UNIVERSITY OF COPENHAGEN

Ph.D. Thesis

Dustin Wright

Machine Understanding of Scientific Language

This thesis has been submitted to the PhD School of The Faculty
of Science, University of Copenhagen

Date: December 2022

Advisor: Isabelle Augenstein

Department	Department of Computer Science
Author(s):	Dustin Wright
Title and subtitle:	Machine Understanding of Scientific Language
Advisor:	Isabelle Augenstein
Date:	20 12 2022

Abstract

Scientific information expresses human understanding of nature. This knowledge is largely disseminated in different forms of text, including scientific papers, news articles, and discourse among people on social media. While important for accelerating our pursuit of knowledge, not all scientific text is faithful to the underlying science. As the volume of this text has burgeoned online in recent years, it has become a problem of societal importance to be able to identify the faithfulness of a given piece of scientific text automatically. This thesis is concerned with the cultivation of datasets, methods, and tools for machine understanding of scientific language, in order to analyze and understand science communication at scale. To arrive at this, I present several contributions in three areas of natural language processing and machine learning: automatic fact checking, learning with limited data, and scientific text processing. These contributions include new methods and resources for identifying check-worthy claims, adversarial claim generation, multi-source domain adaptation, learning from crowd-sourced labels, cite-worthiness detection, zero-shot scientific fact checking, detecting exaggerated scientific claims, and modeling degrees of information change in science communication. Critically, I demonstrate how the research outputs of this thesis are useful for effectively learning from limited amounts of scientific text in order to identify misinformative scientific statements and generate new insights into the science communication process.

Resume

Videnskabelig information udtrykker menneskets forståelse af naturen. Denne viden formidles i høj grad i forskellige former for tekst, herunder videnskabelige artikler, nyhedsartikler og diskurs blandt mennesker på sociale medier. Selvom det er vigtigt for at accelerere vores søgen efter viden, er ikke al videnskabelig tekst tro mod den underliggende videnskab. Som mængden af online tekst er vokset i de senere år, er det blevet en udfordring af samfundsmæssig betydning at kunne identificere troværdigheden af videnskabelig tekst automatisk. Dette speciale beskæftiger sig med skabelsen af datasæt, metoder og værktøjer til maskinforståelse af videnskabeligt sprog med henblik på at analysere og forstå videnskabelig kommunikation i større skala. For at nå frem til dette præsenterer jeg flere bidrag inden for tre områder af naturlig sprogbehandling og machine learning: automatisk faktatjek, læring med begrænset data og videnskabelig tekstbehandling. Disse bidrag omfatter nye metoder og ressourcer til at identificere checkværdige påstande, generering af modstridende krav, tilpasning af flere kilder til domæne, cite-worthiness detektion, nul-shot videnskabelig faktakontrol, opdagelse af overdrevne videnskabelige påstande og modellering af grader af informationsændringer i videnskabelig kommunikation . Slutteligt demonstrerer jeg, hvordan forskningsresultaterne fra denne afhandling er nyttige til effektivt at lære fra begrænsede mængder videnskabelig tekst for at identificere misinformative videnskabelige udsagn og generere ny indsigt i videnskabskommunikationsprocessen.

Acknowledgements

This work would not have been possible without the excellent mentorship of my advisor Isabelle Augenstein, who's insight, advice, and fine-tuned ability to foster quality scholarship enabled me to produce the work I did during my PhD. I would also like to acknowledge my collaborators who contributed immensely to the research described in this thesis. Additionally, I would like to thank my friends and colleagues who helped provide the balance needed to push through difficult periods and celebrate the good moments of my PhD. And finally I'd like to greatly thank my family, who though they are an ocean and a continent away, are always profoundly supportive of me.

Table of contents

1 Executive Summary

Scientific knowledge is ubiquitous online, where people have access to text describing scientific knowledge in all of its forms. These forms are heterogeneous, including scientific papers intended for an expert audience, technical reports intended for science enthusiasts, and news articles and social media posts intended for the general public. Science communication is the pipeline of translation through which scientific information is disseminated at these different levels: from higher complexity (i.e. scientific papers) to lower complexity (e.g. news and social media posts). At the same time, the current media climate is rife with misinformation (content which is false or inaccurate) and disinformation (content which is *intentionally* false or inaccurate). Mis- and dis-informative scientific content online is equally widespread due to the misrepresentation of scientific information through the science communication process [210, 132, 263, 180, 242, 51, 164, 224, 38, 32], which has downstream consequences on people's behavior [135, 77, 101] e.g. how vaccines are talked about in the media has an effect on vaccine uptake [135] and discussions around climate change alter perceptions of efficacy around methods to address it [101]. It is therefore a problem of societal importance to be able to combat the spread of scientific misinformation and improve science literacy among the public.

Towards this goal and due to the sheer volume of scientific text online, the automatic processing of scientific text using methods in natural language processing (NLP) and machine learning is an attractive option for assisting with organizing, understanding, and analyzing it at scale. New deep learning algorithms such as transformers [237], self-supervised learning techniques for text such as masked language modeling [62], large repositories of scientific text such as the Semantic Scholar Open Research Corpus (S2ORC) [149], and the availability of reliable implementations of state of the art models [251] have enabled widespread and trackable progress on a range of important tasks in scientific language understanding. At the same time, given the need to acquire labels in order to perform supervised training, many of the tasks for which models perform acceptably well are relatively low complexity (e.g. determining which words and phrases refer to a limited set of drugs and chemicals, determine the intent of a citation in a scientific paper, which section of a paper a sentence belongs to, etc.), and limited to small set of scientific fields. As such, there is a large gap in the field of scientific NLP addressing machine understanding of scientific language for combating scientific misinformation.

This thesis seeks to fill this gap via the cultivation of methods, tools, and resources for machine processing and understanding of scientific texts. In particular, I'm concerned with how machines can be used to *ensure information quality in science communication*; in other words, how to automatically detect and measure how accurate a piece of scientific information is. Prior to the work presented in this thesis, the area of information quality in science communication was sparsely studied in NLP, as most research in scientific NLP (including my own [259, 258, 19]) had focused on tasks related to extracting information from scientific text such as relationships between biomedical entities [165] and discourse strategies in scientific writing [48]. Here I will present my work on defining tasks and building new datasets and algorithms within the area of scientific NLP for ensuring information quality.

To arrive at this, I first contribute new resources and methods on several tasks which serve as the bridge towards machine understanding of scientific language. As the goal is to build tools for ensuring information quality, I first look at tasks in automated fact checking and create new methods for check-worthiness detection and adversarial claim generation. Next, as data availability is an issue with scientific text, I work on methods for learning with limited data and present new models for domain adaptation, learning from crowd-sourced labels, dataset generation, and prompt-based learning in both the general domain and science. Finally, I apply the knowledge gained from these studies of fact checking and learning with limited data to construct new tasks, datasets, and methods for automating the study of science communication. As such, it will help to provide some background on each of these areas.

1.1 Automated Fact Checking

Automated fact checking is a process which can generally be broken down into three steps:

1. Claim check-worthiness detection
2. Evidence retrieval
3. Veracity prediction

Additionally, a fourth step is added in the case of *explainable* fact checking, namely producing a justification [10]. Here I will provide an overview of each of these steps and datasets available for each.

Check-Worthiness Detection Claim check-worthiness detection is concerned with identifying claims which are worthy of being fact checked. Definitions can vary across datasets, but range from more subjective definitions such as claims for which there is a general public interest [102] to more objective definitions such as a claim which makes an assertion about the world that is checkable [127]. This is presented in a number of different ways, such as identifying statements in political debates which should be checked [23, 72] and identifying statements on Twitter which require verification [277, 276, 274]. The task is usually studied as a first step in fact checking separate from the rest of the process.

Evidence Retrieval Once check-worthy claims have been identified, one must then select existing evidence which can be used to determine the veracity of those claims. This is framed as a retrieval task from some trustworthy source, such as Wikipedia or a set of scientific documents.

