claims (Barron-Cedeno et al., 2020; Elsayed et al., 2019; Nakov et al., 2018), and assessing the veracity of claims (Thorne et al., 2018; Barron-Cedeno et al., 2020; Jiang et al., 2020; Wadden et al., 2020; Elsayed et al., 2019; Nakov et al., 2018). Previous approaches to stance detection, detecting fake news spreaders and evaluating check-worthiness will be explained in the following paragraphs. The task of claim verification will be described in detail in sections 2.2 and 2.3.

Stance Detection

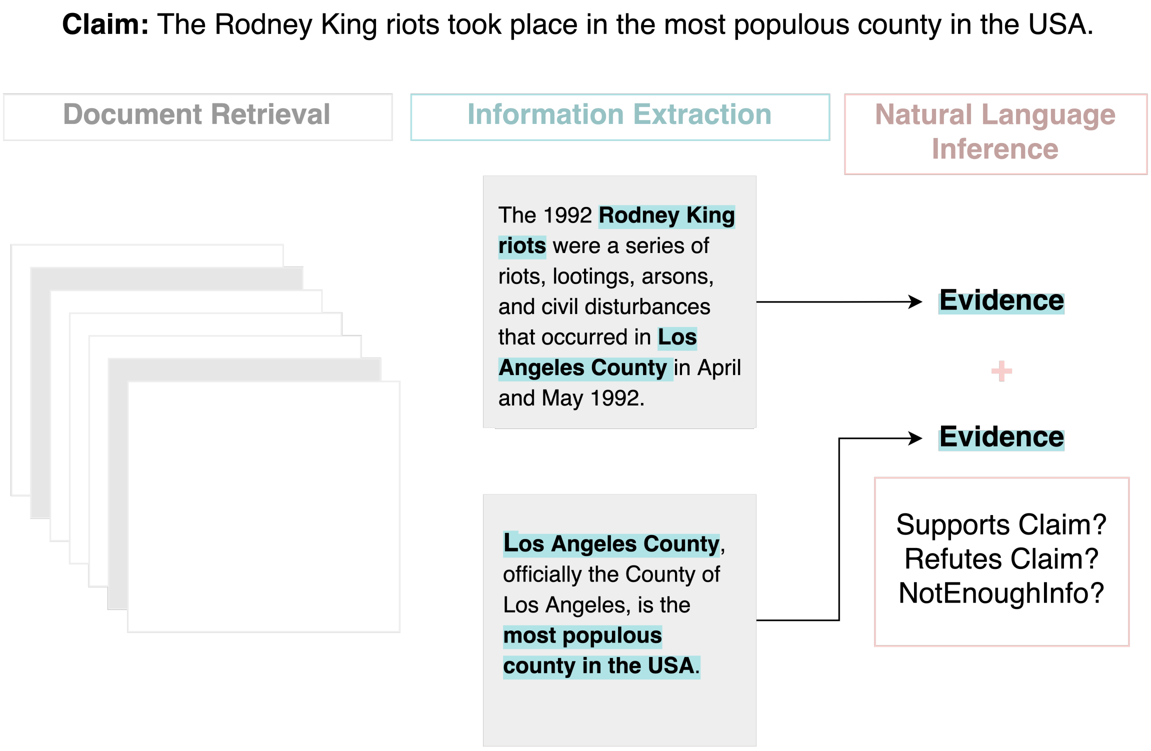
Stance detection consists in identifying attitudes expressed in texts. This task has been found to be relevant as a component to fact-checking and rumour detection tasks, but also on its own (Hardalov et al., 2021). Recent datasets for stance detection have approached this task as a static classification problem: given a text (e.g. news article, tweet, blog post, etc.) and a topic, the system should be able to classify the text into labels such as agrees, disagrees, discusses and unrelated (Hanselowski et al., 2018a; Mohammad et al., 2016). These datasets have been used to develop multiple systems, most of them relying on lexical features (Riedel et al., 2018; Hanselowski et al., 2018a; Ghanem et al., 2018). Other datasets have approached stance detection as a dynamic issue, in which the goal is to predict the stance of a comment in an ongoing discussion (Gorrell et al., 2019). Given a previous text document, these datasets classify another text document (e.g. other tweets, post comments, etc.) into labels such as comment, deny, query and support.

Detecting Fake News Spreaders

Exploring how fake news propagate in social media has been a common approach to tackle disinformation. These approaches include using social media comments to detect fake news (Shu et al., 2019); identifying coordinated disinformation groups of users in social media analysing their behavior (Shao et al., 2020); detecting machine-generated fake news using stylometry (Schuster et al., 2020); or investigating the linguistic aspects of news content to detect disinformation on its source (Zhou et al., 2020). Baly et al. (2018) integrated both stance detection and reliability of the source into a claim verification system, showing the relevance of these features. Atanasov et al. (2019) also combined stance detection and identification of fake news spreaders, by proposing an approach to analyze the behavior patterns of the political trolls according to their political leaning.

Evaluating Check-worthiness of Claims

In order to verify the factuality of a piece of news, it is important to first identify which claims are the most relevant to be fact-checked. This task can consist in ranking sentences in a text in order to choose those that are more likely to need to be fact-checked (Hanselowski et al., 2018a; Elsayed et al., 2019). Alternatively, the task can be seen as a double classification task. First, between factual and non-factual sentences, and then between check-worthy and not-check-worthy claims (Barron-Cedeno et al., 2020; Nakov et al., 2018). Recently, the ClaimBuster dataset (Arslan et al., 2020) has been released. It includes 23k sentences labeled into three categories: non-factual statement, uninportant factual statement and check-worthy factual statement. This dataset wants to become a benchmark for check-worthiness detection.



Claim: The Rodney King riots took place in the most populous county in the USA.

The 1992 **Rodney King riots** were a series of riots, lootings, arsons, and civil disturbances that occurred in **Los Angeles County** in April and May 1992.

Los Angeles County, officially the County of Los Angeles, is the **most populous county in the USA**.

Figure 2: Pipeline of Claim Verification

2.2 The Task of Claim Verification

Claim verification is the task of assessing the veracity of a statement, given some pieces of evidence. The pipeline of claim verification (Figure 2) consists of three main sub-tasks:

- **Document Retrieval**: finding the documents where the information to verify the given claim might be located.
- **Information Extraction**: selecting the information inside this document that might be relevant to the given claim.
- **Natural Language Inference**: understanding the information contained in the pieces of evidence in order to support or refute a claim, or conclude that there is not enough information available.

In the following subsections, we present the state-of-the-art for each of these subtasks.

2.2.1 Document Retrieval

The task of document retrieval has been a relevant issue since the beginning of the World Wide Web. Search engines were some of the first NLP applications: they started as searchers of term salience in text frequency, to evolve into PageRank-based systems, and to finally introduce semantic information and context to account for the intention of the person making the query (Fletcher, 2007). In general, document retrieval is approached as a ranking problem: given a set of documents, the highest ranked ones should be retrieved.

The task of page ranking has been mainly approached with pointwise algorithms: for each text, it should retrieve a score. However, some models approached the issue in a pairwise manner: given two documents, it should decide wether A is more relevant than B (Yates et al., 2021). Up until recently, the most successful architectures for page ranking were regression tree ensembles (Burges, 2010). However, nowadays the most successful page ranking systems are based on the Transformer architecture (Vaswani et al., 2017; Nogueira and Cho, 2020).

In recent applications, document retrieval has been an important part of the ongoing research in open question answering. Open Q&A consists in looking for the location of an answer to a question without knowing the document in which the answer might be located (hypothetically, it could be anywhere in the Web). Wikipedia, the multilingual open-collaborative online encyclopedia[2], has been used as a knowledge database for this task in the past, together with other knowledge resources (Ahn et al., 2005; Buscaldi and Rosso, 2006; Ryu et al., 2014).

Chen et al. (2017a) were the first to develop a system relying solely in Wikipedia articles, what prevented redundancy of sources but also required a much more precise document retrieval system. The document retrieval part of this system compares Wikipedia articles and questions using bag-of-word vectors of bigram counts. This method for document retrieval outperformed the Wikipedia Search API on percentage of questions with the correct retrieved segment in the SQuAD dataset (Rajpurkar et al., 2016).

2.2.2 Information Extraction

Information extraction consists of finding and understanding pieces of text in order to structure its content according to the relevant information. The final goal is to organize information so that it is useful for some purpose. Common goals of information extraction systems have been keyword extraction, relation extraction and named entity and event extraction (Jurafsky, 2000).

In the case of claim verification, the goal of information extraction is to retrieve the sentences that contain the evidence to verify the given claim. Thus, the information extraction module of this task can also be approached as a ranking challenge: for each set of sentences, the most useful ones should be retrieved. For the ranking task, similar approaches to the ones described in Section 2.2.1 can be used.

Additionally, document retrieval can also be performed using keyword extraction (Hanselowski et al., 2018b). The idea is to extract the most important noun phrases of the claim, to then query the Wikipedia API. In this approach, the tasks of document retrieval and information extraction are closely related. Keyword extraction has been performed in several different ways: from TF-IDF computation, to supervised and graph-based models (Firoozeh et al.,

[2] https://en.wikipedia.org/wiki/Wikipedia

2020). Danesh et al. (2015) combined some of these approaches to develop an unsupervised method that ranks ngram candidates with various ranking steps based on traditional statistical features, the position of the first occurrences, and a co-occurrence graph. The task of information extraction has also benefited from Transformer-based models (Baldini Soares et al., 2019; Soleimani et al., 2019).

2.2.3 Natural Language Inference

Natural Language Inference (NLI) is the task of recognizing if there exist textual entailment between one or more premises and a given hypothesis. In this step of claim verification, the retrieved pieces of evidence should be used to assess the truth-status of the initial claim. There exist multiple tasks in NLP that require NLI, such as Question Answering (QA), Natural Language Understanding (NLU), or Summarisation.

MacCartney and Manning (2007) presented a system for inference that used natural language as input. They moved from formal logic representations (Bos and Markert, 2006) to capturing common logical inferences by appealing directly to the structure of language. Angeli and Manning (2014) approached the task as a database completion of common sense facts, demonstrating certain ability of their system to learn these facts.

Language inference can be framed as a relation extraction task: in order to know if a sentence is entailed by another sentence, it is necessary to identify the semantic relation between the verb and the arguments of the premises and hypothesis. For instance, in Figure 3, it is necessary to know that there is a relation between *leading tenor* and *cheap*, given by *comes*; and another relation between *Pavarotti* and *leading tenor* given by *is*, in order to claim that the hypothesis is false.

- Premise: Neither leading tenor comes cheap.
- Premise: One of the leading tenors is Pavarotti.
- Hypothesis: Pavarotti comes cheap.

Figure 3: Modified example of language inference from Cooper et al. (1996)

For this reason, early approaches used semantic information to approach tasks that required NLI. He et al. (2015) introduced the possibility of annotating semantic roles as a question-answering task, showing that predicate-argument structures can be extracted from natural language questions. In the same direction, Stanovsky et al. (2015) demonstrated the contribution of semantic structures, such as OpenIE, when performing text comprehension with a simple unsupervised lexical matching algorithm. We will see more about this in sections 2.4.1 and 2.4.2.

The creation of more extensive datasets, such as SNLI (Bowman et al., 2015), and MNLI (Williams et al., 2018), has allowed researchers to develop systems based on neural networks, which use architectures based on attention (Parikh et al., 2016) and parsers

(Chen et al., 2017b). Previous research noted that one of the main challenges of NLI is extracting meaningful representations of the sentences. Phenomena such as coreference, syntactic ambiguity, quantification, tense, belief and modality should be possible to grasp by the given representation. It has been observed that very long sentences and sentences with negation are the most difficult to represent (Williams et al., 2018).

In most NLI tasks, there is a great need for readily available world knowledge, which is why large pre-trained language models have been so successful. Transformer-based models have substantially improved the performance of NLI benchmarks. We will see more about this in section 2.4.

2.3 State-of-the-art in Claim Verification

Ideally, a claim verification system should be able to take sentences from naturally-occurring texts (e.g. news articles, social media posts or political speeches) and assess their veracity. However, developing training data for this task has some complexities, such as defining the ground truth and creating a knowledge database with boundaries, which allows the annotators to know for sure that the ground truth is right. For this reason, there have been several attempts to approximate the task by creating domain-specific datasets (Scifact, Wadden et al. (2020)) and synthetic datasets (FEVER and HoVer, Thorne et al. (2018); Jiang et al. (2020)). These datasets consist of a set of claims annotated with their ground truth, together with a database of knowledge, in which the truth labels are based (e.g. a set of scientific abstracts or a set of Wikipedia articles). The labels are usually Supports, Refutes and NotEnoughInfo.

There exist other datasets that contain naturally-occurring claims, such as the MultiFC (Augenstein et al., 2019), Liar (Wang, 2017), and other smaller datasets. These are generally scraped from fact-checking websites, and sometimes include the justification of the fact-checker for the given label. However, these datasets do not contain a given and fixed database of evidence. This makes it very difficult to use them to train inference systems, as the ground truth at the moment of fact-checking can be different from the current one (facts change), and there is no gold evidence.

2.3.1 Datasets

FEVER (Thorne et al., 2018) is the benchmark dataset for claim verification. It contains 185,455 claims generated from altering sentences extracted from Wikipedia and labelled as Supports, Refutes or NotEnoughInfo (NEI). It comes with a Wikipedia database of articles, including those from which the claims were extracted. Additionally, the annotators recorded the sentences that were used as pieces of evidence to assess the Supports and Refutes labels.

The task of claim verification requires to build a pipeline which first retrieves the relevant articles from the database, then extracts the most relevant sentences of those articles and finally classifies the claim with respect to the retrieved evidence. The results are evaluated with two measures: accuracy of classification labels only, and accuracy of classification labels given a correct evidence retrieval.

Recent research in NLP has been raising concerns on the extend to which systems exploit unintentional biases and cues that exist on the training dataset (Poliak et al., 2018; Gururangan et al., 2018), instead of actually understanding language (we further explain this issue in section 2.4). These concerns have been gaining relevance as models have been becoming more complex and difficult to interpret. With this in mind, the organizers of FEVER decided to create FEVER 2.0 (Thorne et al., 2019), a shared task with a setup of *build-it, break-it, fix-it*. That means that the participants were expected to submit a system, then create adversarial attacks to break the submitted systems, and finally investigate ways to fix the initial systems given the new attacks. In the breaking phase, participants focused on the shortcomings of the original FEVER dataset, such as the lack of complex claims that required multi-hop inference and temporal reasoning (Hidey et al., 2020), or arithmetic and logical reasoning (Kim and Allan, 2019). These attacks proved to be difficult to solve by the systems of the task.

The HoVer task (Jiang et al., 2020), was built upon one of the perceived shortcomings of FEVER: multi-hop reasoning. The objective was to create a dataset in which evidence can be required from up to 4 different Wikipedia articles. The 26,000 claims of HoVer are, as a consequence, way longer than the claims in FEVER, adding difficulty to both the retrieval and the inference sub-tasks. Overall, the task is very similar to FEVER. However, HoVer is a binary classification task between Supported and Not-Supported (combining Refutes and NEI in this single category).

MultiFC (Augenstein et al., 2019) is a dataset of 34,918 naturally-occurring factual claims retrieved from 26 fact-checking websites in English. The claims have rich metadata, such as the name of fact-checker, the date of fact-checking and often also the reasons for the given label. Additionally, the dataset comes with a set of automatically retrieved evidence snippets. However, this dataset has several shortcomings, namely:

- The original labels from the 26 fact-checking websites are not mapped, which results in a dataset with 126 different labels (compared to 3 in FEVER and 2 in HoVer).
- The dataset does not come with a fixed database of knowledge, which makes it possible that the labels given to claims are no longer true with the current (online) information (e.g. something that was not known when the label was given can be known now).
- The retrieved snippets are not manually annotated gold evidence, which makes it unclear if the labels can be inferred from the given evidence.

For these reasons, MultiFC is not a good dataset to train textual inference. However, it can be used to observe which attributes are more common in naturally-occurring claims in order to take them into account when evaluating systems on synthetic datasets such as FEVER or HoVer.