Evidence documents are assumed to be factual, thus why they must be trustworthy. Additionally, evidence documents are necessary in order to assist in producing justifications for veracity predictions.

Veracity Prediction The next stage is to make a prediction of the veracity of the claim given the retrieved evidence. As such, a claim is determined to either be supported or refuted by the evidence, or that there is not enough evidence to predict either way. The veracity prediction task can also be performed in a late-fusion setup where evidence sentences are re-ranked during inference in order to improve performance [151, 215].

Veracity prediction and evidence retrieval are tightly bound, so fact checking datasets tend contain data for both. Popular early datasets for general domain fact checking include the Liar dataset [246], which contains real world-claims from Politifact, and the FEVER dataset [230], which is a large scale collection of manually written claims paired with evidence from Wikipedia. Fact checking datasets are evolving constantly, such as MultiFC which is largely multi-domain [13], FEVROUS which involves retrieving evidence and verifying claims over structured data [5], and multiple datasets for scientific fact checking such as SciFact [239], COVID-Fact [206], and CoVERT [163]. For a comprehensive overview of methods and datasets for automatic fact checking, see the survey from [93].

Justification Finally, an emerging last step in the fact checking pipeline is to produce a justification which explains the prediction. This is required in order to ensure that the prediction is trustworthy and convince the user of the correctness of the prediction. One of the earliest works trains a joint extractive summarization and veracity prediction model to produce justifications for the predictions [10]. This was later followed up by work on using post-editing to improve the fluency and coherence of these justification [118]. The survey in [93] contains a wide overview of methods and datasets for explainable fact checking and justification generation.

1.1.1 A New View of Fact Checking for Science

One of the contributions of this work is to rethink the fact checking pipeline in the context of scientific knowledge. The existing fact checking setup is designed to predict categorical truth and falsehood for a given claim. With scientific claims, I argue that veracity does not fully capture the types of subtle changes in information which are common in science journalism [224, 38, 77, 132, 263, 180, 242, 51, 164, 32]. As I will demonstrate in Chapter 8 and Chapter 9, we can reframe the problem with at least the following two steps:

1. Identify texts which discuss the same information

2. Measure how that information changes between those two texts

Step 1 is similar to the evidence retrieval component of automatic fact checking, and step 2 is similar to the veracity prediction stage, however they are generalized and decoupled from the notion of veracity. This opens the door to defining different types of information changes one is interested in measuring (e.g. one sentence exaggerating another). The other two stages of the fact checking pipeline, namely check-worthiness detection and justification, can also be included in this framework.

Check-worthiness can be included as is, but redefined towards predicting which scientific statements should be compared to the scientific literature. This is equally as subjective as the general domain version of the task, and is likely something that should be performed in tandem with humans in order to select the most salient scientific claims. The final step, justification, is a critical step in explainability which should be explored in future work. Namely, once one identifies statements describing the same information and measures how the statements differ, it is critical for a system to justify and explain exactly how those differences appear in text. In fact, an ideal system would be able to match scientific statements describing the same information, and simply explain in natural language how those statements differ, while at the same time binning statements into different classes of interest for the sake of triaging different types disinformation strategies e.g. exaggeration, cherry-picking data, changes in degree of certainty, etc. This would then enable new technologies where the average user browsing on the internet could have scientific misinformation automatically flagged for them, with an explanation of how the information they read online differs from what the source scientific literature says. Additionally, this could help improve science journalism by providing journalists with a tool to proof-read their articles and identify critical changes in their messaging which deviates from the scientific literature.

1.2 Learning with Limited Data

NLP has been seeing rapid progress in terms of the generalizability of models in recent years. Up until a few years ago, the dominant paradigm in NLP was to annotate large corpora for individual tasks and train a specialized model (generally an LSTM [106] or a transformer [237]) on that specific task or a set of highly related tasks. With the advent of ever-inflating transformers pre-trained in a self-supervised fashion with different flavors of language modeling [62, 192], models have become more general purpose and able to learn complex tasks with significantly less labeled data than previously. Here, I will focus on three areas which I make contributions to in this work for learning from limited labeled data: transfer learning, domain adaptation, and few-shot learning.

Transfer Learning Transfer learning is ubiquitous in the field of NLP currently. The major shift to transfer learning approaches in NLP started with the works of Elmo [182], ULM-FiT [112], and

BERT [62]. The basic idea is simple: starting with a suitably large network, train the weights of this network on a massive corpus of general purpose text in a self-supervised fashion. Then, fine-tune this model on tasks of interest using labeled data. The main self-supervised learning techniques used are auto-regressive language modeling, where a model is trained to predict the next token in a piece of text given all of the previous tokens, and masked language modeling (MLM), where, given a piece of text, mask out some percentage of tokens in that text and try to predict the missing tokens. The large pre-trained language models resulting from this type of training can be used to achieve state-of-the-art performance on a large array of tasks with less labeled data than previous methods [62, 148]. I make extensive use of large pre-trained language models in this thesis to help alleviate the need to employ massive amounts of labeled data for the scientific tasks I work on while still achieving reasonable results. Additionally, they form the foundation of the methods and models I develop throughout this thesis.

Domain Adaptation Domain adaptation is a form of transfer learning where the goal is to generalize to data from distributions lying outside that of the training data. Domain differences in text can occur in various ways, from differences in the subject or topic of different texts to texts that are in completely different languages. Approaches generally fall into three categories: *supervised* approaches (e.g. [56, 76, 134]), where both labels for the source and the target domain are available; *semi-supervised* approaches (e.g. [66, 261]), where labels for the source and a small set of labels for the target domain are provided; and lastly *unsupervised* approaches (e.g. [31, 80, 225, 146]), where only labels for the source domain are given. Additionally, different approaches exist in the single-source setting, where data in only one source domain is available, versus the multi-source setting, where data in multiple source domains is given [271]. Common methods in NLP for domain adaptation include domain adversarial training [80], pivot-based methods [273], progressive language model fine-tuning [94], and mixture-of-experts [92].

For scientific text, domain adaptation is especially difficult given the stark differences in the language used between scientific domains. The definition of a domain is also particularly tricky – even within the same academic field, say medicine, researchers working on different topics employ vastly different language, oftentimes even for the same concepts [33]. Additionally, researchers in different fields have vastly different needs. For example, a common task in biomedical NLP is relation extraction for the construction of knowledge bases [248], but each sub-discipline has its own particular set of entities and relations that they care about [19]. At the same time, annotating scientific text is highly expensive and time-consuming given the need for domain expertise and the technical nature of science. As such, it is highly beneficial to be able to generalize across domains. Part of this thesis will focus on domain adaptation, in particular multi-source domain adaptation with large pre-trained transformers and building structural scaffolding datasets with scientific text to improve transfer learning to new tasks.

These approaches are promising for two reasons. First, labeled data for scientific text tasks tend to be available for a few popular research areas, including medicine, biology, and computer science. It makes sense to investigate multi-source domain adaptation in order to leverage all available data to generalize to sparsely annotated scientific fields. Second, structural scaffolding tasks are generally much easier to acquire data for as they are acquired automatically. Scaffold tasks are weakly-supervised tasks where data is acquired via the structure of scientific documents and which help improve the generalization of models on tasks involving scientific texts [48]. Examples of scaffolding tasks include predicting the section of a scientific document where a piece of text resides and predicting whether or not a sentence should have a citation. As the training label is embedded in the structure of the document, massive corpora can be built cheaply and lead to gains in performance across tasks, as we will show in Chapter 6.

Learning From Crowd-Sourced Labels One method commonly used to acquire a large amount of annotations for a problem relatively cheaply is to employ crowd annotators on platforms such as Amazon Mechanical Turk to provide such annotations. While this is useful on extremely well-defined tasks, it is very difficult to acquire many high quality labels from crowd-workers for science. As such, one generally makes a tradeoff between annotation abundance and cost in order to acquire high quality labels from experts in science [69, 65, 248].

In order to gain the most from these crowd annotations, recent work has looked into how to learn from them directly without selecting one single ground-truth label for a given sample [235]. The work of [183] was one of the first, which demonstrated that learning directly from crowd annotations treated as soft-labels using the softmax function leads to better out of distribution performance in computer vision. This line of work has been followed by [234] and [78] in NLP, looking at the use of the KL divergence as an effective loss on the soft labels. The survey of [235] provides an extensive set of experiments on different methods for learning from crowd labels on a vast array of datasets. This is a potentially attractive option for scientific text in order to improve generalization from a limited set of crowd-annotations, which are expensive to acquire. In Chapter 5, I develop new methods for learning from crowd annotations, and demonstrate their efficacy on a number of tasks and datasets for out-of-domain performance, including with scientific text.

Prompt-Based Learning In some cases, labeled data is either completely unavailable or it is prohibitively expensive to acquire a large enough set of labeled data to effectively train a model directly. Few-shot learning is an area of study in machine learning where the goal is to develop methods which generalize well from a highly limited set of labeled data. The area of few-shot learning investigated in this thesis for scientific text is prompt-based learning.

Prompt-based learning seeks to leverage language model pre-training for generalization. As such, the two widely used flavors of prompt-based learning are prefix prompting and cloze prompting [147], reflecting the two major forms of language model pretraining used today. The core idea behind both methods for classification is the same: instead of solving a classification problem $P(y|x)$, where x is the input and y is one of a discrete set of labels, solve the problem $P(V(y)|p(x))$, where $p(x)$ is a function which transforms the input x by inserting one or more tokens into x, with at least one token being a masked out token, and $V(y)$ maps the label y to one or more tokens in the language model's vocabulary. Then, instead of learning a completely new classifier over a new label space, one can use the classifier which was trained over the model's vocabulary during language model pretraining and transform the classification task into a language modeling task, where $P(V(y)|p(x))$ is a prediction over the tokens $V(y)$ in the masked positions of $p(x)$. In this, the model has presumably learned useful patterns from the much larger pretraining corpus which can be transferred to the downstream classification task, provided the functions $p(x)$ and $V(y)$ are suitably reflective of the classification problem, particularly with respect to the language model being used.

As a concrete example, consider the problem of sentiment analysis of movie reviews. Given input $x =$ "I was on the edge of my seat!" and label $y =$ "[POSITIVE]", $y \in \{$ [POSITIVE] , [NEGATIVE] $\}$, we can define $p(x)$ to be a function which takes input x and produces output "[x]. The movie is [MASK]" and $V(y)$ to be a function which takes our discrete label y and selects a token $V(y) \in \{$"good", "bad"$\}$ which we will train the model to predict in the position of the "[MASK]" token. In this case, we would hope that the model would predict "good" in the position of the "[MASK]", and that language model pretraining would provide a suitable initialization such that this can be fine-tuned with much fewer examples than training a classifier from scratch over the original input (in a large enough pretraining corpus, the language model has potentially seen similar examples associating "being on the edge of one's seat" as "good" in the context of film or other media). This style of prompt-based learning was popularized in the works of [213, 212]. Recent directions for prompt-based learning include automated prompt searching [217], automated verbalizer searching [212], learning soft prompts [155], and learning from multiple prompts [116].

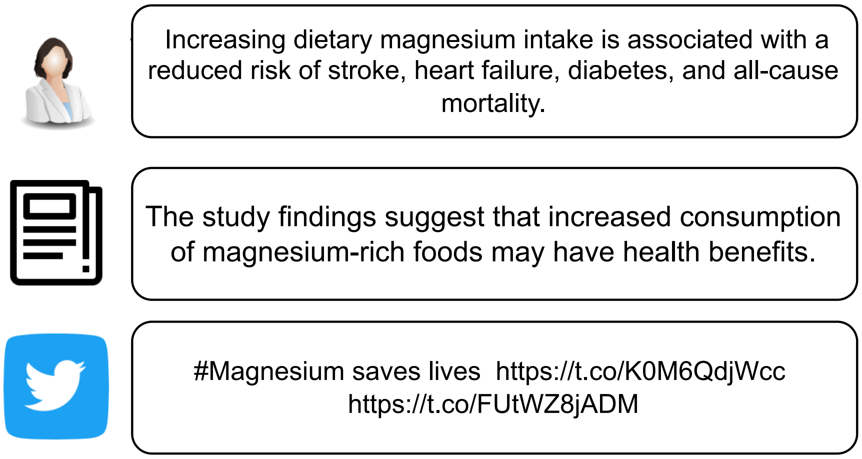
Prompt-based learning presents a promising direction for scientific text understanding, as it both alleviates the need for labeling large corpora and allows for the injection of expert knowledge into the classification problem. One of the core challenges is determining useful prompts and verbalizers for a problem; models can be highly sensitive to the selection of patterns and label verbalizations [212]. The choice of model can also vastly change the optimal prompts for a given problem. As such, there is a tradeoff between the expense of annotating corpora for scientific text tasks and the time needed to train an appropriate language model and engineer or learn prompts which would reduce this expense while still providing acceptable performance. I will demonstrate in Chapter 8 that this is possible on the scientific text task of predicting exaggeration in science communication with as few as 200 training samples. However, I would argue that more work is needed in the direction of prompting for scientific text, particularly for determining which corpora are suitable for pretraining on which tasks, and how domain knowledge can be appropriately applied for a given task.

1.3 NLP for Science

While significant progress has been made in NLP in recent years, much of the progress is restricted to a narrow domain of general text which does not require deep understanding of complex topics or jargon typical to much of science. Scientific text processing requires specialized datasets and resources for performing the various tasks one would wish to perform on scientific text. Additionally, specific scientific fields often require individualized resources, as each field has its own idiosyncrasies and jargon which aren't represented in or representative of other fields [33].

Much work on scientific NLP focuses on tasks relevant to the research community but not necessarily the general public. These tasks include named entity recognition (NER) and entity linking of biomedical concepts such as diseases and chemicals [65, 259], as well as relation extraction to extract associations between these concepts [248]. Tools to perform these tasks are indeed useful: the construction of knowledge bases using automated tools can help researchers quickly organize a field of literature, offsetting the cognitive load required to perform their research and assist in the discovery of e.g. drug treatments for a novel disease. Similarly, a popular NLP task for scientific text is the summarization of scientific papers [262]. The goal is to produce a concise summary of a paper which is easily consumable by a researcher in that field. The task is learned by training a generative model e.g. BART [138] to produce the abstract of a paper given the paper full text. More recent work has sought to build datasets and models for generating meta-summaries which ingest multiple documents on a particular topic and produce a summary of the major findings on that topic [63]. Finally, a variety of classification tasks over scientific texts exist, including citation intent classification [119, 48] and paper field prediction [26]. However, in this work I am concerned with machine understanding of scientific language more broadly than solely academic papers.

For the interface between scientific literature and lay text, recent work has begun to investigate tasks such as lay summarization [45] and scientific fact checking [239, 206, 163]. The goal of lay summarization is to simplify complicated scientific literature and generate summaries which a lay person can understand. This is both a difficult linguistic task and also important for making the findings of science accessible to the general public. Another task at the interface of papers and the public, scientific fact checking is focused on the veracity of scientific information. The problem has been studied both on synthetic data derived from scientific abstracts [239] and on real-world claims sourced from various social media websites [206, 163]. The task is difficult, as it requires models for both information retrieval and entailment prediction, the difficulty if which is exacerbated by the complexity of scientific language.



Increasing dietary magnesium intake is associated with a reduced risk of stroke, heart failure, diabetes, and all-cause mortality.

The study findings suggest that increased consumption of magnesium-rich foods may have health benefits.