Scifact (Wadden et al., 2020) is a dataset of 1,409 scientific claims extracted from referenced sentences of scientific articles by experts. For this reason, claims in Scifact are very close to naturally-occurring ones. The dataset includes a corpus of 5,183 article abstracts, which are the database where the information is stored. Scifact mainly follows FEVER's approach. However, the truth-label of a claim is not absolute. Instead, different abstracts can Support or Refute a single claim, although this does not occur often. This is a dataset of the science domain, and the task is challenging because the model should learn scientific terminology and often perform numerical reasoning.

Finally, the UKP Snopes dataset (Hanselowski et al., 2019) is the largest dataset of naturally-occurring claims with manually annotated evidence. It contains 6,422 claims and 14,296 documents to retrieve the evidence from. The claims were crawled from fact-checking websites and the annotators noted both the stance (agrees, refutes and no stance), and the sentences which served as evidence to the claim. This is a potentially very useful dataset. However, its size is still far from FEVER and MultiFC.

2.3.2 Systems

Several systems have been developed to deal with the task proposed in FEVER, as it is the most used dataset for claim verification. These systems focus on dealing with one, two or three of the sub-tasks: document retrieval, sentence retrieval and natural language inference.

The baseline of the FEVER task (Thorne et al., 2018) uses an evidence and sentence retrieval approach based on Chen et al. (2017a) (see section 2.2.1) and the feature extraction approach proposed by Riedel et al. (2018) in the FakeNewsChallenge (Hanselowski et al., 2018a). Riedel et al. (2018)'s approach computes term frequency vectors for both the retrieved evidence and the main claim, as well as the cosine similarity between the normalised TF-IDF vectors of both. It then structures these features concatenating the TF-IDF vectors with the cosine similarity values in between. The final feature vector has size of 10,001, and is fed into a multi-layer perceptron with one hidden layer.

The shared task proposed by the authors of the task in July 2018 had 23 different participants. The team that achieved the highest performance in evidence recall (85.19) was Athene UKP TU Darmstadt (Hanselowski et al., 2018b). They used noun phrases to query the Wikipedia search API. The team that obtained the highest label accuracy, UNC-NLP (Nie et al., 2019), concatenated the evidence sentences into a single string and included an additional token-level feature: the sentence similarity score from the sentence retrieval module. They achieved an accuracy of 68.21 on label accuracy.

After the end of shared task, and given the fast improvement of many NLP tasks due to the release of Transformer-based language models, such as BERT (Devlin et al., 2019), new systems have been released for the FEVER task. Soleimani et al. (2019) achieved a label accuracy of 71.70 using BERT for the inference module and the evidence retrieved by Hanselowski et al. (2018b), two points higher than Nie et al. (2019).

Later on work has focused on new approaches to feature extraction and structuring, which allow for interaction and reasoning between different pieces of evidence using graph-based representations (Zhou et al., 2019; Zhong et al., 2020).

Zhou et al. (2019) proposed to move from a claim-evidence concatenation system to a fully-connected evidence graph that allows for information propagation among evidences. This system starts by encoding the claim and the claim-evidence pairs with BERT (Devlin et al., 2019), which results in 6 input sentence pairs (given that they use 5 evidence sentences per claim). Each of these pairs is a node of the graph. An attention layer is then used to propagate the information within the nodes. A second attention layer combines the previous computations with the representation of the claim. The resulting representation is used to make the decision about the final output. They call this system GEAR (Graph-based Evidence Aggregating and Reasoning) and use the evidence retrieved by Hanselowski et al. (2018b) as input.

Zhong et al. (2020) followed a similar approach, but used semantic information in order to create the knowledge graph. More specifically, they extract Semantic Roles with the AllenNLP parser (Gardner et al., 2018; Shi and Lin, 2019), and structure the claim and the pieces of evidence into tuples of predicates and arguments. They encode these propositions using XLNet and re-define the relative distance between arguments. They then use each argument as a node of the graph and propagate and aggregate information from neighbouring nodes of the graph.

The baseline created by the authors of HoVer uses fine-tuned BERT models for all the steps of the pipeline (evidence retrieval, sentence retrieval and claim verification) (Jiang et al., 2020). This system only gets the right label in 67.6% of the development set, and it drops to just 14.9% if the right retrieved evidence is also required. The only other system developed for this task focuses on improving the retrieval module (Khattab et al., 2021). Their system introduces a condensed retrieval architecture that summarises the retrieved facts and uses them as part of the query to subsequent hops. They also allow different parts of the same query to match different passages of the evidences. They significantly improve the retrieval module, consequently improving the accuracy of the labels: they get 73.7% of the labels right in the development set.

Augenstein et al. (2019) developed a multi-task learning system to deal with the task in MultiFC. They account for the multiple labels by creating embeddings for each of these labels, and combining those with the evidence-claim embedding. With this approach, the semantic closeness between labels is learned automatically, which is additional knowledge to learn the labels of each claim-evidence pair. They also use metadata as additional input, and they achieve a Macro F1 of labels of 49.2% in the test set.

To sum up, many of the state-of-the-art systems for claim verification use large pre-trained language models, such as BERT or XLNet, as the backbone of their model. Thus, in the following section we introduce these models, highlighting some of their limitations, and present work that have been trying to overcome these drawbacks.

2.4 Language and Knowledge Representation

Significant improvements on downstream NLP tasks have been made with transfer learning. In transfer learning, neural networks are first trained on a different but related task, with the goal of capturing relevant knowledge. Then, the pre-trained language model is fine-tuned on the target task, with the goal of reusing the knowledge captured in the pre-training phase to improve performance on the aimed task.

The release of the language model BERT (Devlin et al., 2019) revolutionised the performance of many NLP tasks, specially tasks involving inference (see Section 3.2). After BERT, many other models that use different amounts of data, parameters or training tasks have been developed; RoBERTa (Liu et al., 2019) and XLNet (Yang et al., 2019) are some of the most successful ones. These two last models (RoBERTa and XLNet), obtained state-of-the-art results in the MNLI benchmark, both of them achieving an accuracy of around 90% in the task.

Despite these very promising results, researchers have been raising concerns suggesting that the success of natural language inference models has been overestimated (Gururangan et al., 2018; Gupta et al., 2021). Recent work has shown that automated models tend to look for shortcuts and rely on linguistic cues when being trained for specific NLP tasks. These cues are present in the training dataset and very often they come from the annotation process. This is specially the case for datasets where humans generated the data (Poliak et al., 2018), as is the case of SNLI, MNLI and FEVER. Some common annotation cues that have been observed in NLI tasks are: adding negation for contradictory statements, using generic words for entailed sentences, or adding purpose clauses to neutral hypothesis. Removing these cues causes a significant accuracy drop for state-of-the-art systems (Gururangan et al., 2018). In the FEVER dataset, for instance, Schuster et al. (2019) developed a claim-only model in BERT, which achieved a performance of 61.7%, way above the 33.3% that would be expected with no evidence, this indicates the existence of certain linguistic cues in the FEVER dataset.

Ideas on how to deal with this issue have come from proposing new evaluation techniques that go beyond accuracy: such as creating adversarial examples to break shallow patterns (Jia and Liang, 2017), or applying attribution techniques to identify the key elements of the input that contributed to the output (Mudrakarta et al., 2018). Recently, Ribeiro et al. (2020) introduced CHECKLIST, a set of tests to evaluate the different linguistic capabilities expected by a model.

Other approaches have proposed incorporating linguistic knowledge into deep language models in order to make them grasp natural language better, as well as to make them more explainable. This new direction suggests using information that had been helpful for NLI models before the arrival of deep learning, in order to guide the self-attention mechanisms (Zhang et al., 2020b). Zanzotto et al. (2020) designed a system that explicitly embeds syntax parse trees into sentence embeddings using distributed tree kernels, and can visualise the decisions made (KERMIT). Zhang et al. (2020a) introduced a modified BERT architecture, that maps semantic role labels to embeddings in parallel and integrates the text representation with the contextual explicit semantic embedding to obtain a joint representation. This last system improves the state-of-the-art of NLI tasks, such as SNLI and SQuAD 2.0 (Rajpurkar et al., 2018). We will expand into the use of semantic representations in sections 2.4.1 and 2.4.2.

Finally, there have been growing concerns over the lack of explainability of current

NLP models. Some efforts towards this direction have already been mentioned just above: like the attribution techniques presented by Mudrakarta et al. (2018), or the syntactic visualisation of KERMIT (Zanzotto et al., 2020). Other interesting efforts to explain the behaviour of deep-learning models and NLI models in general will be described in Section 2.4.3.

2.4.1 Semantic Role Labels

Semantic roles (also called thematic labels) represent the different arguments that a predicate might have. These semantic categories are relations between noun phrases and verbs. An ideal set of roles should be able to concisely label the arguments of any relation. Nonetheless, the exact set of these relations is an open discussion inside the linguistic community (Bonial et al., 2011).

Lexical resources such as FrameNet (Baker et al., 1998), VerbNet (Kipper et al., 2000), and PropBank (Palmer et al., 2005) have been largely used to deal with NLP tasks. Although these three annotation frameworks all have the goal of creating semantic representations between predicates and its arguments, their focus is different.

- **FrameNet**: [Mr. Bean]$_{BUYER}$ bought [the sweater]$_{GOODS}$ [from the second hand store]$_{SELLER}$ [for 400 pounds]$_{PAYMENT}$.
- **VerbNet**: [Mr. Bean]$_{Agent}$ bought$_{get-13.5.1}$ [the sweater]$_{Theme}$ [from the second hand store]$_{Source}$ [for 400 pounds]$_{Asset}$.
- **PropBank**: [Mr. Bean]$_{Arg0}$ [bought]$_{V}$ [the sweater]$_{Arg1}$ [from the second hand store]$_{Arg2}$ [for 400 pounds]$_{Arg3}$.

Figure 4: Example of each semantic representation

FrameNet is focused on semantic frames: schematic representations of situations involving various participants, propositions, and other conceptual roles (Fillmore, 1976). This approach starts by choosing a semantic frame (e.g. commerce) to then look for its participants and other elements (e.g. BUYER, SELLER, PAYMENT) through the different lexical predicates that are common in that frame (e.g. buy, sell, pay).

VerbNet is a hierarchical verb lexicon that groups verbs into classes based on similarities in their syntactic and semantic properties. In each class, VerbNet includes a group of member verbs and the semantic roles used in the arguments of the predicate. In the example in Figure 4, the verb *buy* is grouped under the hierarchical class *get-13.5.1*, which commonly has an Agent (a participant who gets something), a Theme (what is being gotten) and a Source (from whom/where does it get it).

Finally, PropBank was created as a practical approach to semantic representation. Its goal is to create a shallow but broad representation that covers every instance of every verb in a corpus to allow representative statistics to be calculated. For this reason, the PropBank framework has been the most broadly used in NLP. PropBank defines semantic roles on a verb-by-verb basis: individual verb's semantic arguments are numbered, beginning with zero. In the example in Figure 4, the Agent in VerbNet becomes Arg0 in PropBank, and the Theme becomes Arg1.

PropBank was designed to be used in automated tasks. For this reason, multiple models have made use of this representation for tasks such as Question Answering and Text Comprehension (Shen and Lapata, 2007; Khashabi et al., 2018; Zhang et al., 2020b). With the popularization of deep learning architectures, and specially of contextual word embeddings, semantic representations based on sets of labels seem to have lost some relevance. However, recent work has proved their usefulness as additional information to Transformers (Zhang et al., 2020a). Zhong et al. (2020) used SRL tuples to structure information graphs for automated claim verification.

2.4.2 Open Information Extraction

Open Information Extraction (OpenIE) was first introduced as an extraction paradigm to tackle an unbounded number of relations (Etzioni et al., 2008). Systems based on OpenIE extract relational tuples from text by identifying relation phrases and the arguments associated to these relations (Mausam et al., 2012). Stanovsky et al. (2015) were the first to propose this task as an intermediate structure for other semantic tasks, similar to what was already being done with other linguistic information, such as semantic roles, syntactic dependencies or lexical representations. They demonstrated that for the tasks of text comprehension, word similarity and word analogy, OpenIE structures can be more useful than the sentence representation structures mentioned above.

- **PropBank**: [John]$_{Arg0}$ [refused]$_V$ [to visit a Vegas casino]$_{Arg1}$.
  [John]$_{Arg0}$ refused to [visit]$_V$ [a Vegas casino]$_{Arg1}$.
- **OpenIE**: [John]$_A$ [refused to visit]$_V$ [a Vegas casino]$_A$.

Figure 5: Example of the representations extracted with OpenIE from Stanovsky et al. (2015).

In the example in Figure 5, the semantic role labels extracted with the PropBank framework identify two different propositions because there are two different verbs. The first one extracts the tuple *(John, refused, to visit a Vegas casino)*, and the second one the tuple *(John, visit, a Vegas casino)*. This representation could mislead a textual inference model, as the first and the second extracted propositions seem to contradict each other. In OpenIE, instead, the model identifies the multi-word predicate *refused to visit*. The resulting representation *(John, refused to visit, a Vegas casino)* seems intuitively more useful for a task of language inference.

2.4.3 Explainability

With the appearance of large language models, performance in complex NLP tasks, such as language understanding, has been improving, to the point of overcoming human performance in many datasets. However, recent work has found that models often use linguistic cues embedded in datasets and other strategies to perform its predictions (Poliak et al., 2018; Gururangan et al., 2018). For this reason, as NLP models become more complex and data-hungry, it becomes more important to not just get the performance of a model, but also to understand how does it get to these conclusions. In the case of claim verification, it is crucial to know where did the model get the information to decide the truth-label of the claim (Atanasova et al., 2020b).

To overcome these drawbacks, it has become a regular practise to try to train NLI models using just the claim or just the evidence as input (Poliak et al., 2018). This is a way to test the linguistic cues embedded in the dataset. If it is possible to train a model using just the claims, it means that there are enough clues in the claims itself to guess the label, and consequently the dataset should be rethought.

Explainability methods for NLP models have been proposed in three directions: perturbation-based explanations, gradient-based explanations, and generation of text serving as an explanation (Atanasova et al., 2020a). The first approach consists in generating adversarial attacks to the model, which modify current input instances to identify the capacities that the model has and the ones that it does not (Jia and Liang, 2017; Mudrakarta et al., 2018; Ribeiro et al., 2020). This can also consist in replacing tokens with zeros and measuring the change in output (Zeiler and Fergus, 2014). A second approach has come from extracting saliency scores, which indicate which elements of the input had more influence in the final output. These saliency scores can then be compared to human annotations of salient input regions, to asses if the rationales of the model agree with human ones (Atanasova et al., 2020a; DeYoung et al., 2020). These scores can also be compared to assess confidence or consistency. Similarly, other approaches have consisted in looking at the layers of deep-learning models to try to understand what does each of these layers learn (Vig, 2019; Zanzotto et al., 2020). Regarding explainability of claim verification, Atanasova et al. (2020b) went further and focused on generating automated justifications for verdicts on claims. For this purpose, they created a multi-task model that generates the explanations and predicts the veracity of the claims at the same time. In this direction, other work has focused on generating explanations of why a piece of news is detected as fake in social media (Lu and Li, 2020; Shu et al., 2019).

We have concluded this background review by emphasizing the field's shifting focus towards more explainable models in NLP, specially in critical topics such as fake news. In the following section, we present the resources used in our experiments.

3 Resources

In this section we introduce all the resources that were used in this work. First, we present the FEVER dataset in more detail, observe its attributes and describe evidence retrieval module for our experiments. Then, we describe the pre-trained model that we use for transfer learning (BERT), the architecture we use in our experiments (SemBERT), and the parsers we use to extract the semantic information.

3.1 FEVER

In our experiments we used the FEVER dataset (Thorne et al., 2018). This is a synthetic dataset, therefore its generation process potentially conditions our results and conclusions. In the following paragraphs, we introduce how it was created and then point at some of its drawbacks. We also do an attribute comparison to other claim verification datasets.