#Magnesium saves lives https://t.co/K0M6QdjWcc https://t.co/FUtWZ8jADM

Figure 1.1: We are interested in measuring the information similarity of statements about scientific findings between different sources, including scientific papers, news, and tweets, shown here with real examples. The finding in this figure comes from [73] and the news quote is from [195].

This thesis addresses three gaps in the literature around scientific text understanding. The first is that information quality in science goes beyond veracity. While categorical falsehoods do exist in science communication, and it is important both as a semantics problem as well as for public well being, categorical falsehood only addresses one type of scientific misinformation. In practice, more subtle distortions such as exaggeration and hedging permeate science communication, and this has an impact on people's behavior [224, 38, 253, 252, 95, 77, 135]. Even well intentioned science communicators are prone to these distortions, as the quotes in Figure 1.1 demonstrate. Here, I present a new paradigm with which to view the scientific misinformation problem in NLP: as one where we wish to identify the degree to which scientific statements are different and what those differences are.

The second gap I address is that the majority of works on scientific language understanding, particularly for ensuring information quality, ignore most of the text of scientific papers. Most works will use paper abstracts as sources of claims and evidence to perform fact-checks against [239, 206, 163]. While the abstract provides presumably the most salient information in a scientific article, tasks which focus solely on paper abstracts will fail to capture the more subtle pieces of information in an article such as caveats to findings and limitations, a well documented phenomenon for journalists as well [77]. In Chapter 9 I will obviate the need to ingest the full texts of papers for fully understanding scientific language and the science communication pipeline.

The third gap I address in this thesis is that the majority of existing work around scientific information quality is narrow in scope. Popular datasets for scientific fact checking and summarization, including [239], focus largely on biology and medicine. Additionally, they restrict themselves to just scientific papers or just scientific papers and one other domain (e.g. Reddit [206], Twitter [163], etc.). I present work in this thesis (particularly, Chapter 9) which broadly

Sentence 1	**Sentence 2**
The polar bear is sliding on the snow.	A polar bear is sliding across the snow.
A plane is taking off	An air plane is taking off
A dog rides a skateboard	A dog is riding a skateboard
A man is playing the drums	A man plays the drum

Table 1.1: Samples of sentence pairs in STSB which have a similarity score of 5

covers three stages of the science communication pipeline (papers, news, and Twitter) and four scientific fields (medicine, biology, computer science, and psychology). Additionally, I argue that this is necessary in the context of ensuring information quality in science and machine understanding of scientific language in general, as it is important to build tools which are robust across fields of research and level of complexity for the sake of increased public good.

1.4 Machine Understanding of Scientific Language

The core problem explored in this thesis is how to enable machine understanding of scientific language. In particular, I'm concerned with how machines can understand what information is expressed in a scientific sentence, and how to determine the degree to which two scientific sentences express the same information. As a preliminary, it is important to explicitly define what I mean by *scientific information*. Scientific information is expressed through scientific *findings*, where a scientific finding has the following definition:

Definition 1.1 *A scientific finding is a statement that describes a particular research output of a scientific study, which could be a result, conclusion, product, etc.*

This general definition holds across fields; for example, many findings from medicine and psychology report on effects on some dependent variable via manipulation of an independent variable, while in computer science many findings are related to new systems, algorithms, or methods. The goal is to be able to build systems which can help analyze and improve science communication through automatic processing of scientific findings, a critically important topic which has large impact on both people's behavior and public policy [167, 135].

There are several core challenges on the path to this goal. First, what tasks are necessary in order to achieve it? As discussed in Section 1.3, there are several existing tasks in scientific NLP. Information extraction can potentially help, but the information measured in these tasks are explicit entities and relations, which would potentially require defining and building labeled datasets for each type of relation which expresses a scientific finding we would be interested in. The task of automatic fact checking [230] (and indeed the scientific version of it [239]) is

Sentence 1	**Sentence 2**
Higher-income professionals had less tolerance for smartphone use in business meetings.	We are intrigued by the result that professionals with higher incomes are less accepting of mobile phone use in meetings.
If we allow people to retract recently posted comments, then we may be able to minimize regret from posting in the heat of the moment.	Allowing users to retract recently posted comments may help minimize regret .
Papers with shorter titles get more citations #science #metascience #sciencemetrics	Our analysis suggests that papers with shorter titles do receive greater numbers of citations.
Low levels of self-esteem and poor emotional processing skills were significantly correlated with gang involvement, as were low levels of parental monitoring, poor parental communication and housing instability.	Major findings also indicated that low levels of parental monitoring, poor parental communication and housing instability were significantly associated with gang involvement.

Table 1.2: Samples of sentence pairs in SPICED (Chapter 9) which have a matching score of 5.

perhaps a better place to start, though it is concerned with a very specific type of information change: veracity. While it is important to be able to measure when one scientific statement is contradicted or supported by another, in practice the types of information changing between different utterances of the same finding tend to be more nuanced and not necessarily categorical falsehood [224, 38, 253, 252]. It would seem that a more broad notion of information change is needed in order to achieve the goal of this work.

A promising line of work to emulate is semantic textual similarity (STS) [44, 81]. Here, the goal is to measure how similar are the meanings of two pieces of text, measured as a scalar from 1-5. Some examples of what would be considered highly similar sentences in a typical STS task (from STSB [44]) are given in Table 1.1. Here we see a very strict notion of similarity: for a pair to be highly similar, the entire meaning of the sentence must be preserved from one sentence to the other. While closer to the type of information change concerned with in this work, this definition of similarity is too restrictive to be useful in the context of scientific information. As described in Definition 1.1, a finding is an expression of a research output. In this, the salient information relates to what is said about the research output, so some information in a piece of text may change the semantics of the text but not what is meant by the finding. Consider the following sentences:

> **Sentence 1**: The study showed that increased dietary sugar led to weight gain in humans.
>
> **Sentence 2**: "If a person eats more sugar, they'll gain more weight," said the researchers.

The meaning of these two sentences is slightly different, but the information in the findings is equivalent. This is further demonstrated in real examples from the dataset I present in Chapter 9, shown in Table 1.2. Given this, STS is a good starting point if we can modulate the task to focus solely on the information in the scientific findings.

This thesis will build up to and ultimately define the task of measuring scientific information change, as well as develop and evaluate different ways of modeling and learning it. Prior to this work, this framing of the problem of scientific misinformation was not defined, and the most related work came in the forms of automatic fact checking [230] and causal claim strength prediction of scientific statements [265, 266, 143]. Because of this, I will ask several questions in this thesis: what tasks are relevant for understanding scientific language and how do we define them? How do we collect data for these tasks? How do we evaluate them? How do we model and learn them? I tackle these questions systematically in the following way, initially using existing datasets and tasks and eventually building new methods and datasets to achieve the goal of scientific language understanding for ensuring information quality:

General Domain Fact Checking I first present novel solutions to problems in general domain fact checking (Chapter 2 and 3). This includes predicting when statements should be fact checked, as well as generating adversarial inputs for fact checking models in order to evaluate their robustness.

Modeling and Dataset Creation for Scientific Text Tasks I next investigate several problems related to dataset creation and modeling for scientific tasks. This is predicated on the fact that dataset creation with scientific text is both expensive and time consuming [239, 224, 38], and as such, datasets for scientific tasks tend to be small and/or difficult to acquire. Additionally, one generally must acquire data for each scientific field and target task of interest [65, 259, 239, 206, 163, 248, 130, 262, 63, 45]. To alleviate some of these problems, with an eye to building up new datasets for the tasks required for scientific language understanding and measuring information change, I present contributions to domain adaptation (Chapter 4), learning from noisy crowd-sourced labels (Chapter 5), automatic dataset generation (Chapter 7), and few-shot learning (Chapter 8).

Towards Scientific Language Understanding Finally, I define, model, evaluate, and analyze several tasks in scientific language understanding, in particular with respect to measuring information change. For this I look into existing tasks such as cite-worthiness detection (Chapter 6) and scientific fact checking (Chapter 7), curate better evaluation data and develop models for the task of detecting exaggerated scientific statements (Chapter 8), and define and build a comprehensive dataset for the new task of measuring information change in science communication (Chapter 9). In addition, I demonstrate how the dataset and models built in Chapter 9 can be used to help both with other tasks in scientific language understanding and with analyzing science communication broadly.