3.1.1 Creation of the Dataset

The FEVER dataset has been a benchmark for claim verification since it was released in 2018. The dataset consists of 185,445 generated claims with its truth label and the evidence for that label. The first construction phase of this dataset consisted in generating the claims. To this goal, the authors took the June 2017 Wikipedia dump, processed it with Stanford CoreNLP (Manning et al., 2014), sampled the introductory sections of approximately 50,000 popular pages, and indexed the resulting sentences. The claims were then generated by annotators following this procedure:

1. The annotators were given one sentence at random and had to generate some claims, each containing a single piece of information, focusing on the entity that its original Wikipedia page was about.

2. To allow for some (controlled) increase of the complexity of the claims (an avoid mere paraphrases), the annotators were allowed to use information coming from the first sentence of the Wikipedia entries of all the hyper-linked terms in the original sentence.

3. The annotators were then asked to generate mutations of the claims, altering them in ways that may or may not change their truth label. The types of mutations were: paraphrasing, negation, substitution of entity/relation, and making the claim more general or specific.

4. The annotators were asked to avoid trivial negations, such as sentences using *not*.

The second phase to build the dataset consisted in labeling the claims as Supports, Refutes or NotEnoughInfo (NEI). The annotation interface displayed all sentences of the introductory section of the article that the claim came from and of the articles of every hyper-linked entity. When labelling as Supports or Refutes, the annotators had to record

	Supports	**Refutes**	**NEI**
Training	80,035	29,775	35,639
Development	3,333	3,333	3,333
Test	3,333	3,333	3,333
Reserved	6,666	6,666	6,666

Table 1: Statistics of the FEVER dataset

which pieces of evidence were needed to be certain about the given label. Adding other Wikipedia entries was also allowed. But the annotators were advised not to spend more than 2-3 minutes per claim.

The annotators team consisted of 50 people, 25 of which were involved in the first phase. They were native US English speakers. The Fleiss k score of both finding the evidence and giving the claim a label was 0.684. The final dataset has four partitions: training, development, test and a reserved set for the shared task (Thorne et al., 2018). The statistics can be seen in Table 1.

3.1.2 Datasets Attribute Comparison

As stated in Sections 2.3 and 2.4, growing concerns have noted that the results of deep learning models are often biased by the dataset that is used for training them (Poliak et al., 2018; Gururangan et al., 2018). For this reason, we found it necessary to get to know the attributes of our dataset and compare them in more detail to other claim verification datasets that have been introduced in Section 2.3.1.

Synthetic datasets are useful given the difficulties to create a structured database with all the knowledge needed to verify naturally-occurring claims. However, Thorne and Vlachos (2019) already pointed out that FEVER misses some of the complexity that naturally-occurring claims have. Some types of reasoning that are commonly needed in naturally-occurring claims but rarely appear in FEVER are:

- Claims that require multi-hop document/sentence retrieval.
- Claims that contain rich semantics in long and complex sentences, which also often imply multi-hop reasoning.
- Claims that require temporal reasoning.
- Claims that require mathematical reasoning.

These drawbacks have been noted by previous work, however we did not find any empirical study showing how often these phenomena appear in FEVER compared to naturally-occurring claims. For this reason, we decided to perform exploratory annotations of a random sample of 300 claims from FEVER, HoVer and MultiFC (100 claims per dataset), which currently are the largest existing datasets for claim verification.

We annotated semantic complexity, relevance of time reasoning, time complexity, and mathematical reasoning. In the following paragraphs, we will describe the annotation guidelines for each of these issues and note the observations we extracted from this process.

Semantic Complexity

As a proxy for semantic complexity, we decided to annotate the number of verbs (or predicates) per claim excluding gerunds and auxiliaries. As can be observed in Figure 6, while claims in FEVER are almost always simple (contain one single verb), that is not the case in the other two datasets. HoVer is synthetically created to have claims that require multi-hop, so there are a lot of complex claims. MultiFC follows a Benford distribution, which is seen as more natural, in which the number of claims decreases when complexity increases.

Guidelines of the annotation are:

1. The claim just has one verb.
2. The claim has two verbs.
3. The claim has three verbs.
4. The claim has more than three verbs.

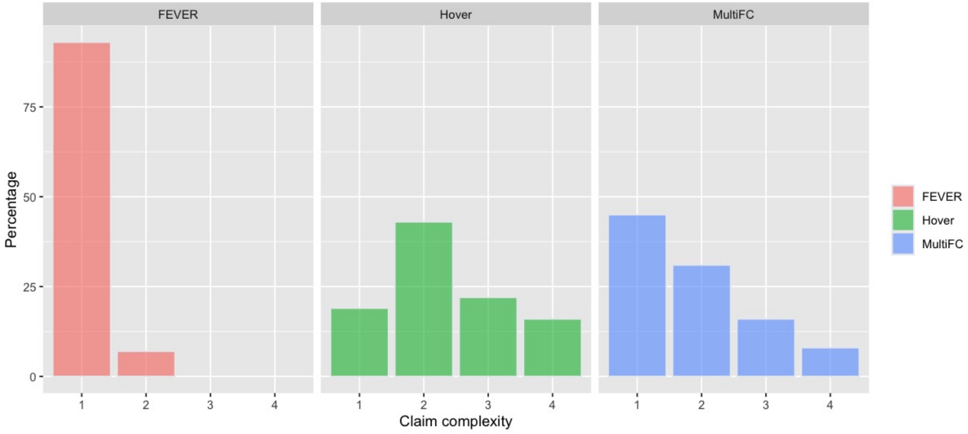


Figure 6: Semantic complexity of the claims in FEVER, HoVer and MultiFC

Temporal Reasoning

Annotating time was challenging, as we wanted to know if there was a need of reasoning through time in order to verify the claim, and also the complexity of that reasoning. For this reason we performed two different annotations: time reasoning and time complexity.

In Figure 7, we observe that claims that can be verified without knowing the date in which the claim was stated do not exist in naturally-occurring claims (at least not in our

Guidelines for **time reasoning**:

0. Time is NOT relevant to the claim.
1. Implicitly, time of the claim is relevant.
2. The claim explicitly mentions time which is relevant to the claim, but the date itself is not included (e.g. yesterday, last week).
3. The claim explicitly mentions time which is relevant to the claim.

Guidelines for **time complexity**:

0. No time reference.
1. One date is relevant.
2. Two or more dates are relevant.
3. There is a range of time that is relevant.
4. The claim compares facts of different dates.

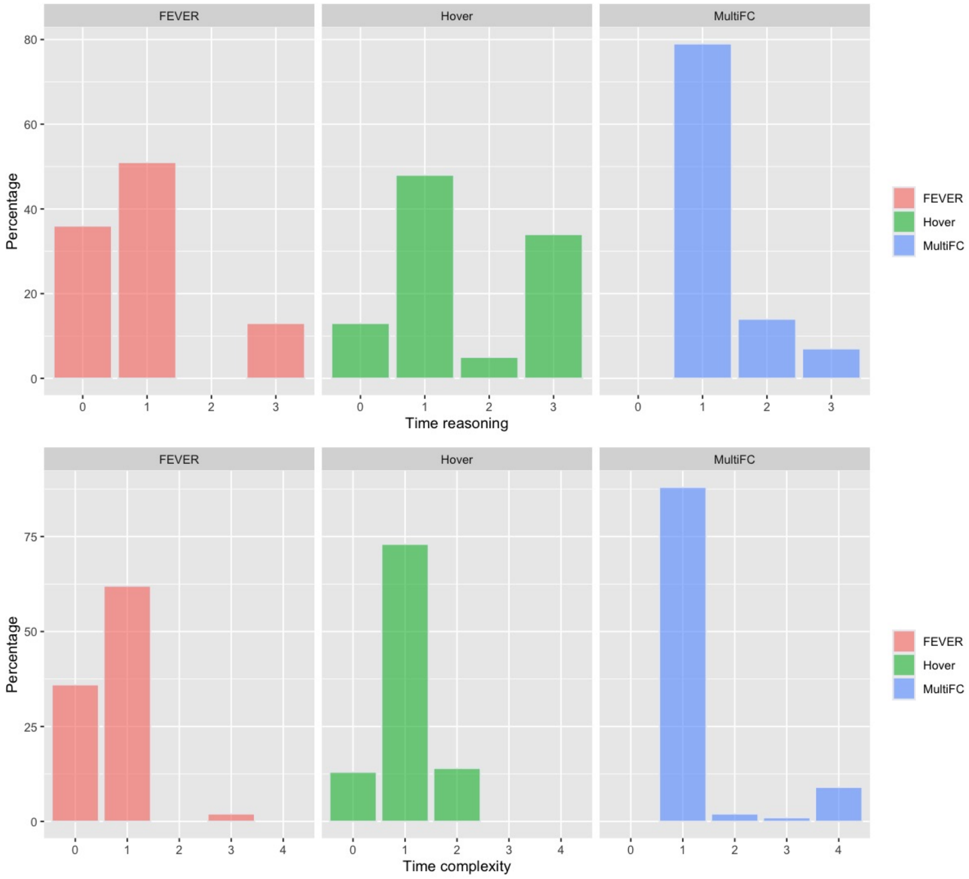


Figure 7: Time references and complexity of the claims in FEVER, HoVer and MultiFC

random sample), while they are a big part of the FEVER dataset. In HoVer, it seems like the extra complexity comes with additional explicit time references and comparison between different dates. We also observe that comparing facts for different dates is something common in naturally-occurring claims, but it never happens in the synthetic datasets.

Mathematical Reasoning

We annotated the complexity of mathematical reasoning in a similar fashion to time complexity. In Figure 8, it can be observed that math reasoning is more common in naturally-occurring claims than in synthetic ones. Mathematical reasoning has been a subject of research in multiple work (Dua et al., 2019; Andor et al., 2019), but none of it has focused in the task of claim verification. Kim and Allan (2019) tried to account for these complexities in the FEVER2.0 shared task, but the instances they created failed to meet the guidelines of the shared task and were not included.

Guidelines of the annotation are:

0. No maths needed.
1. One operation needed.
2. Two operations needed.
3. More than two operations needed.

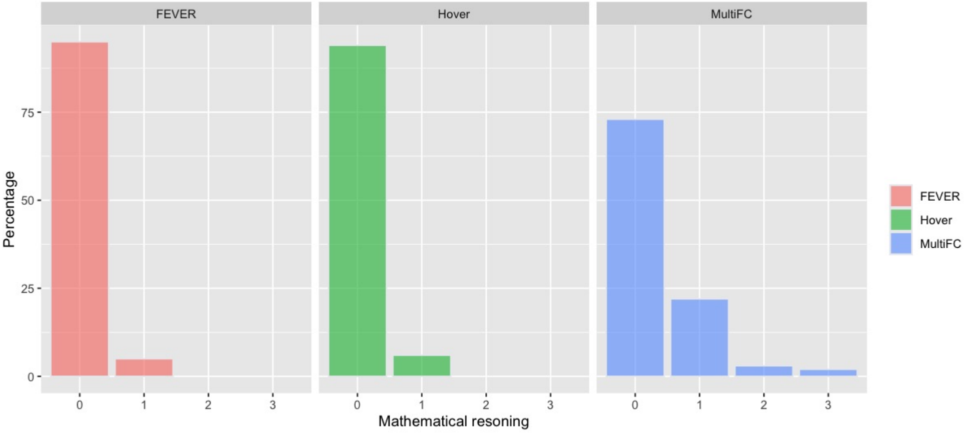


Figure 8: Mathematical reasoning in the claims in FEVER, HoVer and MultiFC

3.1.3 Evidence Retrieval Module

Given that this research project focuses on the natural language inference module of claim verification, we do not perform evidence retrieval, and instead we use the evidences retrieved by the system that had the highest evidence recall in the FEVER shared task.

We have used the top 5 evidences extracted by Hanselowski et al. (2018b) and have used the scripts from Zhou et al. (2019) to put the data in the right format for BERT. From these scripts, we have removed the part where they concatenated named entities to the end of each evidence, as it was found not to be useful and it made the evidences too long for an efficient computation. Here we can see an example of each of the values contained for each instance in the input:

- ID: 0
- Label: NOTENOUGHINFO
- Claim: Colin Kaepernick became a starting quarterback during the 49ers 63rd season in the National Football League.
- Evidence 1: He remained the team 's starting quarterback for the rest of the season and went on to lead the 49ers to their first Super Bowl appearance since 1994 , losing to the Baltimore Ravens .
- Evidence 2: Kaepernick began his professional career as a backup to Alex Smith , but became the 49ers ' starter in the middle of the 2012 season after Smith suffered a concussion .
- Evidence 3: During the 2013 season , his first full season as a starter , Kaepernick helped the 49ers reach the NFC Championship , losing to the Seattle Seahawks .
- Evidence 4: In the following seasons , Kaepernick lost and won back his starting job , with the 49ers missing the playoffs for three years consecutively .
- Evidence 5: Colin Rand Kaepernick -LRB- -LSB- ‘ kæprnk -RSB- ; born November 3 , 1987 -RRB- is an American football quarterback who is currently a free agent .

3.2 BERT

Our experiments are based on Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) and variants of this model. These models have a high performance in the topic studied in this work, namely, natural language inference for fact-checking. The Transformer architecture (Vaswani et al., 2017) consists of several layers of multi-headed self-attention with feed-forward layers and skip connections. As previous architectures, they have an encoder-decoder structure. Unlike RNNs, that maintain a recurrent state and process an input sequentially, Transformers can compute all features of a vector in parallel. This allows Transformers to be trained significantly faster than architectures based on recurrent or convolutional layers.

BERT is trained with two different objectives: Masked LM (MLM) and Next Sentence Prediction (NSP). MLM consists in replacing 15% of the tokens in a text with $[MASK]$ and letting the model attempt to predict the original value of the masked words; this task computes the word embeddings. NSP consists in pairing two sentences with the special token $[SEP]$ and trying to predict if the second sentence is the subsequent sentence in the original document. A $[CLS]$ token is inserted at the beginning of the first sentence to account for the joint representation of the pair of sentences. Together with the token

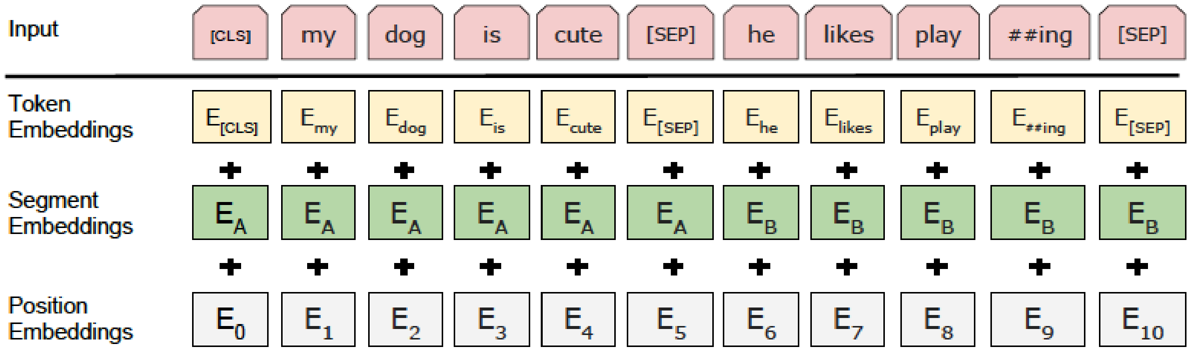


Figure 9: BERT input representation, from Devlin et al. (2019)

contextualized embeddings, the input also contains a sentence embedding indicating which token corresponds to which sentence, and a positional embedding. In Figure 9, an example of the BERT input representation can be observed: the tokens are given as input, and a token embedding, a segment embedding and a positional embedding are used to represent each single token.

Language models trained with Transformer architectures can be used either to extract text representations – using its contextualized word embeddings as features; or as a system – by adding a fine-tuning layer on top of the pre-trained model. In our experiments, we use BERT as a system.

3.3 SemBERT

BERT is designed to be given plain natural text as input. However, recent work suggests that it could benefit from additional linguistic knowledge. Zhang et al. (2020a) proposed an architecture that is able to encode both natural text and semantic information: SemBERT. We are going to use this architecture for our experiments with SRL and OpenIE.

As a first step, SemBERT encodes text in the same way that BERT does: tokenizing the text into sub-tokens and computing contextualized embeddings for each of these sub-tokens. In parallel, SemBERT takes the semantic representation that it is given, which should have one tag per word (SRL in the original paper), and computes tag embeddings. Given that a single sentence can have several predicates, and consequently several argument-predicate structures (propositions), the authors allow for up to three different representation vectors. In order to combine the BERT sub-token representation with the semantic representations (which is computed by word), they need to be aligned. A convolutional neural network does this by merging back the sub-tokens to obtain a BERT word-level representation. Additionally, a linear layer aggregates the three semantic representation vectors (for the three propositions per sentence allowed) into one final semantic embedding. Then, the BERT word representation and the final semantic representation are concatenated, in the

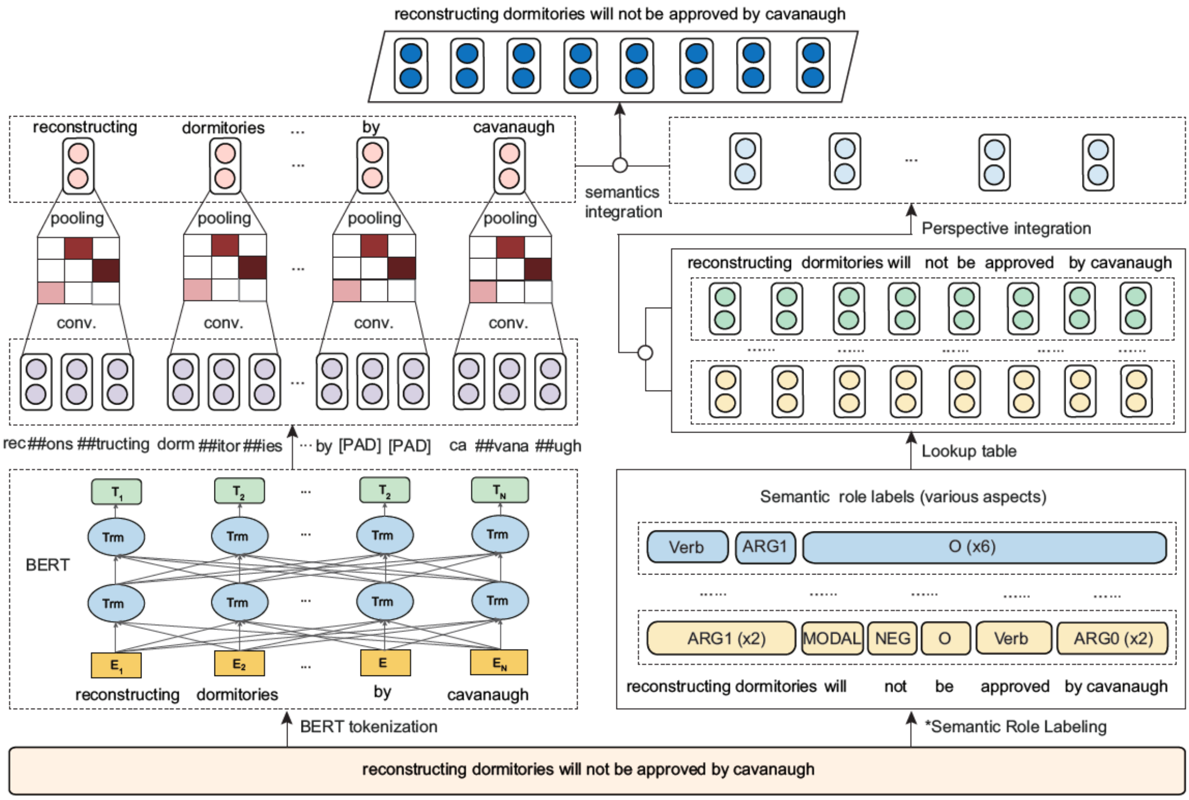

reconstructing dormitories will not be approved by cavanaugh
reconstructing
dormitories
by
cavanaugh
pooling
conv.
semantics integration
Perspective integration
reconstructing dormitories will not be approved by cavanaugh
Lookup table
rec##ons ##tructing dorm ##itor ##ies ··· by [PAD] [PAD] ca ##vana ##ugh
BERT
Trm
T_1
T_2
T_N
E_1
E_2
E
E_N
reconstructing dormitories by cavanaugh
Semantic role labels (various aspects)
Verb
ARG1
O (x6)
ARG1 (x2)
MODAL
NEG
O
Verb
ARG0 (x2)
reconstructing dormitories will not be approved by cavanaugh
BERT tokenization
*Semantic Role Labeling
reconstructing dormitories will not be approved by cavanaugh

Figure 10: SemBERT architecture from Zhang et al. (2020a)

step that is referred as *semantics integration* in Figure 10.

According to the authors, SemBERT outperforms BERT in NLI tasks increasing the final accuracy between 1 and 3 percentage points (Zhang et al., 2020a).

3.4 Semantic Parsers

This project aims to integrate semantic information to perform inference for claim verification. The hypothesis is that this information might facilitate the reasoning in complex semantic structures. We extract two kinds of semantic structures: Semantic Role Labels (SRL) and OpenIE. We use SRL because previous work has shown that it can be useful for the task of claim verification (Zhong et al., 2020), and OpenIE because other work has shown that it is a very intuitive structure for the task of text comprehension (Stanovsky et al., 2015), which is directly related to claim verification. The parsers that we used to extract both of these structures are presented in the following sections.

3.4.1 Semantic Role Labeling Parser

Semantic Role Labeling consists in extracting the predicate-argument structures of each sentence (see Section 2.4.1). Therefore, the automatic extraction of semantic role labels implies four subtasks: predicate detection, predicate sense disambiguation, argument identification and argument classification. PropBank-based approaches usually represent arguments as spans – they look for the beginning and end of the argument and annotate the whole chunk.

Shi and Lin (2019) developed a BERT-based model to extract PropBank SRL, where the predicate is already identified. Their model performs predicate sense disambiguation using BERT for sequence labeling. To identify and classify arguments, they encode the input as *sentence [SEP] predicate(verb)*, to allow the predicate to interact with the whole sentence. This model had a In Figure 11, the architecture of the model is shown. AllenNLP (Gardner et al., 2018), a semantic NLP platform, incorporated this model in its library, and we used it in this project to extract the SRL.[3] The output of this model is a dicitonary in which every identified predicate is a verb entry, and comes with a list of tags that correspond to each word in the sentence, having the tag *O* for words outside the proposition. The tags follow the BIO[4] tagging scheme (Ramshaw and Marcus, 1995) and the PropBank set of arguments. The output of the example on Figure 4 would look like:

```
{
    description: [ARG0: Mr. Bean] [V: bought]
        [ARG1: the sweater] [ARG2: from the second hand store]
        [ARG3: for 400 pounds] .,
    tags: [B-ARG0, I-ARG0, B-V,B-ARG1,I-ARG1,B-ARG2,
        I-ARG2,I-ARG2,I-ARG2,I-ARG2,B-ARG3,I-ARG3,
        I-ARG3,O],
    verb: bought,
    words: [Mr.,Bean,bought,the,sweater,from,
        the,second,hand,store,for,400,pounds,.]
}
```

3.4.2 Open Information Extraction Parser

To extract OpenIE tuples we also use the parser provided in the AllenNLP platform, which comes from the model designed by Stanovsky et al. (2018).[5] Similar to the SRL task described above, they frame OpenIE as a sequence labelling task and use the BIO tagging scheme. The output of this model is tuples of predicates and arguments which are tagged as P for predicates and as A_i for arguments, where i is the natural order of the arguments. Additionally, multi-word predicates are allowed, and a single predicate can be in more than a tuple in certain syntactic constructions (e.g. apposition, co-ordination or coreference).

Stanovsky et al. (2018) developed a bi-LSTM system to perform OIE labelling which takes as input a word with respect to a predicate representation, which consists of a word

[3] A demo of this model can be found in `https://demo.allennlp.org/semantic-role-labeling`

[4] Beginning, Inside, Outside

[5] A demo of this model can be found in https://demo.allennlp.org/open-information-extraction

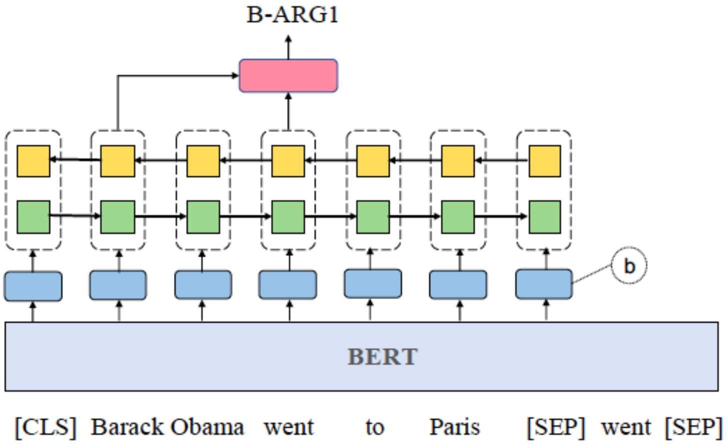


Figure 11: Architecture of the argument identification and classification model in Shi and Lin (2019)

representation (orange circle in Figure 12) concatenated with a predicate representation (yellow circle in Figure 12). Both of these representations are the word embedding of the corresponding token plus the word embedding of the part-of-speech of the token.

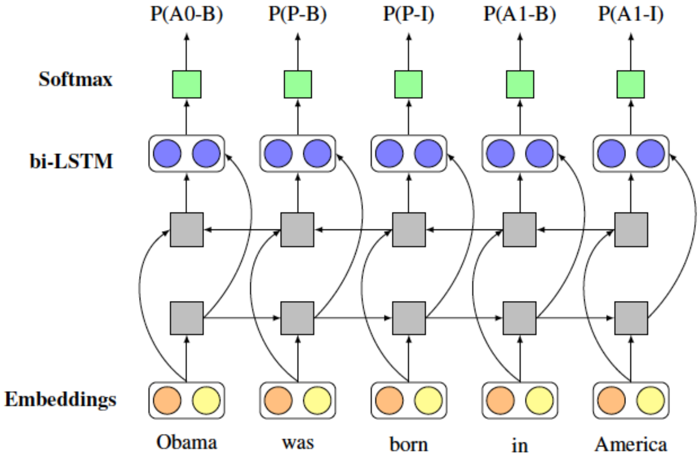


Figure 12: Architecture of the OpenIE model in Stanovsky et al. (2018)

4 Experiments

Following the success of pre-trained Transformer-based language models and taking as a reference the state-of-the-art systems for claim verification, we have decided to investigate if the integration of semantic knowledge to the inference module can improve the task of claim verification.

To this goal, we perform several experiments that will be described in the following sections. First, we use the base BERT model (Devlin et al., 2019) to perform the inference (Section 4.1). Then, we leverage the SemBERT architecture (Zhang et al., 2020a) to incorporate the Semantic Role Labels (Section 4.2). Finally, we apply the same SemBERT architecture to add linguistic knowledge extracted in the form of Open Information Extraction (Etzioni et al., 2008) triples (Section 4.3).

4.1 Baseline: a BERT model

The sequence classification model from BERT takes two sequence inputs separated by the special token *[SEP]*. The first sequence of our input is always the text of the claim (previously encoded into token ids). We tried two different ways to structure the evidences (the part after the *[SEP]* token).

In the first experiment, an input for each evidence was created, encoding the sentence like *claim_text [SEP] evidence_text*. This resulted in 5 (possibly) different predicted labels, one per each claim-evidence pair. For this reason, a voting system which picked the label that occurred most frequently was implemented at the end of the training pipeline. The results from this first implementation were unsatisfactory, most likely because not all evidences contained actual information to get to the right label. Let's recall that these are automatically extracted evidences from Hanselowski et al. (2018b), which means they are not all necessarily relevant.

An alternative way to structure the data consisted in concatenating all the evidences in a single string. The input to the BERT model looked like *claim_text [SEP] evidence_text. evidence_text. evidence_text. evidence_text. evidence_text.* The concatenated evidence was considered a better structure for our input, given that the label was extracted by taking into account every evidence available. The concatenated structure resulted in a long input, so we set the maximum sequence length to 250 tokens. We used the BERT tokenizer to encode the input. For training, we gave the model 4 epochs with a batch size of 20, we used the AdamW optimizer (Loshchilov and Hutter, 2019) and a linear scheduler, which linearly increases during the warmup period until it reaches the learning rate set to 2e-5 and then decreases linearly. These hyper-parameters were selected based on the results obtained by Zhang et al. (2020a).

The model described above will be called from now on bert_base, and is the baseline for our experiments. This is a strong baseline that has a 73.82 label accuracy. Next sections will evaluate if adding semantic information to this baseline can improve the results.

4.2 Incorporating SRL

Previous work had shown some improvement in NLI tasks when incorporating Semantic Role Labels (SRL) to Transformer architectures (Zhong et al., 2020; Zhang et al., 2020a) (explained in Sections 2.4 and 2.3). We decided to bring these findings to the task of claim verification by using the SemBERT architecture (see Section 3.3).

On first instance, we trained a model with all the semantic roles (from now on we will call them tags) retrieved by the AllenNLP parser. This resulted in a tags-vocabulary of size 22 (including the special tokens *[SEP]*, *[CLS]* and *[PAD]*), so the encoding layer contained 22 contextualized embeddings of length 10 (see the tags in Table 2).

In this case, the structure of the input was also the claim followed by *[SEP]* and the concatenation of all the retrieved evidences. We kept the maximum sequence length to 250. The original paper of SemBERT is tested in SNLI, which encodes pairs of sentences. For this reason, they conclude that allowing for a maximum of 3 predicate-argument structures is enough. In our case, given that we usually have around 6 sentences (1 claim and 5 evidences), we allow for the system to have up to 12 predicate-argument structures.

The architecture of this model is presented in Figure 13. Just like in the BERT model, the input of the top part of the diagram is the concatenation of the claim and the evidences, which is encoded with BERT (sub-)word embeddings. These sub-words (the tokenised units that BERT produces) are then reconstructed to become full word representations by using a convolutional layer. The lower part of the diagram shows the SRL part. For each proposition in the original input (up to 12 propositions), the semantic tags are given as input and encoded using tag embeddings. Then, a linear layer reduces the dimensionality of these 12 representation to 1. The result of the upper and lower part of the diagram are then concatenated, and are used to obtain the final decision (see Section 3.3 for more details).

The results of the base SemBERT experiment improved the performance of our baseline by reaching a 75.05 label accuracy.

4.2.1 Mapping SRL Tags

Given that the set of tags was quite large, it was considered that the sparsity of the SRL data could be preventing the model from learning patterns. We decided to make additional experiments reducing the set of tags by doing two different mappings. One mapping reduced the amount of tags by removing the positional part of the tags, which was given in BIO notation (e.g. I- B-), and reducing the amount of modifier arguments to just *temporal*, *location* or *other modifiers*, leaving a total of 13 tags. We call this mapping *tags1*, and the correspondence with the tags of the first model can be seen in Figure 2. The second tag set came from using the mapping of the DREAM system (Zhong et al., 2020), which additionally reduces all the ARG tags to a single *argument* tag, leaving a total of 8 tags. The correspondence can be seen in Table 2.