In the following sections, I will break down and summarize each of these components and how they contribute to the goal of machine understanding of scientific language.

1.4.1 General Domain Fact Checking

As a part of automatically ensuring information quality in science, I develop new methods for ensuring information quality in general domain texts. For this, I work on two important issues in fact checking: detecting when a claim should be fact checked (Chapter 2) and fact checking model robustness against adversarial attacks (Chapter 3).

As the first step in automatic fact checking, check-worthiness detection involves determining if a statement “makes an assertion about the world that is checkable” [127]. This step is useful both for further processing by machine learning models to determine veracity as well as for notifying fact checkers of information worthy of a fact check. In this work, I use the observation that this is a highly subjective task [127] to hypothesize that while samples labeled as positive are likely true positives, not all negative samples are true negatives. As such, I experiment with positive unlabeled (PU) learning for the check-worthiness detection task on three datasets: Wikipedia citation needed detection, rumor detection on Twitter, and political speech check-worthiness detection. I find that while PU learning is helpful for Wikipedia and Twitter, it is detrimental to performance in the political domain, noting some inconsistencies in the labeling of that dataset. This work and the observations made become some of the basis for the work I perform on check-worthiness in science in the form of cite-worthiness detection: the task of identifying scientific sentences which require a citation.

The second general domain fact checking task I investigate is adversarial claim generation. Adversarial claims are deceptive model inputs designed to mislead an ML system into making the wrong prediction. Its important to reveal such system vulnerabilities in order to correct them, especially for fact checking systems where an adversarial claim can trick a model into predicting a false claim is true. In this work I explore universal adversarial triggers – single tokens which can be prepended to a wide range of inputs to force a particular model prediction to be changed in a certain direction (e.g. “SUPPORTS” to “REFUTES” in a fact checking model) [82]. The primary issue with these types of attacks is that they tend to truly flip the input label (e.g. prepending a negation word such as “None” in order to flip a supported claim to a refuted claim) and make the input nonsensical. To address this, I introduce a secondary objective in the adversarial trigger search which optimizes the semantic textual similarity of the original claim and adversarial claim. To ensure the coherence of the adversarial claim, I additionally introduce a generation component using GPT-2 [192] which is trained to include the trigger token in the output claim. Combining these two modules for adversarial claim generation results in more robust adversarial claims which are coherent and don’t trivially flip the original label of the claim.

1.4.2 Learning from Limited Data

The next part of this thesis presents several contributions in the area of building and utilizing datasets for scientific language understanding tasks in the presence of limited data. The methods presented are general and applicable to a wide range of machine learning and NLP tasks, so I evaluate them on both general domain and scientific text. The methods I build are in the following areas of machine learning:

- Domain adaptation
- Learning from crowd-sourced data
- Generation
- Few-shot learning

Domain adaptation In Chapter 4 I present work on using large pre-trained transformer models to perform multi-source domain adaptation (MSDA). The main idea behind MSDA is to leverage data for a particular task but drawn from disparate modes of the underlying distribution in order to perform inference on a target mode of data for which no training labels are available (see Figure 1.2). Examples of these different modes are different types of products in the case of reviews on Amazon or different fields of study in the case of scientific text. Domain adaptation is relevant to scientific language understanding due to the cost of obtaining high quality human annotated data in science. If we can make better use of less data and already existing data, we will have made progress towards improving NLP for science.

The particular methods I explore in this work are mixture-of-experts techniques and domain adversarial training [92, 80]. The idea behind mixture-of-experts is to train individual models on particular domains and subsequently learn how to mix their predictions for the target domain. This is based on the hypothesis that some domains are more relevant than others for the target e.g. the language used in medicine is more similar to the language used in biology than in computer science, therefore a model trained on biology texts will be more relevant than one

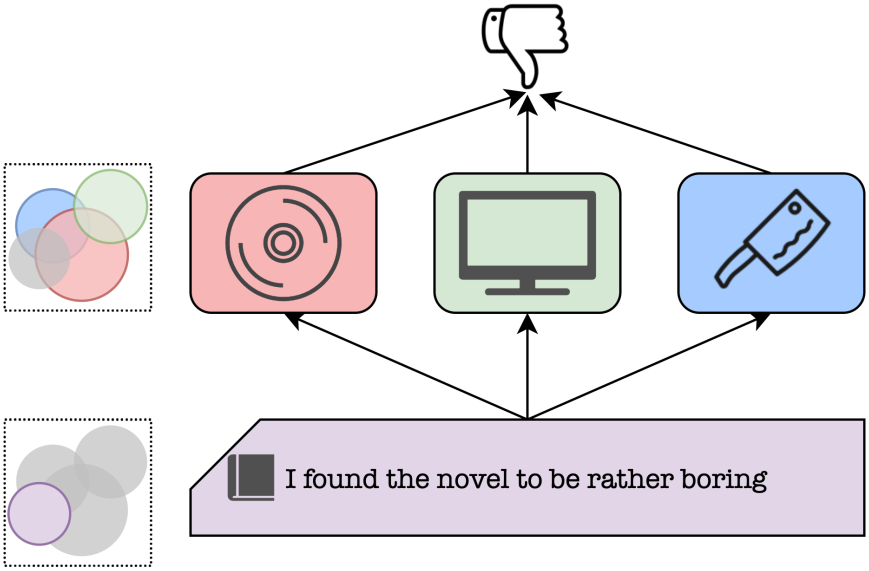


Figure 1.2: In multi-source domain adaptation, a model is trained on data drawn from multiple parts of the underlying distribution. At test time, the model must make predictions on data from a potentially non-overlapping part of the distribution.

trained on computer science texts. Domain adversarial training on the other hand aims to induce a more uniform internal representation of data across domains such that the representations of data in the target domain are similar to the representations of data in the source domains. The net effect of this is that the classifier trained on source domain data generalizes better to target domain data, as the target domain data appears to lie within the distribution of data that the model was trained on.

I examine how mixture-of-experts and domain adversarial training can be effectively utilized with the current dominant large pretrained transformer models in NLP. I do so with several different types of mixing strategies, from simple ensembling to a learned attention mechanism, as well as including or excluding domain adversarial training. I find in this work that while simple ensembling provides some gains in performance across tasks, more complex mixing strategies provide no gain in performance. An analysis of the predictions of each individual domain expert reveals that these large transformer models learn highly homogeneous classifiers for a particular task despite being trained on *completely different* data, helping to explain the result that complex mixing functions provide no gain in performance. Additionally, I find that while domain adversarial training does indeed induce a more uniform representation in a given model, this does not translate into improved generalization to target data.

Learning from Crowd-Sourced Data In Chapter 5, I propose new methods for learning from crowd annotations treated as soft-targets that confer more robust performance in the out-of-domain setting across a number of tasks. This is based on the fact that selecting an effective training signal for tasks in natural language processing is difficult: collecting expert annotations is expensive, and crowd-sourced annotations may not be reliable. Recent work in machine learning has demonstrated that learning from soft-labels acquired from crowd annotations can be effective [234, 78, 235, 183] especially when there is distribution shift in the test set [183]. However, the best method for acquiring these soft labels is inconsistent across tasks.

To address this, I propose new methods for acquiring soft-labels from crowd-annotations by aggregating the distributions produced by existing methods. In particular, I propose to find a distribution over classes by learning from multiple-views of crowd annotations via temperature scaling and finding the Jensen-Shannon centroid of their distributions. I demonstrate that using these aggregation methods leads to best or near-best performance across four NLP tasks on out-of-domain test sets, mitigating fluctuations in performance when using the constituent methods on their own. Additionally, these methods result in best or near-best uncertainty estimation across tasks. I argue that aggregating different views of crowd-annotations as soft-labels is an effective way to ensure performance which is as good or better than the best individual view, which is useful given the inconsistency in performance of the individual methods.