These new models slightly improved the performance of the previous ones. The sembert_tags1 model obtained a 75.37 label accuracy, while the sembert_DREAM model ob-

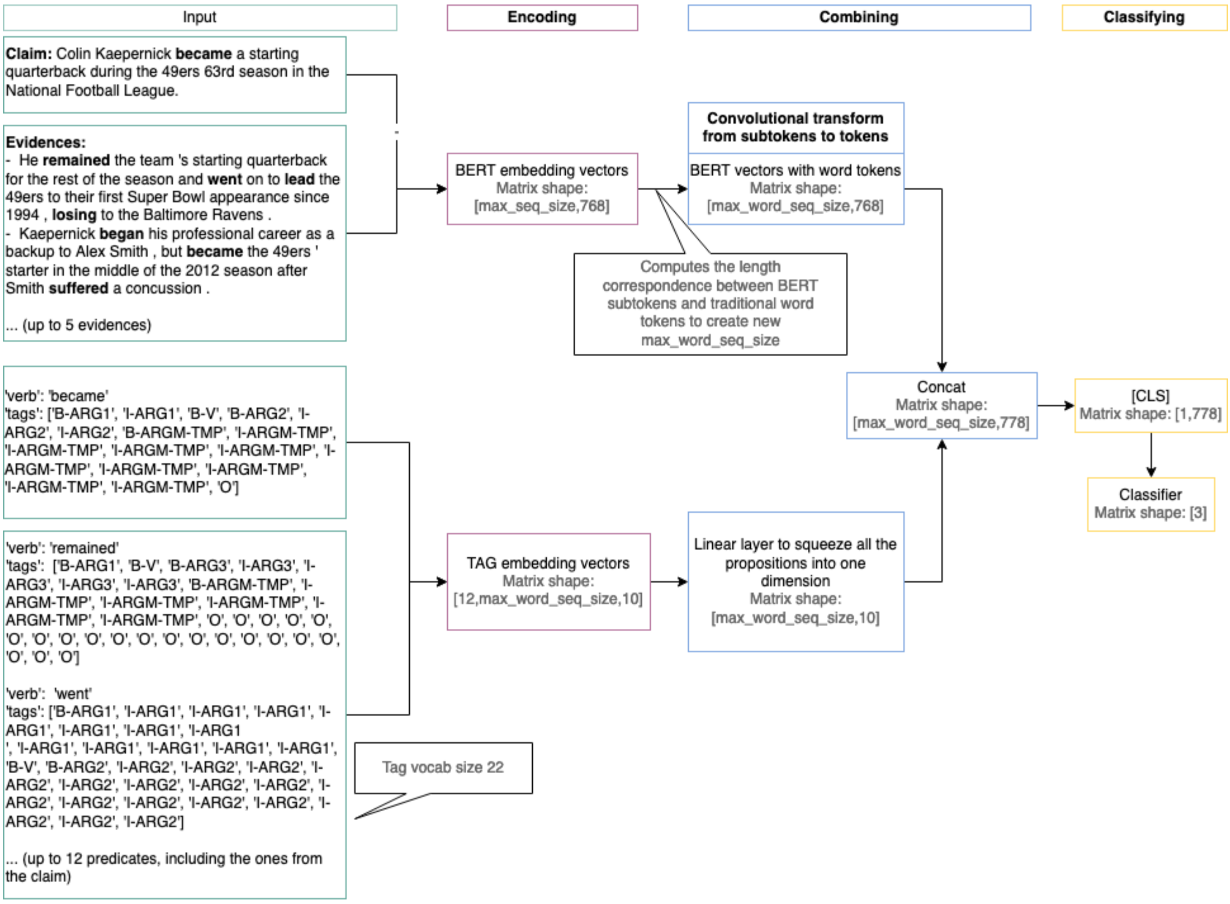

Input
Encoding
Combining
Classifying
Claim: Colin Kaepernick became a starting quarterback during the 49ers 63rd season in the National Football League.
Evidences:
- He remained the team 's starting quarterback for the rest of the season and went on to lead the 49ers to their first Super Bowl appearance since 1994 , losing to the Baltimore Ravens .
- Kaepernick began his professional career as a backup to Alex Smith , but became the 49ers ' starter in the middle of the 2012 season after Smith suffered a concussion .
... (up to 5 evidences)
'verb': 'became'
'tags': ['B-ARG1', 'I-ARG1', 'B-V', 'B-ARG2', 'I-ARG2', 'I-ARG2', 'B-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'O']
'verb': 'remained'
'tags': ['B-ARG1', 'B-V', 'B-ARG3', 'I-ARG3', 'I-ARG3', 'I-ARG3', 'B-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'I-ARGM-TMP', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O', 'O']
'verb': 'went'
'tags': ['B-ARG1', 'I-ARG1', 'I-ARG1', 'I-ARG1', 'I-ARG1', 'I-ARG1', 'I-ARG1', 'I-ARG1', 'I-ARG1', 'I-ARG1', 'I-ARG1', 'B-V', 'B-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2', 'I-ARG2']
... (up to 12 predicates, including the ones from the claim)
Tag vocab size 22
BERT embedding vectors
Matrix shape: [max_seq_size,768]
TAG embedding vectors
Matrix shape: [12,max_word_seq_size,10]
Convolutional transform from subtokens to tokens
BERT vectors with word tokens
Matrix shape: [max_word_seq_size,768]
Computes the length correspondence between BERT subtokens and traditional word tokens to create new max_word_seq_size
Linear layer to squeeze all the propositions into one dimension
Matrix shape: [max_word_seq_size,10]
Concat
Matrix shape: [max_word_seq_size,778]
[CLS]
Matrix shape: [1,778]
Classifier
Matrix shape: [3]

Figure 13: Architecture of SemBERT for claim verification

tained an accuracy of 75.12. Even if the difference in performance was not big, all the subsequent experiments are done using *tags1* mapping, as a smaller set of tags also helps in understanding the model.

4.2.2 Adding an Attention Mechanism

The given SemBERT model uses a linear layer to squeeze all the 12 predicates into one. That is needed to delete the multiple predicates dimension and be able to concatenate the representation coming from the SRL to the one produced by BERT (see Section 3.3). We hypothesized that this linear layer could be replaced by an attention mechanism that allowed evidences to reason between them, inspired in the self-attention mechanism from the GEAR system (Zhou et al., 2019), described in Section 2.3.

This self-attention mechanism concatenates the vectors of each predicate in pairs, to then compute self-attention between them and use that information to reshape the 12 representations into one using a linear layer. We used the *tags1* mapping and call the model sembert_tags1_att.

All Tags	Tags1 Tags	DREAM Tags
O	O	O
B-V	V	verb
I-V	V	verb
B-ARG0	ARG0	argument
I-ARG0	ARG0	argument
B-ARG1	ARG1	argument
I-ARG1	ARG1	argument
B-ARG2	ARG2	argument
I-ARG2	ARG2	argument
B-ARG4	ARG4	argument
I-ARG4	ARG4	argument
B-ARGM-TMP	TMP	temporal
I-ARGM-TMP	TMP	temporal
B-ARGM-LOC	LOC	location
I-ARGM-LOC	LOC	location
B-ARGM-CAU	ARGM	argument
I-ARGM-CAU	ARGM	argument
B-ARGM-PRP	ARGM	argument
I-ARGM-PRP	ARGM	argument

Table 2: Mapping between sets of tags

Contrary to what we had hypothesized, the new self-attention mechanism did not lead to an improvement of the model, but it did not decrease much either (75.15 label accuracy). Given that this model was more complex than the previous ones, we decided to stick to the sembert_tags1 model as our best model so far.

4.3 Incorporating Open Information Extraction

SRL is the most widespread semantic parsing, but not the only one. Open Information Extraction (OpenIE) was designed to extract unknown relations from millions of documents. Its first implementations were framed as a task of hand-crafted pattern-matching, which later evolved into the creation of automated systems (Etzioni et al., 2008). Stanovsky et al. (2015) were the first to propose using these representations as an intermediate structure for other tasks. In this work, they found that OpenIE could potentially be effective for the task of text comprehension. This finding motivated us to develop a SemBERT system that uses OpenIE instead of SRL as an underlying structure.

For this experiment we have used the AllenNLP OpenIE parser, which is the implementation of Stanovsky et al. (2018). After parsing, we have kept the tags *argument*, *verb* and $O-O$ meaning that the word is not part of the predicate. This system is trained in the exact same way as the one described in Section 4.2.

The obtained results were better than the base BERT baseline, but they did not improve the performance of the SemBERT model with SRL tags. The label accuacy was 74.34, indicating that the simpler tags of OpenIE (just 3) did help the model, but missed some information contained in SRL.

5 Evaluation

Having presented our experiments, in this section we report our results. We first report the accuracy of all our models, and then focus on comparing examples of our baseline and our best model (Section 5.1). Then, we perform some explainability tests to evaluate the capabilities of these models (Section 5.2). Finally, we report the results of our best model on the test set and compare it to previous work (Section 5.3).

5.1 Model Comparison

In our experiments we tried several variations of the SemBERT model, by using different sets of semantic tags and adding different reasoning mechanisms. In Table 3, we put together the accuracy of the predictions of all these models, which have already been reported in Section 4. We observe that all the SemBERT experiments have a better performance than the BERT baseline. This difference is of 1 to 2 percentage points.

	Accuracy
bert_base (baseline)	73.82
sembert_base	75.06
sembert_tags1	**75.37**
sembert_dream	75.12
sembert_attention_tags1	74.92
sembert_openie	74.34

Table 3: Results from all the models in the FEVER dev set

Our best model is the SemBERT model with the SRL set tags1. The confusion matrices in Figures 14 and 15 show that the improvement does not come from a clear refinement of one single class. Instead, we see that many instances that had been wrongly classified as Supports in BERT are now correctly classified as NEI, and many instances predicted as NEI by BERT are now correctly refuted by SemBERT. This happens the other way around too: BERT classifies as NEI a lot of instances that had wrongly been classified by SemBERT as Refutes, and classifies as Supports instances that SemBERT had wrongly classified as NEI. In general we observe that, while BERT is biased towards predicting Supports, SemBERT is more inclined to predicting Refutes, and both systems get easily confused with the class NEI.

The general trends shown in the confusion matrices indicate that the improvement of SemBERT over BERT is not unidirectional. However, it does not show which inference capabilities each of these models have so we will qualitatively analyze some examples of the outputs of these models.

In Table 4, we can see examples that both models got correctly. The first example of this table claims that *Aristotle* had spent time in the city of Athens. Given two clear evidence sentences that place this person in this city, both systems output Supports. In the second example, the answer is also very straightforward, as the claim says *Telemundo* is an

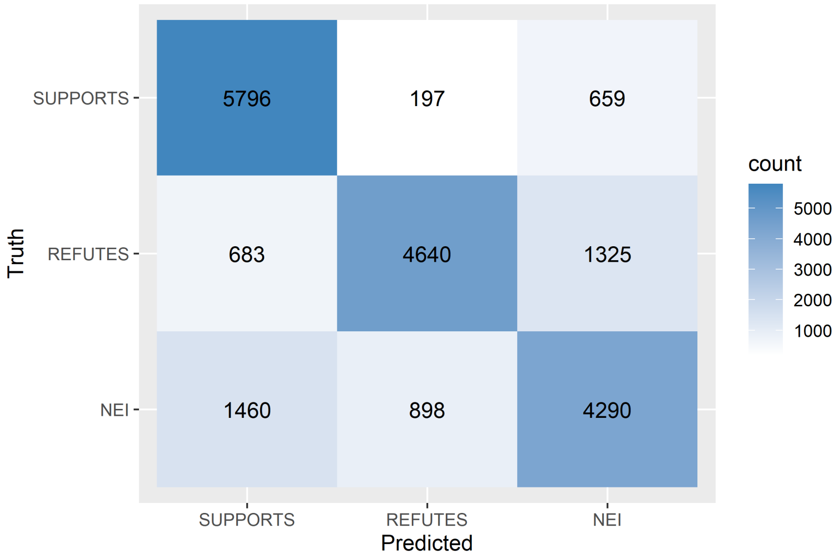


Figure 14: Confusion matrix of the predictions with bert_base

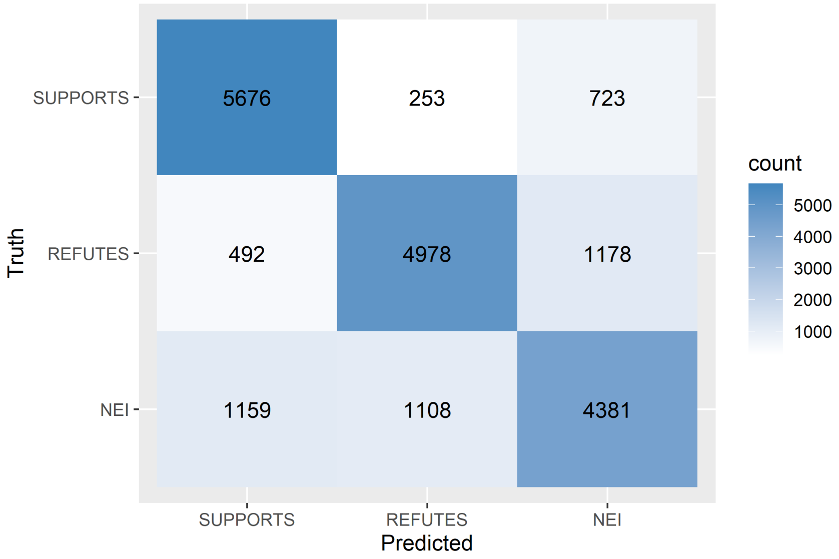


Figure 15: Confusion matrix of the predictions with sembert_tags1

English-language channel and the first evidence sentence already claims that it actually is in *Spanish-language*. Finally, the last example is more complicated because, even though the system retrieved many evidence pieces that speak about the style of the *Paris's album*, none of them mention anything related to *German*. Both systems output NEI rightly.

In Table 5, we see a couple of examples that SemBERT correctly refuted, but BERT decided to label as Supports. In the first example, the claim states that *Bert V. Royal* had directed the film *Easy A*. Looking at the first evidence, it is possible to see that

Label	Instance
SUPPORTS	**Claim**: Aristotle spent time in Athens. **Evidence**: At seventeen or eighteen years of age , he joined Plato 's Academy in Athens and remained there until the age of thirty-seven c. 347 BC. **Evidence**: Shortly after Plato died , Aristotle left Athens and , at the request of Philip II of Macedon , tutored Alexander the Great beginning in 343 BC .
REFUTES	**Claim**: Telemundo is a English-language television network. **Evidence**: Telemundo is an American Spanish-language terrestrial television network owned by Comcast through the NBCUniversal division NBCUniversal Telemundo Enterprises . **Evidence**: It is the second largest provider of Spanish content nationwide behind American competitor Univision , with programming syndicated worldwide to more than 100 countries in over 35 languages .
NEI	**Claim**: Paris (Paris Hilton album) incorporates elements of German. **Evidence**: It also incorporates elements of other genres , such as reggae , soul and pop rock , in its production . **Evidence**: Musically , Paris is a pop and R&B album that is influenced by hip hop . **Evidence**: The self-titled album , Paris , was released worldwide on August 22 , 2006 . **Evidence**: Paris is the debut studio album by American media personality , actress and singer Paris Hilton .

Table 4: Correct examples in both bert_base and sembert_tags1

the BERT model probably got confused because the first evidence does include both the predicate *directed by* and the name of *Bert V. Royal*. However, if we take into account the semantic structure of the sentence, we realise that the name and the predicate do not belong to the same proposition, and instead the name *Bert V. Royal* is an argument of the predicate *written by*. This is an interesting example where the semantic structure added in SemBERT seems to have an influence. The second example of this table is also a semantic complication given by the expression *the first*.

When reviewing the instances that BERT had correctly labeled as Supports and SemBERT had refuted, we observe that many of these claims required certain numerical reasoning. In the first example in Table 6 the system had to reason that *more than 70%* has to be true if *more than 80%* is true. In the second example, the claim states that the series took place in the *1970s*, and both evidences claim that the series was in fact set in *1979*. This hypothesis will be further investigated later, as we do not find any reason why BERT would be better at mathematical reasoning.

From the qualitative review of the instances which had been labelled right and wrong by BERT and SemBERT we have observed that SemBERT seems to have certain ability

Label	Instance
REFUTES	**Claim**: Easy A is directed by Bert V. Royal. **Evidence**: Easy A -LRB- stylized as easy A -RRB- is a 2010 American teen comedy film directed by Will Gluck , written by Bert V. Royal and starring Emma Stone , Stanley Tucci , Patricia Clarkson , Thomas Haden Church , Dan Byrd , Amanda Bynes , Penn Badgley , Cam Gigandet , Lisa Kudrow and Aly Michalka . **Evidence**: Bert V. Royal , Jr. -LRB- born October 14 , 1977 -RRB- is an American screenwriter , playwright , and former casting director .
REFUTES	**Claim**: Marco Polo was not a European. **Evidence**: Marco Polo was not the first European to reach China -LRB- see Europeans in Medieval China -RRB- , but he was the first to leave a detailed chronicle of his experience .