Generation I next propose novel methods for dataset generation in the context of scientific fact checking in Chapter 7. Again due to the cost of annotation for scientific text tasks, one attractive option is to leverage existing data in order to automatically create new data with which to train models. One of the primary datasets for scientific fact checking, namely SciFact [239], is one such dataset in which human experts were required to manually write claims and pair them with ground truth statements from scientific abstracts which either support or refute those claims. In this work, I explore how the existing data in SciFact can be used to automatically generate new training data both in an unsupervised and supervised fashion. I find that both methods are effective, as training data generated using both can be used to train a model to within 90% of the performance of a model trained on manually written claims on the veracity prediction task of scientific fact checking.

Few-shot Learning Next, I develop methods for few-shot learning evaluated on the nascent task of scientific exaggeration detection (Chapter 8). Few-shot learning is an area of study which aims to achieve as much generalization as possible from as little data as possible. The particular area of few-shot learning explored in this work is prompt-based learning with large pretrained language models. For this we develop multi-task pattern exploiting training (MT-PET), a multi-task version of pattern exploiting training (PET) [213, 214].

As discussed in Section 1.2, the core idea behind PET is to transform a classic supervised learning task, in which the goal is to learn a classifier which produces a probability distribution over K classes for a given input, to a cloze-style question answering problem which can make effective use of masked langauge model (MLM) pretraining. In this, one engineers a “prompt” for

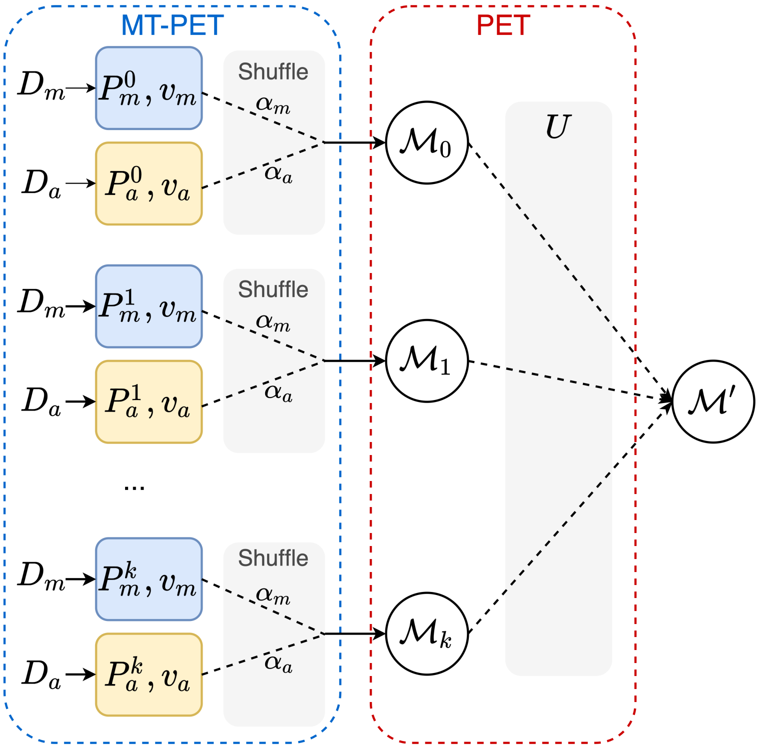


Figure 1.3: MT-PET design. We define pairs of complementary pattern-verbalizer pairs for a main task and auxiliary task. These PVPs are then used to train PET on data from both tasks.

their input data with one or more tokens in this prompt masked, and the classification task is to predict the appropriate token in the language model's vocabulary which would fill the mask token in the prompt. These tokens are explicit verbalizations (a.k.a *verbalizers*) of the classes which the model should be trained to predict.

With PET, one defines prompts and verbalizers for the single task one wishes to solve. In Chapter 8 I develop **MT-PET** (see Figure 1.3), a multi-task version of this which can leverage prompts and verbalizers from complementary tasks to the main task one wishes to perform. The hypothesis is that some transfer learning may occur between similar tasks (e.g. semantic textual similarity and natural language inference), and thus having complementary patterns and verbalizers when training PET and using all of the training data from both tasks should help with few-shot learning. Indeed I find that MT-PET does help for the task of scientific exaggeration detection when using the complementary tasks of detecting exaggerated statements and detecting the causal claim strength of a statement with as few as 200 samples from each task.

1.4.3 Tasks in Scientific Language Understanding

Finally, I contribute to a number of tasks in scientific language understanding, culminating in a new task and dataset on measuring information change in science across different media. The tasks explored in this thesis are the following:

- Cite-worthiness detection
- Scientific fact checking
- Scientific exaggeration detection
- Modeling information change in science

Cite-worthiness detection I first present new data and models for cite-worthiness detection (Chapter 6). The task of cite-worthiness detection is: given a statement from a scientific paper, predict if that statement should have a citation i.e. that it requires external evidence in order to be validated. This task is similar to the check-worthiness detection tasks examined in Chapter 2. Additionally, as a structural scaffold it is easy to acquire large amounts of data for this task.

For this work, I observe that existing datasets for cite-worthiness are limited in size, limited in the number of domains studied, have high class imbalance, and are low-quality in terms of dataset cleanliness [223, 29, 75, 74]. In response to this, I develop CiteWorth, a large, rigorously curated, and high quality dataset for cite-worthiness detection across 10 scientific domains. CiteWorth contains over 1.1M sentences, of which 300K are cite-worthy and 800K are non-cite-worthy. Additionally, I develop a strict set of rules for curating and cleaning cite-worthy sentences such that the vast majority of trivial markers of possible citations are removed. The dataset is also *contextualized* – data is collected at the paragraph level, such that all surrounding context within a paragraph is available for each sentence.

I perform several baseline experiments on CiteWorth, finding that including context sentences can improve cite-worthiness detection by 5 points in F1 score. I additionally perform a domain analysis to show that CiteWorth is useful in the study of domain adaptation for scientific text as there exists strong differences in representations and cross-domain performance for different fields. Finally, I show that pre-training on the cite-worthiness detection task provides gains on several downstream tasks in scientific text understanding tasks, providing further evidence for the usefulness of scaffolding tasks in scientific NLP [48].

Scientific fact checking The next task I look into is scientific fact checking (Chapter 7). Scientific fact checking consists of the following: given a scientific claim c and a corpus of scientific abstracts D, retrieve evidence abstracts from D and predict if c is either *supported* or *refuted* by those documents, or if there is *not enough information* to make a prediction. Several datasets have been introduced for this task (e.g. [239, 206, 163]). I focus here on the SciFact dataset [239], which is manually created.

The fact checking task is relevant in the context of combating scientific misinformation, but data is difficult to acquire. The SciFact dataset is built by having domain experts manually write scientific claims based off of findings described in scientific abstracts. These claims are then paired with source abstracts and sentences which support the claim. Negative instances are also created manually, where annotators manually rewrite claims to be contradicted by the source abstract. This is an expensive and time-consuming process, resulting in a somewhat small dataset ($\sim$1,400 claims). Given this, I build and test new methods for scientific fact checking dataset generation, achieving competetive performance on veracity prediction with no manually labeled samples.

Scientific fact checking addresses veracity, which is an important type of information change to model. As such, datasets such as SciFact are good starting points for building tools to combat scientific misinformation, but they are limited in scope both in terms of covering the types of misinformation that appear in science and in covering scientific language beyond academic papers. The next works I present are attempts to go beyond veracity and beyond solely scientific literature, to propose a different paradigm with which to think about and examine the problem of automating the process of ensuring information quality in science.

Scientific exaggeration detection As a first step, I present work on scientific exaggeration detection in Chapter 8. Similar to scientific fact checking, one of the goals of performing exaggeration detection is to combat scientific misinformation online and flag particular types of information change between statements made in source scientific literature and popular media. Exaggeration is one of the well documented issues in science communication [224, 38].

While an important issue, little data was available for training machine learning models on this task prior to our study, and the problem had mostly been studied in NLP as one of detecting the causal strength of scientific claims as opposed to directly measuring differences in this claim strength [265, 266, 143]. One of the goals of this work was to present a study which used real world data one would find in the wild, as opposed to artificially created data. Therefore, as a first step, I curate existing data from various studies on exaggeration in science communication into a comprehensive test set and small training set, necessary for measuring model performance and progress on the task.

The dataset comes from the studies in [224] and [38], where domain experts manually label the primary findings as described in scientific papers and press releases along with their causal claim strength. Overall I curate 100 pairs of findings from papers and press releases for training and 553 pairs for evaluation. As the training dataset is small, I develop methods for prompt-based learning for this task, and demonstrate that one can achieve moderate levels of performance with only the 100 training instances in the data.

Modeling information change Finally, I address the problem of modeling general information change in scientific findings between different media. This task is inspired by the fact that no comprehensive dataset had existed for the basic task of pairing together sentences which describe the same scientific findings. This is a necessary step if one wishes to make comparisons between how scientists and the media describe scientific findings, in order to analyze this communication and provide indications of where such communications fail.

To address this gap, I build a dataset of paired scientific findings labeled with the degree to which the two findings describe **the same** findings using a 5-point scale which I call the Information Matching Score (IMS). Some examples of paired findings with a matching score of 5 are given in Table 1.2. The dataset, namely the SCIENTIFIC PARAPHRASE AND INFORMATION CHANGE DATASET (SPICED), is built by first pairing together potential scientific findings as presented in scientific papers, news media, and Twitter using SentenceBERT (SBERT) [197], then presenting the potential pairs to human annotators. The Prolific platform[1] is used in order to hire domain experts in the scientific fields represented in the data: medicine, biology, psychology, and computer science. After constructing the dataset and ensuring the data is high quality, I train several baseline models and benchmark their performance, finding that SBERT models fine-tuned on SPICED are best suited to the task.

Next, I show how models trained on SPICED are beneficial for multiple downstream applications. First, I show how models trained on SPICED perform significantly better on the task of evidence retrieval for scientific fact checking, despite differences in the domain and source of scientific claims. Then, I perform several large scale analyses of science communication using models trained on SPICED as well as the exaggeration detection dataset from Chapter 8. I make three primary observations in this analysis:

1. General news outlets systematically express higher information change than press releases and science and technology news outlets.
2. Verified users and users with more followers express higher information change on average than organizational accounts.
3. Findings as expressed in the limitations sections of papers tend to be exaggerated more in the media.

Importantly, I show that models trained on SPICED can be used to reveal large scale trends in science communication, making the dataset and models useful for answering new research questions about how the message of science changes across media. These results also show that one shouldn't ignore the full-text of a paper when analyzing science communication, as stark differences exist between different sections of a paper in terms of how the message can change. These results and resources represent a new way to think about and study the problem of scientific misinformation.

[1] https://www.prolific.co/

1.5 Towards Scientific Language Understanding

The components of this thesis culminate into the first study on information change in science communication within natural language processing. As an entry point to building tools for combating scientific misinformation, I first develop new methods for general domain fact checking and scientific fact checking given the availability of data. I then contribute tools for modeling and dataset creation in order to assist with building new resources for new tasks in scientific language understanding. I gradually build upon various tasks in scientific language understanding, ultimately defining information change as an important and useful task for automatically analyzing scientific texts at all stages of the science communication pipeline. At the same time, there is still much work to be done in order to improve datasets, methods, and problem formulations for ensuring information quality in science communication.

1.5.1 Datasets

The datasets developed in this thesis, while demonstrated to be useful, are still rather limited in size. The dataset presented in Chapter 8 consists of only 653 samples, and the dataset in Chapter 9 only 6,000. Additionally, they have limited scope, covering only the most popular scientific disciplines. As such, more resources should be invested in building larger and more comprehensive datasets for these tasks; in particular, most of those resources should be invested in developing difficult *test sets*. In my view, the main need for more data is in order to track progress on these tasks as opposed to developing more accurate models. We should follow the current trend in NLP and machine learning to build models and methods which require less training data in order to mitigate the expense of annotation and collection of data. It therefore makes more sense to invest more resources into making difficult testing data covering a broad range of fields, topics, and tasks.

1.5.2 Methods

In line with the need for larger testing data, new methods should be created for working with limited scientific training data. Massive language models the likes of GPT-3 [39] are capable of impressive few- and zero-shot performance on general domain text. Given the training sets available in popular scientific domains, one avenue of research could be to first determine how to insert appropriate domain knowledge in a prompt-based fashion to perform well on those domains, as well as to explore how to develop prompting methods which transfer across domains. Those practices could then be applied to new scientific domains and tasks, where the main expense would come from hiring experts to develop small sets of prompts as opposed to hand annotating large training sets.

1.5.3 Problem Formulations

The problem formulation presented in Chapter 9 poses measuring information change between scientific sentences very generally. This is useful for revealing trends in science communication with very broad strokes. For example, we can ask research questions such as “to what degree do different organizations change the message of science?” and “how do different social factors affect degree of information change?”, which are answerable with sufficiently large sets of unlabeled data.

Narrowing down the specific types of information change and strategies used by organizations is a different story. In the current formulation, one can narrow down the types of information change in a pipeline fashion, first by matching findings (considering a matched pair to be one where the IMS exceeds a certain threshold) and then performing a second analysis (human or machine) to identify what information changes between the pair. This is the setup used in Section 9.6.3 to determine what sections of a scientific paper tend to be overstated. The problem in this setup is: what types of information change do we care about measuring? Certain types of information change have been identified as being prevalent in science communication [224, 38, 77], but as of now there isn’t a central resource defining all of them. I would argue that an important next step in building tools for measuring information change in science is to define a **taxonomy of information change in science**. Such changes should be of societal relevance and prevalent in science communication. Examples of changes that could be included in such a taxonomy are veracity, exaggeration, certainty, and cherry-picking.

Once such a taxonomy is defined, the next step is to determine how to automatically identify the specific changes listed in that taxonomy. Training a model for each type of information change would be cumbersome, potentially requiring separate training data for each label of interest. As such, new methods in the areas of learning from limited data explored in this thesis could be useful for overcoming the need to develop specialized training sets for every type of information change. For example, methods in multi-task learning, domain adaptation, and prompt-based learning could prove useful, given the proper injection of domain expert knowledge into the model. Large generative models (e.g. GPT-3) could also be one avenue to explore given their impressive zero-shot ability. Additionally, these models have an even more promising feature of potentially being able to explain their predictions in natural language. An ideal model would be able to both mark the types of changes occurring between two scientific sentences, as well as explain exactly how those changes appear in text.

One limitation of the current problem formulation is that I consider the matching problem to be 1-to-1. In practice, one may wish to compare a scientific sentence to multiple sources, and it may require integrating several different scientific findings to determine how a piece of science communication gets the message right and wrong. While it is possible in the current setup to simply rank multiple sentences and select any sentences above a certain threshold as the body for comparison, how to consolidate all of that information and select appropriate thresholds is something to explore in future work.

A final consideration is: what do we consider to be “truth”? This is a problem in fact checking as well, where one must decide what source of information is considered the ground truth state of the world e.g. Wikipedia [230]. In this thesis, I have assumed that scientific documents represent truth; in the real world, this isn’t always the case [114, 228, 218, 40]. In fact, it is in the nature of science to change, and what is considered truth at one point in time will likely be disproved, replaced, or amended at a later point. It is then a critical next step to conjure new ways of predicting the trustworthiness and accuracy of scientific papers. This is a difficult task, since without a set ground truth, how does one know whether a scientific article is accurate? One could consider social factors such as the number of citations a paper has or the track record of the authors, but this route is fraught with potential for inadvertent biases and missteps. It is therefore, in my opinion, less a question to be answered solely by computer scientists, but an important question to be engaged with in an interdisciplinary conversation between social scientists, science of science researchers, ethicists, and the public.