Table 5: Examples that are correct for sembert_tags1 and not for bert_base

Label	Instance
SUPPORTS	**Claim**: The Indian Army comprises more than 70% of the country's active defense personnel. **Evidence**: It is an all-volunteer force and comprises more than 80 % of the country 's active defence personnel .
SUPPORTS	**Claim**: Season 2 of Fargo takes place in the 1970s. **Evidence**: A prequel to the events in its first season , season two of Fargo takes place in the Midwestern United States in March 1979 . **Evidence**: The second season , set in 1979 and starring Kirsten Dunst , Patrick Wilson , Jesse Plemons , Jean Smart , and Ted Danson , was met with even greater acclaim .

Table 6: Examples that are correct for bert_base and not for sembert_tags1

to understand semantically complex sentences, while BERT is better at numerical reasoning. However, it is difficult to confirm these observations without looking deeper into the decision making process of the systems. For this reason, we have performed a set of explainability tests.

5.2 Explainability Tests

The explainability tests that we perform in this project are gradient-based tests (saliency scores) and adversarial attacks. We will use the same examples presented above.

5.2.1 Saliency Scores

Extracting the saliency of each of the tokens given as input is not a trivial task for deep-learning models. Simonyan et al. (2014) proposed to compute them as the gradient of the output with respect to each input. Later improvements to this technique proposed to then multiply these gradients to the input (*InputX-Gradient*), or to overwrite the gradients of the ReLU functions in order to prevent negative gradients from being propagated (*Guided Backpropagation*) (Kindermans et al., 2016; Springenberg et al., 2015).

We will use the saliency scores proposed above to get a better grasp of where the model focuses in order to make its inference decisions. For an interpretable output we want to have one saliency value for each token. Given that the last layer that we can compute the gradients for is the embedding layer, we will get one gradient for each value in the embedding of each token. In order to aggregate these values and get one single value per token we will use the L2 norm (Atanasova et al., 2020a).

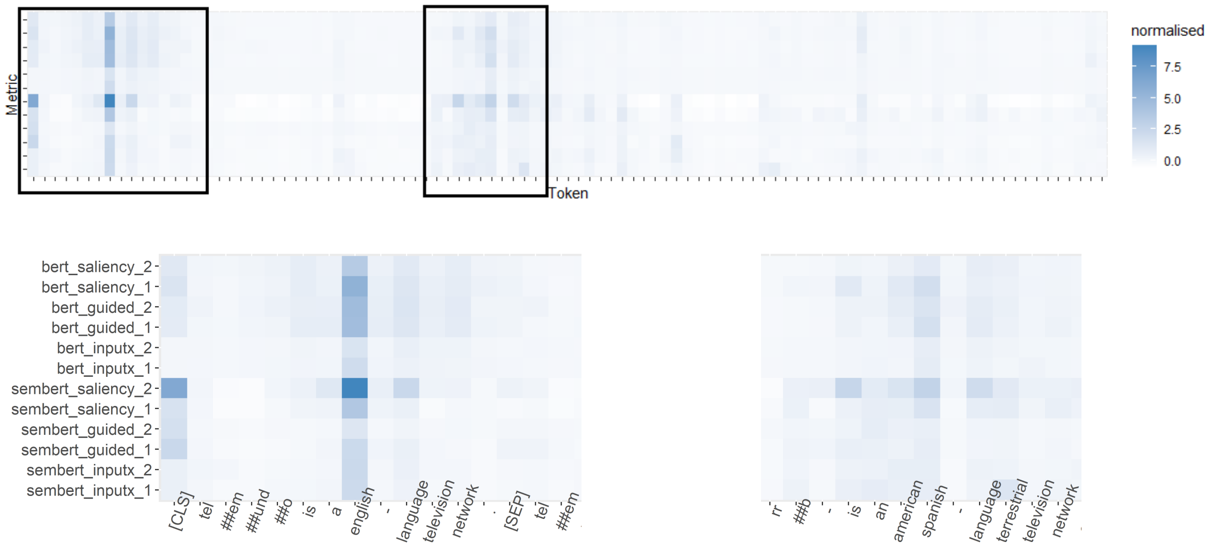


Figure 16: Saliency Scores of the *Telemundo* example with BERT and SemBERT. The above plot shows the entire claim and evidence input, and the plots under it zoom into the relevant parts, delimited with black frames above.

In Figure 16, the visualisation of the saliency scores for one of the examples is shown. We compute each of the measures presented above (Saliency, InputxGradient and Guided Backpropagation) two times, in order to account for variability, and aggregate the results for each token using L2 norm. The three metrics have been normalised. It can be seen that the tokens found to be more salient are *English* in the claim, and *Spanish* in the evidence, which matches with what a human would focus on while verifying this claim. Additionally, it can also be seen that both the BERT and SemBERT model agree with these rationales.

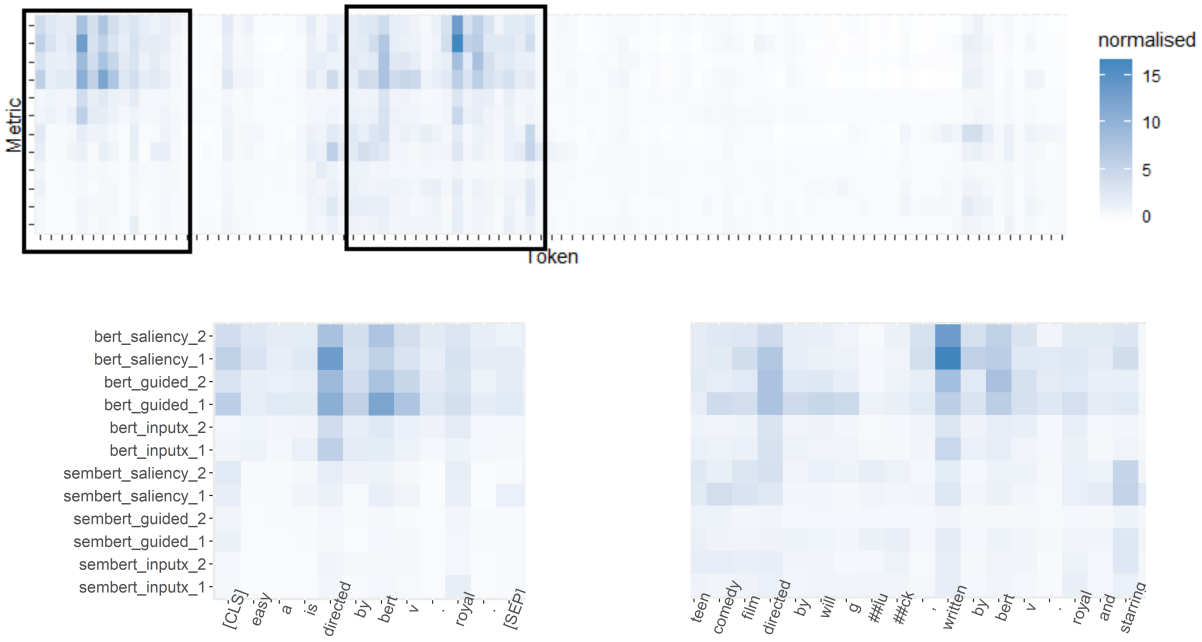


Figure 17: Saliency Scores of the *Easy A* example with BERT and SemBERT. The above plot shows the entire claim and evidence input, and the plots under it zoom into the relevant parts, delimited with black frames above.

The first example of Table 5 is interesting, as it looks as if SemBERT is taking advantage of the given semantic structure to correctly predict Refutes. Let's recall this example is wrongly labelled as Supports by BERT. In Figure 17 we can see that the saliency scores in the BERT model rely on the tokens *directed* and *bert* in the claim, and *directed*, *written* and *bert* in the evidence part. The SemBERT model seems to not have any salient token in the claim, and only the words *written* and *starring* seem to be slightly relevant in the evidence. It is not clear from this plot where does the output come from in SemBERT. However, it has to be taken into account that the SemBERT model also has the semantic structure as additional input, which is not shown in this plot. We could hypothesize that some of the focus of the model is in the semantic part of the input, but this can not be concluded from the displayed saliency scores.

Finally, to observe if the numbers are better dealt by BERT than by SemBERT, we show the first example of Table 6 in Figure 18. In this visualisation it becomes clear that the decision taken by the BERT model relies on the numbers 70 and 80. SemBERT, instead, seems to be trying to verify some other information, as it puts the relevance in the tokens *defense* in the claim, and *volunteer* and again *defense* in the evidence. This remark seems to reinforce the observation that BERT might be better in numerical reasoning than SemBERT, which we pointed in the previous section.

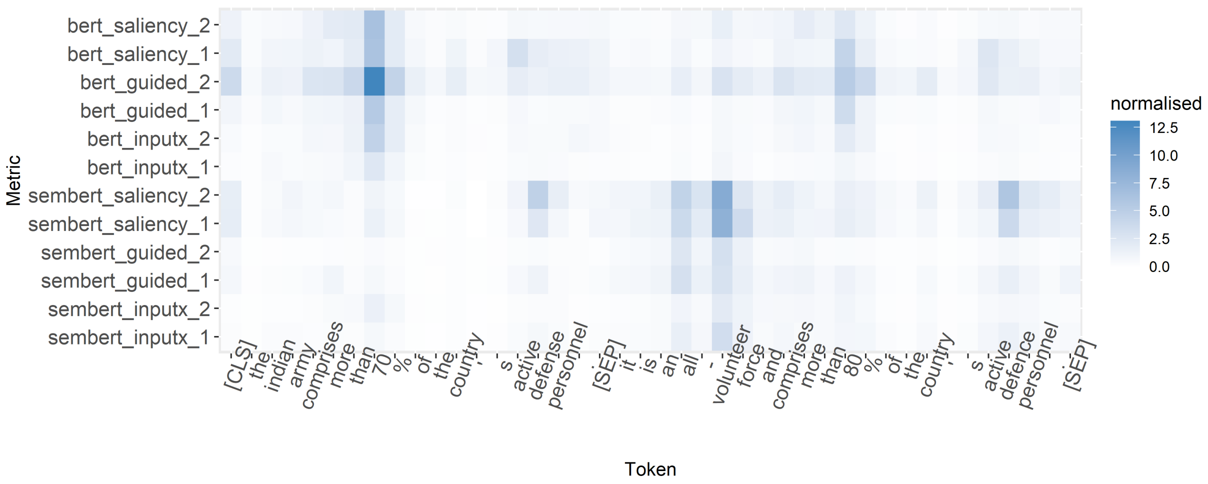


Figure 18: Saliency Scores of the *Indian Army* example with BERT and SemBERT. The above plot shows the entire claim and evidence input, and the plots under it zoom into the relevant parts, delimited with black frames above.

5.2.2 Adversarial Attacks

Another explainability technique that has been proposed in previous work is changing the input in order to assess the influence that it has over the output. This has been done both by removing input tokens systematically (Zeiler and Fergus, 2014), and by altering the input instances to generate adversarial attacks which can show what the model actually understands (Ribeiro et al., 2018; Ebrahimi et al., 2018). In this section, we are going to create some manual adversarial attacks in order to test the capabilities of our models.

Ribeiro et al. (2020) designed a CheckList to be used for testing NLP models looking at the different desired capabilities using adversarial attacks. These can be either label-preserving modifications (e.g. *I like apples* versus *I still like apples*), or label-changing tests (e.g. *I like apples* versus *I don't like apples*). Following their approach, we are manually going to generate instances to test capabilities such as vocabulary+POS, NER, negation, semantic structure and logic.

We start by testing basic capabilities for the instances both models got right. A first check should ensure that the given labels are not random by creating attacks which change the original label. We modify the claims in Table 4 to:

- *Telemundo is a Spanish-language television network.* ← Supports
- *Telemundo is an American television network.* ← Supports
- *Aristotle never spent time in Athens.* ← Refutes
- *Paris (Paris Hilton Album) incorporates elements of soul.* ← Supports
- We add *German* into the list of elements that the Paris album incorporates in the first evidence. ← Supports

In these tests, both models change the labels as expected with no errors. We also perform some tests which should preserve the initial labels, such as: *Telemundo is a Chinese-language television network*, or adding the word *German* as the nationality of Paris Hilton in the third evidence. These changes seem to be dealt correctly by both models. These first tests show certain capacities of the model to deal with changes in vocabulary+POS (e.g. removing *-language* or using German as a nationality), NER (e.g. changing English to Spanish and Chinese), and negation (e.g. with the word *never* in the Aristotle example).

We then want to test more complex behaviour that was not dealt the same way by both models. Using the first example in Table 5 (the one about *Easy A*), we want to investigate if SemBERT is in fact dealing correctly with the complex semantic structure.

The original instance is:

- **Claim**: *Easy A is directed by Bert V. Royal*
- **Evidence 1**: *Easy A -LRB- stylized as easy A -RRB- is a 2010 American teen comedy film directed by Will Gluck , written by Bert V. Royal and starring Emma Stone , Stanley Tucci , Patricia Clarkson , Thomas Haden Church , Dan Byrd , Amanda Bynes , Penn Badgley , Cam Gigandet , Lisa Kudrow and Aly Michalka .*
- **Evidence 2**: *Bert V. Royal , Jr. -LRB- born October 14 , 1977 -RRB- is an American screenwriter , playwright , and former casting director .*

SemBERT correctly labeled this instance as Refutes, but BERT labeled it as Supports. We start by checking that the Refutes label of SemBERT is not random by changing the claim to *Easy A is written by Bert V. Royal.* SemBERT passes this test and outputs Supports. Following the tests for semantic structure in Ribeiro et al. (2020)'s CheckList, we modify evidence 1 by changing the order of the propositions, swapping them to active form, and creating symmetric relations. The new versions of the evidence are:

1. **Order change**: *Easy A -LRB- stylized as easy A -RRB- is a 2010 American teen comedy film* ***written by Bert V. Royal, directed by Will Gluck*** *, and starring Emma Stone , Stanley Tucci , Patricia Clarkson , Thomas Haden Church , Dan Byrd , Amanda Bynes , Penn Badgley , Cam Gigandet , Lisa Kudrow and Aly Michalka.* ← Refutes
2. **Order change**: *Easy A -LRB- stylized as easy A -RRB- is a 2010 American teen comedy film written by Bert V. Royal , starring Emma Stone , Stanley Tucci , Patricia Clarkson , Thomas Haden Church , Dan Byrd , Amanda Bynes , Penn Badgley , Cam Gigandet , Lisa Kudrow and Aly Michalka ,* ***and directed by Will Gluck****.* ← Refutes
3. **Symmetric relation**: *Easy A -LRB- stylized as easy A -RRB- is a 2010 American teen comedy film* ***directed by Will Gluck and Bert V. Royal*** *and starring Emma Stone , Stanley Tucci , Patricia Clarkson , Thomas Haden Church , Dan Byrd , Amanda Bynes , Penn Badgley , Cam Gigandet , Lisa Kudrow and Aly Michalka.* ← Supports

4. **Remove the *written by* proposition**: *Easy A -LRB- stylized as easy A -RRB- is a 2010 American teen comedy film directed by Will Gluck , and starring Emma Stone , Stanley Tucci , Patricia Clarkson , Thomas Haden Church , Dan Byrd , Amanda Bynes , Penn Badgley , Cam Gigandet , Lisa Kudrow and Aly Michalka.* ← Refutes

5. **Active form**: *Easy A -LRB- stylized as easy A -RRB- is a 2010 American teen comedy film.* ***Will Gluck directed the film , and Bert V. Royal wrote it.*** ← Refutes

With all the variations of evidence 1 presented above, SemBERT always outputs the right label, while BERT just outputs the right label in the last piece of evidence, which contains the same information but in active form. These tests suggest that SemBERT does have capabilities regarding semantic structure that are missing in BERT. However, more systematic tests should be performed in this direction.

It has to be noted that, when we remove the proposition *directed by Will Gluck* from evidence 1 the label should become NotEnoughInfo. Both models fail at this prediction and instead output Refutes, again showing that NEI is the most difficult class.

Continuing with these experiments, we try to investigate the reason for the failure of BERT in the sentence *Marco Polo was not a European.* What we find by making changes to both the claim and the evidences, is that, for both models, whenever there is the word *not* in the claim, the model outputs Refutes, and whenever we remove the *not* the label becomes Supports. It is true that BERT labelled the original instance as Supports, but that seems to be the exception and not the norm. Based on this observation, we decide to try adding the word *not* to the other claims we have just investigated, finding that both *Telemundo is not a English-language television network* and *Easy A is not directed by Bert V. Royal* are wrongly labelled as Refutes for both models. What we find here is a clear bias towards Refutes whenever there is the word *not* in the claim. It seems like this is an issue coming from the creation of the dataset. It has to be noted that the guidelines of FEVER specifically required to try to avoid trivial negations with *not* (see Section 3.1). However, this guideline does not seem to have prevented it from happening.

Finally, we want to review the capabilities of the model to deal with numerical reasoning and logic. Before we hypothesised that BERT might be better at this task based on the examples from Table 6. Going back to these examples, we created several adversarial attacks that should prove if the numerical reasoning is actually happening. The first example required the model to reason that if it is true that the army comprises more than 80% of the active defense personnel, it should also be true that the army comprises 70% of the active defense personnel. We try to create examples changing these numbers, spelling the numbers or changing the comparative *more than* to *less than.* The attacks are:

1. **Change the number**: *The Indian Army comprises more than 60 % of the country's active defense personnel.* ← Supports

2. **Change the number**: *The Indian Army comprises more than 20 % of the country's active defense personnel.* ← Supports

3. **Change the number**: *The Indian Army comprises more than 90 % of the country's active defense personnel.* ← Refutes

4. **Change the numbers**: Exchange 70 and 80 in the claim and the evidence.← Refutes

5. **Remove *more than***: *The Indian Army comprises 70 % of the country's active defense personnel.* ← Refutes

6. **Use *less than***: *The Indian Army comprises less than 70 % of the country's active defense personnel.* ← Refutes

7. **Spell numbers**: *The Indian Army comprises more than seventy percent of the country's active defense personnel.* (also in the evidence we write *eighty*) ← Supports

What we observe from these alternative instances is that, in general, BERT and SemBERT output very similar results. When changing the numbers, SemBERT gets the first example right and BERT gets it wrong, then they both get examples 2 and 3 wrong, and both get right example 4. These are inconclusive observations, which shows that there is not a clear numerical reasoning behind the models. In examples 5 and 6, both models get the labels right, showing that the model does get comparative clauses. Finally, spelled numbers seem to also be dealt with rightly.

We do a test trial with the example *Season 2 of Fargo takes place in the 1970s.* The alternative instances are:

1. **Change the numbers**: Changing the year in the evidences to *1982.* ← Refutes

2. **Change the subject**: *Season 1 of Fargo takes place in the 1970s.* ← Refutes

3. **Un-spell the numbers**: In the evidences we change *second season* to *Season 2.* ← Supports

4. **Spell the numbers**: *Season 2 of Fargo takes place in the seventies.* ← Supports

The results of these attacks to the numerical reasoning are also inconclusive. SemBERT gets right the first instance, they both get wrong the second one, and BERT gets right the last two examples. From these observations, we again conclude that numerical reasoning is not handed well by any of the models, and this should be an issue to focus on future work.

5.3 Generalisation of the models

While explainability tests are important to assess the relation between the system's reasoning and human judgement, the end goal of a NLP system should be to be able to perform well on unseen data. For this reason, many datasets include a blind test set which should

	Evidence F1	**Label accuracy**	**Fever Score**
UKP-Athene	36.97	65.46	61.58
GEAR	36.87	71.60	67.10
DREAM	39.45	76.85	70.60
bert_base (baseline)	36.87	70.86	65.52
sembert_tags1	36.87	**72.18**	67.16

Table 7: Results on the test set of my best model and previous models

only be used for final evaluation. In the case of FEVER, the models can be evaluated in the test set in their leaderboard in Codalab.[6]

The evaluations in the test set can be seen in Table 7. In the unseen data, the SemBERT model still outperforms the BERT baseline by 1.3 percentage points in label accuracy. Both models drop around 3 percentage points with respect to the development set. Additionally, we also report the results on the test set of previous work such as UKP-Athene (Hanselowski et al., 2018b), GEAR (Zhou et al., 2019), and DREAM (Zhong et al., 2020). For our model, we used the evidences extracted by UKP-Athene, and some pre-processing scripts from GEAR, which explains why all three models have (almost) the same F1 for evidence retrieval. Our model outperforms both of these models in the inference module. We got inspired by the work in DREAM to integrate semantic information for reasoning. However, instead of using a graph-based approach, we used the SemBERT architecture to incorporate the semantic information. As observed, DREAM performs better than our model, suggesting that graph-based architectures might be a better representation for semantic information. Even though the Codalab leaderboard has better-scoring submissions than DREAM, this is the highest-scoring published system so far.

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

6 Conclusion

In this work we have investigated if semantic information could facilitate the reasoning process when inferring the truth label of a claim given some pieces of evidence. To this goal, we have used two different semantic parsers and the architecture of the pre-trained model SemBERT (Zhang et al., 2020a). For our experiments, we have used the FEVER dataset (Thorne et al., 2018), which requires building a model that, given some pieces of evidence, can output if a claim is supported, refuted, or the evidence does not give enough information.

We have performed several experiments on top of the SemBERT architecture, such as training models with different kinds of semantic information, different sets of semantic tags, and with an additional attention mechanism to represent the semantic information. In terms of label accuracy, all our experiments have outperformed the baseline, which was a BERT model with no additional semantic information. Our best model uses Semantic Role Labels and a set of 13 different tags, with no additional attention mechanism. This model achieves a label accuracy of 75.37 on the development set and 72.18 on the test set, outperforming the baseline by 1.5 and 1.3 percentage points respectively.

To better understand the contribution of the semantic information, we have performed some explainability tests with our best model. These have shown that the SRL knowledge might be contributing to guiding the model in semantically complex sentences that include several propositions or passive forms. Additionally, we have also found that, as have been pointed before, FEVER contains some linguistic cues that give both true and false hints to the model, such as the word *not*.

Our contributions in this work have been (1) performing annotations to understand the attributes of the FEVER dataset, (2) building a competitive system to deal with claim verification, (3) testing the impact of semantic information for NLI, and (4) performing explainability tests to understand the contributions of the additional semantic information. All the code used for this project is available in the Github repository.[7]

As highlighted during the annotation process, FEVER is a synthetic dataset which does not include as many semantically complex sentences as naturally-occurring claims. Future work should focus on investigating if the semantic capabilities acquired by integrating semantic knowledge contribute to claim verification in naturally-occurring claims. The lack of claims that require temporal and mathematical reasoning is another issue that differentiates FEVER from datasets with naturally-occurring claims. Future work should also deal with these issues in order to make it possible to use NLP systems for claim verification in real-world scenarios, such as fact-checking of news and public claims in general.

To keep moving towards systems that can contribute to the work of fact-checkers, future research on claim verification should take two directions. On the one hand, there is a need to develop large datasets that are more similar to naturally-occurring claims and contain less linguistic cues. On the other hand, NLI models for claim verification should output

[7] https://github.com/BlancaCalvo/Claim-Verification-FakeNews

more explanatory justifications to their conclusions, which would make these systems more trust-worthy.

In this work, we have not dealt with the task of evidence retrieval. In FEVER, this task is limited by the static Wikipedia database that comes with the dataset. However, in real-world scenarios defining the boundaries of what is trust-worthy information is a challenge that goes beyond research in NLP and reaches the fields of journalism, politics and even philosophy. The non-static nature of what is a true fact is an additional challenge to evidence retrieval.

References

D. D. Ahn, V. Jijkoun, G. A. Mishne, K. E. Müller, M. de Rijke, and K. S. Schlobach. Using Wikipedia at the TREC QA Track. 2005. URL https://dare.uva.nl/search?arno.record.id=165576.

Firoj Alam, Fahim Dalvi, Shaden Shaar, Nadir Durrani, Hamdy Mubarak, Alex Nikolov, Giovanni Da San Martino, Ahmed Abdelali, Hassan Sajjad, Kareem Darwish, and Preslav Nakov. Fighting the COVID-19 Infodemic in Social Media: A Holistic Perspective and a Call to Arms. *arXiv:2007.07996 [cs]*, April 2021. URL http://arxiv.org/abs/2007.07996. arXiv: 2007.07996.

Hunt Allcott and Matthew Gentzkow. Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives*, 31(2):211–236, May 2017. ISSN 0895-3309. doi: 10.1257/jep.31.2.211. URL https://www.aeaweb.org/articles?id=10.1257%2Fjep.31.2.211&fbclid=IwAR04My3aiycypMJKSI58e84gDvdrodsB9fqCycH9YfepWDDDwT--fZnVPvo;%20https://www.nyu.edu/about/news-publications/news/2019/january/fake-news-shared-by-very-few--but-those-over-65-more-likely-to-p.html.

Daniel Andor, Luheng He, Kenton Lee, and Emily Pitler. Giving BERT a Calculator: Finding Operations and Arguments with Reading Comprehension. *arXiv:1909.00109 [cs]*, September 2019. URL http://arxiv.org/abs/1909.00109. arXiv: 1909.00109.

Gabor Angeli and Christopher 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, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1059. URL https://www.aclweb.org/anthology/D14-1059.

Fatma Arslan, Naeemul Hassan, Chengkai Li, and Mark Tremayne. A Benchmark Dataset of Check-worthy Factual Claims. *arXiv:2004.14425 [cs]*, April 2020. URL http://arxiv.org/abs/2004.14425.

Atanas Atanasov, Gianmarco De Francisci Morales, and Preslav Nakov. Predicting the Role of Political Trolls in Social Media. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 1023–1034, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/K19-1096. URL https://www.aclweb.org/anthology/K19-1096.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. A Diagnostic Study of Explainability Techniques for Text Classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3256–3274, 2020a.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. Generating Fact Checking Explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7352–7364, Online, July 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.656. URL `https://www.aclweb.org/anthology/2020.acl-main.656`.

Isabelle Augenstein, Christina Lioma, Dongsheng Wang, Lucas Chaves Lima, Casper Hansen, Christian Hansen, and Jakob Grue 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, 2019.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The Berkeley FrameNet Project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98/COLING '98, pages 86–90, USA, August 1998. Association for Computational Linguistics. doi: 10.3115/980845.980860. URL `https://doi.org/10.3115/980845.980860`.

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the Blanks: Distributional Similarity for Relation Learning. *arXiv e-prints*, 1906:arXiv:1906.03158, June 2019. URL `http://adsabs.harvard.edu/abs/2019arXiv190603158B`.

Ramy Baly, Mitra Mohtarami, James Glass, Lluís Màrquez, Alessandro Moschitti, and Preslav Nakov. Integrating Stance Detection and Fact Checking in a Unified Corpus. 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 21–27, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2004. URL `https://www.aclweb.org/anthology/N18-2004`.

Alberto Barron-Cedeno, Tamer Elsayed, Preslav Nakov, Giovanni Da San Martino, Maram Hasanain, Reem Suwaileh, Fatima Haouari, Nikolay Babulkov, Bayan Hamdan, Alex Nikolov, Shaden Shaar, and Zien Sheikh Ali. Overview of CheckThat! 2020: Automatic Identification and Verification of Claims in Social Media. *arXiv:2007.07997 [cs]*, July 2020. URL `http://arxiv.org/abs/2007.07997`. arXiv: 2007.07997.

Marco T. Bastos and Dan Mercea. The Brexit Botnet and User-Generated Hyperpartisan News. *Social Science Computer Review*, 37(1):38–54, February 2019. ISSN 0894-4393. doi: 10.1177/0894439317734157. URL `https://doi.org/10.1177/0894439317734157`. Publisher: SAGE Publications Inc.

Claire Bonial, Susan Brown, W. Corvey, Martha Palmer, Volha Petukhova, and Harry Bunt. An Exploratory Comparison of Thematic Roles in VerbNet and LIRICS. January 2011.

Johan Bos and Katja Markert. Recognising Textual Entailment with Robust Logical Inference. In Joaquin Quiñonero-Candela, Ido Dagan, Bernardo Magnini, and Florence d'Alché Buc, editors, *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, Lecture Notes in Computer Science, pages 404–426, Berlin, Heidelberg, 2006. Springer. ISBN 978-3-540-33428-6. doi: 10.1007/11736790_23.

Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher 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, 2015.

Chris J. C. Burges. From RankNet to LambdaRank to LambdaMART: An Overview. June 2010. URL `https://www.microsoft.com/en-us/research/publication/from-ranknet-to-lambdarank-to-lambdamart-an-overview/`.

Davide Buscaldi and Paolo Rosso. Mining Knowledge fromWikipedia for the Question Answering task. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy, May 2006. European Language Resources Association (ELRA). URL `http://www.lrec-conf.org/proceedings/lrec2006/pdf/332_pdf.pdf`.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine 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, 2017a.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana 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, 2017b.

Robin Cooper, Richard Crouch, Jan van Eijck, Chris Fox, Josef Genabith, Jan Jaspars, Hans Kamp, David Milward, Manfred Pinkal, Massimo Poesio, Steve Pulman, Ted Briscoe, Holger Maier, and Karsten Konrad. Using the Framework. March 1996.

Soheil Danesh, Tamara Sumner, and James H. Martin. SGRank: Combining Statistical and Graphical Methods to Improve the State of the Art in Unsupervised Keyphrase Extraction. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 117–126, Denver, Colorado, June 2015. Association for Computational Linguistics. doi: 10.18653/v1/S15-1013. URL `https://www.aclweb.org/anthology/S15-1013`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina 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, 2019.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. ERASER: A Benchmark to Evaluate Rationalized NLP Models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.408. URL `https://www.aclweb.org/anthology/2020.acl-main.408`.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. DROP: A Reading Comprehension Benchmark Requiring Discrete Reasoning Over Paragraphs. 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 2368–2378, 2019.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing 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, 2018.

Tamer Elsayed, Preslav Nakov, Alberto Barrón-Cedeño, Maram Hasanain, Reem Suwaileh, Giovanni Da San Martino, and Pepa Atanasova. Overview of the CLEF-2019 CheckThat! Lab: Automatic Identification and Verification of Claims. In Fabio Crestani, Martin Braschler, Jacques Savoy, Andreas Rauber, Henning Müller, David E. Losada, Gundula Heinatz Bürki, Linda Cappellato, and Nicola Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, Lecture Notes in Computer Science, pages 301–321, Cham, 2019. Springer International Publishing. ISBN 978-3-030-28577-7. doi: 10.1007/978-3-030-28577-7_25.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel Weld. Open Information Extraction from the Web. *Commun. ACM*, 51:68–74, December 2008. doi: 10.1145/1409360.1409378.

Charles J. Fillmore. Frame Semantics and the Nature of Language*. *Annals of the New York Academy of Sciences*, 280(1):20–32, 1976. ISSN 1749-6632. doi: https://doi.org/10.1111/j.1749-6632.1976.tb25467.x. URL `https://nyaspubs.onlinelibrary.wiley.com/doi/abs/10.1111/j.1749-6632.1976.tb25467.x`. _eprint: https://nyaspubs.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1749-6632.1976.tb25467.x.

Nazanin Firoozeh, Adeline Nazarenko, Fabrice Alizon, and Béatrice Daille. Keyword extraction: Issues and methods. *Natural Language Engineering*, 26(3):259–291, May 2020.

ISSN 1351-3249, 1469-8110. doi: 10.1017/S1351324919000457. URL https://www.cambridge.org/core/journals/natural-language-engineering/article/abs/keyword-extraction-issues-and-methods/84BFD5221E2CA86326E5430D03299711. Publisher: Cambridge University Press.

William H. Fletcher. *Concordancing the web: promise and problems, tools and techniques.* Brill Rodopi, January 2007. ISBN 978-94-012-0379-1. doi: 10.1163/9789401203791_004. URL https://brill.com/view/book/edcoll/9789401203791/B9789401203791-s004.xml. Pages: 25-45 Publication Title: Corpus Linguistics and the Web Section: Corpus Linguistics and the Web.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. AllenNLP: A Deep Semantic Natural Language Processing Platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, 2018.

Bilal Ghanem, Paolo Rosso, and Francisco Rangel. Stance Detection in Fake News A Combined Feature Representation. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5510. URL https://www.aclweb.org/anthology/W18-5510.

Genevieve Gorrell, Elena Kochkina, Maria Liakata, Ahmet Aker, Arkaitz Zubiaga, Kalina Bontcheva, and Leon Derczynski. SemEval-2019 Task 7: RumourEval, Determining Rumour Veracity and Support for Rumours. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 845–854, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/S19-2147. URL https://www.aclweb.org/anthology/S19-2147.

Nir Grinberg, Kenneth Joseph, Lisa Friedland, Briony Swire-Thompson, and David Lazer. Fake news on Twitter during the 2016 U.S. presidential election. *Science*, 363:374–378, January 2019. doi: 10.1126/science.aau2706.

Ashim Gupta, Giorgi Kvernadze, and Vivek Srikumar. BERT & Family Eat Word Salad: Experiments with Text Understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12946–12954, 2021. Issue: 14.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah 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, 2018.

Andreas Hanselowski, PVS Avinesh, Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M Meyer, and Iryna Gurevych. A Retrospective Analysis of the Fake

News Challenge Stance-Detection Task. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1859–1874, 2018a.

Andreas Hanselowski, Hao Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. UKP-Athene: Multi-Sentence Textual Entailment for Claim Verification. *EMNLP 2018*, page 103, 2018b.

Andreas Hanselowski, Christian Stab, Claudia Schulz, Zile Li, and Iryna Gurevych. A Richly Annotated Corpus for Different Tasks in Automated Fact-Checking. In *Proceedings of the 22nd Conference on Computational Natural Language Learning (CoNLL2019)*, 2019.

Momchil Hardalov, Arnav Arora, Preslav Nakov, and Isabelle Augenstein. A Survey on Stance Detection for Mis- and Disinformation Identification. *arXiv:2103.00242 [cs]*, February 2021. URL http://arxiv.org/abs/2103.00242. arXiv: 2103.00242.

Luheng He, Mike Lewis, and Luke Zettlemoyer. Question-Answer Driven Semantic Role Labeling: Using Natural Language to Annotate Natural Language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 643–653, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1076. URL https://www.aclweb.org/anthology/D15-1076.

Alfred Hermida. Twittering the news. *Journalism Practice*, 4:297–308, August 2010. doi: 10.1080/17512781003640703.

Christopher Hidey, Tuhin Chakrabarty, Tariq Alhindi, Siddharth Varia, Kriste Krstovski, Mona Diab, and Smaranda Muresan. DeSePtion: Dual Sequence Prediction and Adversarial Examples for Improved Fact-Checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8593–8606, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.761. URL https://www.aclweb.org/anthology/2020.acl-main.761.

Robin Jia and Percy Liang. Adversarial Examples for Evaluating Reading Comprehension Systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, 2017.

Yichen Jiang, Shikha Bordia, Zheng Zhong, Charles Dognin, Maneesh Singh, and Mohit Bansal. HoVer: A Dataset for Many-Hop Fact Extraction And Claim Verification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3441–3460, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.309. URL https://aclanthology.org/2020.findings-emnlp.309.

Dan Jurafsky. *Speech & language processing.* Pearson Education India, 2000.

Daniel Khashabi, Tushar Khot, Ashish Sabharwal, and Dan Roth. Question Answering as Global Reasoning Over Semantic Abstractions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), April 2018. ISSN 2374-3468. URL `https://ojs.aaai.org/index.php/AAAI/article/view/11574`. Number: 1.

Omar Khattab, Christopher Potts, and Matei Zaharia. Baleen: Robust Multi-Hop Reasoning at Scale via Condensed Retrieval. *arXiv:2101.00436 [cs]*, April 2021. URL `http://arxiv.org/abs/2101.00436`. arXiv: 2101.00436.

Youngwoo Kim and James Allan. FEVER Breaker's Run of Team NbAuzDrLqg. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)*, pages 99–104, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-6615. URL `https://www.aclweb.org/anthology/D19-6615`.

Pieter-Jan Kindermans, Kristof Schütt, Klaus-Robert Müller, and Sven Dähne. Investigating the influence of noise and distractors on the interpretation of neural networks. *arXiv e-prints*, 1611:arXiv:1611.07270, November 2016. URL `http://adsabs.harvard.edu/abs/2016arXiv161107270K`.

Karin Kipper, Hoa Trang Dang, and Martha Palmer. Class-Based Construction of a Verb Lexicon. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 691–696. AAAI Press, July 2000. ISBN 978-0-262-51112-4.

Nir Kshetri and Jeffrey Voas. The Economics of "Fake News". *IT Professional*, 19:8–12, November 2017. doi: 10.1109/MITP.2017.4241459.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692 [cs]*, July 2019. URL `http://arxiv.org/abs/1907.11692`. arXiv: 1907.11692.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=Bkg6RiCqY7`.

Yi-Ju Lu and Cheng-Te Li. GCAN: Graph-aware Co-Attention Networks for Explainable Fake News Detection on Social Media. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 505–514, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.48. URL `https://www.aclweb.org/anthology/2020.acl-main.48`.

Bill MacCartney and Christopher D. Manning. Natural logic for textual inference. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 193–200, 2007.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David 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, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-5010. URL `https://www.aclweb.org/anthology/P14-5010`.

Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. Open Language Learning for Information Extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/D12-1048`.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. SemEval-2016 Task 6: Detecting Stance in Tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/S16-1003. URL `https://www.aclweb.org/anthology/S16-1003`.

Pramod Kaushik Mudrakarta, Ankur Taly, Mukund Sundararajan, and Kedar Dhamdhere. Did the Model Understand the Question? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1896–1906, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1176. URL `https://www.aclweb.org/anthology/P18-1176`.

Preslav Nakov, Alberto Barrón-Cedeño, Tamer Elsayed, Reem Suwaileh, Lluís Màrquez, Wajdi Zaghouani, Pepa Atanasova, Spas Kyuchukov, and Giovanni Da San Martino. Overview of the CLEF-2018 CheckThat! Lab on Automatic Identification and Verification of Political Claims. In Patrice Bellot, Chiraz Trabelsi, Josiane Mothe, Fionn Murtagh, Jian Yun Nie, Laure Soulier, Eric SanJuan, Linda Cappellato, and Nicola Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, Lecture Notes in Computer Science, pages 372–387, Cham, 2018. Springer International Publishing. ISBN 978-3-319-98932-7. doi: 10.1007/978-3-319-98932-7_32.

Yixin Nie, Haonan Chen, and Mohit Bansal. Combining fact extraction and verification with neural semantic matching networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6859–6866, 2019. Issue: 01.

Rodrigo Nogueira and Kyunghyun Cho. Passage Re-ranking with BERT. *arXiv:1901.04085 [cs]*, April 2020. URL `http://arxiv.org/abs/1901.04085`. arXiv: 1901.04085.

Ray Oshikawa, Jing Qian, and William Yang Wang. A Survey on Natural Language Processing for Fake News Detection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6086–6093, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL `https://aclanthology.org/2020.lrec-1.747`.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106, March 2005. ISSN 0891-2017, 1530-9312. doi: 10.1162/0891201053630264. URL https://www.mitpressjournals.org/doi/abs/10.1162/0891201053630264.

Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob 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, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1244. URL https://aclanthology.org/D16-1244.

Javier Pastor-Galindo, Mattia Zago, Pantaleone Nespoli, Sergio López Bernal, Alberto Huertas Celdrán, Manuel Gil Pérez, José A Ruipérez-Valiente, Gregorio Martínez Pérez, and Félix Gómez Mármol. Spotting political social bots in Twitter: A use case of the 2019 Spanish general election. *IEEE Transactions on Network and Service Management*, 17(4):2156–2170, 2020. Publisher: IEEE.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin 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, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-2023. URL https://www.aclweb.org/anthology/S18-2023.

Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. DeClarE: Debunking Fake News and False Claims using Evidence-Aware Deep Learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 22–32, Brussels, Belgium, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1003. URL https://aclanthology.org/D18-1003.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy 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, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. URL https://aclanthology.org/D16-1264.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know What You Don't Know: Unanswerable Questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://aclanthology.org/P18-2124.

Lance Ramshaw and Mitch Marcus. Text Chunking using Transformation-Based Learning. In *Third Workshop on Very Large Corpora*, 1995. URL https://www.aclweb.org/anthology/W95-0107.

Francisco Rangel, Paolo Rosso, Martin Potthast, and Benno Stein. Overview of the 5th Author Profiling Task at PAN 2017: Gender and Language Variety Identification in Twitter. page 26, 2017.

Francisco Rangel, Paolo Rosso, Manuel Montes-y Gómez, Martin Potthast, and Benno Stein. Overview of the 6th Author Profiling Task at PAN 2018: Multimodal Gender Identification in Twitter. page 38, 2018.

Francisco Rangel, Anastasia Giachanou, Bilal Ghanem, and Paolo Rosso. Overview of the 8th Author Profiling Task at PAN 2020: Profiling Fake News Spreaders on Twitter. page 18, 2020.

Marco Tulio Ribeiro, Sameer Singh, and Carlos 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 `http://aclweb.org/anthology/P18-1079`.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer 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, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.442. URL `https://aclanthology.org/2020.acl-main.442`.

Benjamin Riedel, Isabelle Augenstein, Georgios P. Spithourakis, and Sebastian Riedel. A simple but tough-to-beat baseline for the Fake News Challenge stance detection task. *arXiv:1707.03264 [cs]*, May 2018. URL `http://arxiv.org/abs/1707.03264`. arXiv: 1707.03264.

Pum-Mo Ryu, Myung-Gil Jang, and Hyunki Kim. Open domain question answering using Wikipedia-based knowledge model. *Information Processing & Management*, 50:683–692, September 2014. doi: 10.1016/j.ipm.2014.04.007.

Tal Schuster, Darsh Shah, Yun Jie Serene Yeo, Daniel Roberto Filizzola Ortiz, Enrico Santus, and Regina 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, 2019.

Tal Schuster, Roei Schuster, Darsh J. Shah, and Regina Barzilay. The Limitations of Stylometry for Detecting Machine-Generated Fake News. *Computational Linguistics*, 46 (2):499–510, June 2020. doi: 10.1162/coli_a_00380. URL `https://www.aclweb.org/anthology/2020.cl-2.8`.

H. Shao, S. Yao, A. Jing, S. Liu, D. Liu, T. Wang, J. Li, C. Yang, R. Wang, and T. Abdelzaher. Misinformation Detection and Adversarial Attack Cost Analysis in Directional Social Networks. In *2020 29th International Conference on Computer Communications and Networks (ICCCN)*, pages 1–11, August 2020. doi: 10.1109/ICCCN49398.2020.9209609. ISSN: 2637-9430.

Dan Shen and Mirella Lapata. Using Semantic Roles to Improve Question Answering. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 12–21, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/D07-1002`.

Peng Shi and Jimmy Lin. Simple BERT Models for Relation Extraction and Semantic Role Labeling. *arXiv:1904.05255 [cs]*, April 2019. URL `http://arxiv.org/abs/1904.05255`. arXiv: 1904.05255.

Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huang Liu. dEFEND: Explainable Fake News Detection. *In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 395–405, July 2019.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *In Workshop at International Conference on Learning Representations*. Citeseer, 2014.

Amir Soleimani, Christof Monz, and Marcel Worring. *BERT for Evidence Retrieval and Claim Verification.* October 2019. URL `https://www.groundai.com/project/bert-for-evidence-retrieval-and-claim-verification/1`. GroundAI.

J Springenberg, Alexey Dosovitskiy, Thomas Brox, and M Riedmiller. Striving for Simplicity: The All Convolutional Net. In *ICLR (workshop track)*, 2015.

Gabriel Stanovsky, Ido Dagan, and Mausam. Open IE as an Intermediate Structure for Semantic Tasks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 303–308, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-2050. URL `https://www.aclweb.org/anthology/P15-2050`.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. Supervised Open Information Extraction. 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 885–895, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1081. URL `http://aclweb.org/anthology/N18-1081`.

Qi Su, Mingyu Wan, Xiaoqian Liu, and Chu-Ren Huang. Motivations, Methods and Metrics of Misinformation Detection: An NLP Perspective. *Natural Language Processing Research*, 1(1-2):1–13, June 2020. ISSN 2666-0512. doi: 10.2991/nlpr.d.200522.001. URL https://www.atlantis-press.com/journals/nlpr/125941255. Publisher: Atlantis Press.

James Thorne and Andreas Vlachos. Adversarial attacks against Fact Extraction and VERification. *arXiv:1903.05543 [cs]*, March 2019. URL http://arxiv.org/abs/1903.05543. arXiv: 1903.05543.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit 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, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1074. URL https://www.aclweb.org/anthology/N18-1074.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. The FEVER2.0 Shared Task. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)*, pages 1–6, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-6601. URL https://www.aclweb.org/anthology/D19-6601.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, \Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Jesse Vig. A Multiscale Visualization of Attention in the Transformer Model. In *arXiv:1906.05714 [cs]*, June 2019. URL http://arxiv.org/abs/1906.05714. arXiv: 1906.05714.

Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *Science (New York, N.Y.)*, 359(6380):1146–1151, March 2018. ISSN 1095-9203. doi: 10.1126/science.aap9559.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh 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, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.609. URL https://www.aclweb.org/anthology/2020.emnlp-main.609.

William Yang 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, 2017.

Matti Wiegmann, Benno Stein, and Martin Potthast. Overview of the Celebrity Profiling Task at PAN 2019. page 19, 2019.

Adina Williams, Nikita Nangia, and Samuel 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, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL https://www.aclweb.org/anthology/N18-1101.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

Andrew Yates, Rodrigo Nogueira, and Jimmy Lin. Pretrained Transformers for Text Ranking: BERT and Beyond. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 1154–1156, 2021.

Fabio Massimo Zanzotto, Andrea Santilli, Leonardo Ranaldi, Dario Onorati, Pierfrancesco Tommasino, and Francesca Fallucchi. KERMIT: Complementing Transformer Architectures with Encoders of Explicit Syntactic Interpretations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 256–267, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.18. URL https://www.aclweb.org/anthology/2020.emnlp-main.18.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. Semantics-Aware BERT for Language Understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9628–9635, April 2020a. ISSN 2374-3468. doi: 10.1609/aaai.v34i05.6510. URL https://ojs.aaai.org/index.php/AAAI/article/view/6510. Number: 05.

Zhuosheng Zhang, Yuwei Wu, Junru Zhou, Sufeng Duan, Hai Zhao, and Rui Wang. SG-Net: Syntax-guided machine reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9636–9643, 2020b. Issue: 05.

Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. Reasoning Over Semantic-Level Graph for Fact Checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6170–6180, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.549. URL https://www.aclweb.org/anthology/2020.acl-main.549.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong 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, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1085. URL `https://www.aclweb.org/anthology/P19-1085`.

Xinyi Zhou and Reza Zafarani. A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities. *ACM Computing Surveys*, 53(5):109:1–109:40, September 2020. ISSN 0360-0300. doi: 10.1145/3395046. URL `https://doi.org/10.1145/3395046`.

Xinyi Zhou, Atishay Jain, Vir V. Phoha, and Reza Zafarani. Fake News Early Detection: A Theory-driven Model. *Digital Threats: Research and Practice*, 1(2):12:1–12:25, June 2020. ISSN 2692-1626. doi: 10.1145/3377478. URL `https://doi.org/10.1145/3377478`.

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