The following chapters are prints of the various peer-reviewed papers which constitute this work, and hopefully represent a strong contribution in the area of natural language processing for scientific language understanding.

References for Presented Papers

Wright, D., & Augenstein, I. (2020). Claim check-worthiness detection as positive unlabelled learning. In *Findings of EMNLP*. Association for Computational Linguistics.

Atanasova, P.*, **Wright, D.***, & Augenstein, I. (2020). Generating label cohesive and well-formed adversarial claims. In *EMNLP 2020*. Association for Computational Linguistics.

* denotes equal contribution

Wright, D., & Augenstein, I. (2020). Transformer based multi-source domain adaptation. In *EMNLP 2020*. Association for Computational Linguistics.

Wright, D., & Augenstein, I. (2022). Multi-View Knowledge Distillation from Crowd Annotations for Out-of-Domain Generalization. *arXiv preprint arXiv:*.

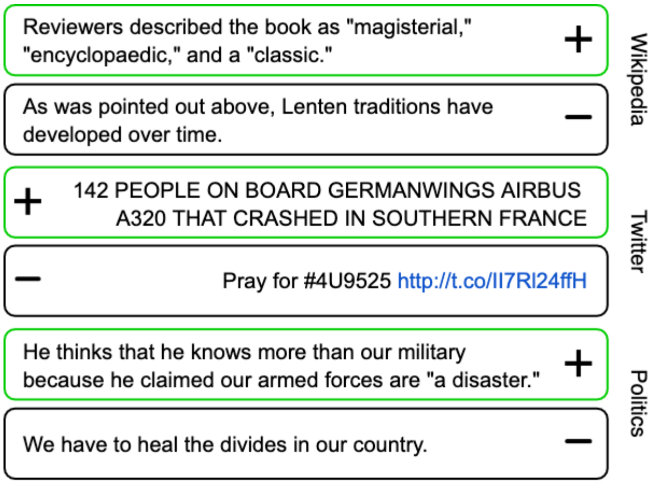
Wright, D., & Augenstein, I. (2021). CiteWorth: Cite-Worthiness Detection for Improved Scientific Document Understanding. In *Findings of ACL 2021*. Association for Computational Linguistics.

Wright, D., & Augenstein, I. (2021). Semi-Supervised Exaggeration Detection of Health Science Press Releases. In *EMNLP 2021*. Association for Computational Linguistics.

Wright, D., Wadden, D., Lo, K., Kuehl, B., Cohan, A., Augenstein, I., & Wang, L. L. (2022). Generating Scientific Claims for Zero-Shot Scientific Fact Checking. In *ACL 2022*. Association for Computational Linguistics.

Wright, D.*, Pei J.*, Jurgens D., & Augenstein, I. (2022). Modeling Information Change in Science Communication with Semantically Matched Paraphrases. In *EMNLP 2022*. Association for Computational Linguistics.

* denotes equal contribution



Reviewers described the book as "magisterial," "encyclopaedic," and a "classic." +

As was pointed out above, Lenten traditions have developed over time. −

+ 142 PEOPLE ON BOARD GERMANWINGS AIRBUS A320 THAT CRASHED IN SOUTHERN FRANCE

− Pray for #4U9525 http://t.co/Il7Rl24ffH

He thinks that he knows more than our military because he claimed our armed forces are "a disaster." +

We have to heal the divides in our country. −

Figure 2.1: Examples of check-worthy and non check-worthy statements from three different domains. Check-worthy statements are those which were judged to require evidence or a fact check.

2 Claim Check-Worthiness Detection as Positive Unlabelled Learning

2.1 Introduction

Misinformation is being spread online at ever increasing rates [59] and has been identified as one of society's most pressing issues by the World Economic Forum [113]. In response, there has been a large increase in the number of organizations performing fact checking [90]. However, the rate at which misinformation is introduced and spread vastly outpaces the ability of any organization to perform fact checking, so only the most salient claims are checked. This obviates the need for being able to automatically find check-worthy content online and verify it.

The natural language processing and machine learning communities have recently begun to address the problem of automatic fact checking [238, 102, 229, 13, 10, 11, 175, 3]. The first step of automatic fact checking is claim check-worthiness detection, a text classification problem where, given a statement, one must predict if the content of that statement makes “an assertion about the world that is checkable” [127]. There are multiple isolated lines of research which have studied variations of this problem. Figure 2.1 provides examples from three tasks which are studied in this work: rumour detection on Twitter [277, 275], check-worthiness ranking in political debates and speeches [8, 72, 22], and citation needed detection on Wikipedia [196]. Each task is concerned with a shared underlying problem: detecting claims which warrant further verification. However, no work has been done to compare all three tasks to understand shared challenges in order to derive shared solutions, which could enable improving claim check-worthiness detection systems across multiple domains.

Therefore, we ask the following main research question in this work: are these all variants of the same task, and if so, is it possible to have a unified approach to all of them? We answer this question by investigating the problem of annotator subjectivity, where annotator background and expertise causes their judgement of what is check-worthy to differ, leading to false negatives in the data [127]. Our proposed solution is *Positive Unlabelled Conversion (PUC)*, an extension of Positive Unlabelled (PU) learning, which converts negative instances into positive ones based on the estimated prior probability of an example being positive. We demonstrate that a model trained using *PUC* improves performance on English *citation needed detection* and *Twitter rumour detection*. We also show that by pretraining a model on citation needed detection, one can further improve results on Twitter rumour detection over a model trained solely on rumours, highlighting that a unified approach to these problems is achievable. Additionally, we show that one attains better results on *political speeches* check-worthiness ranking without using any form of PU learning, arguing through a dataset analysis that the labels are much more subjective than the other two tasks.

The **contributions** of this work are as follows:

1. The first thorough comparison of multiple claim check-worthiness detection tasks.
2. *Positive Unlabelled Conversion (PUC)*, a novel extension of PU learning to support check-worthiness detection across domains.
3. Results demonstrating that a unified approach to check-worthiness detection is achievable for 2 out of 3 tasks, improving over the state-of-the-art for those tasks.

2.2 Related Work

2.2.1 Claim Check-Worthiness Detection

As the first step in automatic fact checking, claim check-worthiness detection is a binary classification problem which involves determining if a piece of text makes “an assertion about the world which can be checked” [127]. We adopt this broad definition as it allows us to perform a structured comparison of many publicly available datasets. The wide applicability of the definition also allows us to study if and how a unified cross-domain approach could be developed.

Claim check-worthiness detection can be subdivided into three distinct domains: rumour detection on Twitter, check-worthiness ranking in political speeches and debates, and citation needed detection on Wikipedia. A few studies have been done which attempt to create full systems for mining check-worthy statements, including the works of [127], ClaimRank [115], and ClaimBuster [102]. They develop full software systems consisting of relevant source material retrieval, check-worthiness classification, and dissemination to the public via end-user applications. These works are focused solely on the political domain, using data from political TV shows, speeches, and debates. In contrast, in this work we study the claim check-worthiness detection problem across three domains which have publicly available data: Twitter [276], political speeches [8], and Wikipedia [196].

Rumour Detection on Twitter Rumour detection on Twitter is primarily studied using the PHEME dataset [277], a set of tweets and associated threads from breaking news events which are either rumourous or not. Published systems which perform well on this task include contextual models (e.g. conditional random fields) acting on a tweet's thread [276, 275], identifying salient rumour-related words [1], and using a GAN to generate misinformation in order to improve a downstream discriminator [152].

Political Speeches For political speeches, the most studied datasets come from the Clef CheckThat! shared tasks [8, 72, 22] and ClaimRank [115]. The data consist of transcripts of political debates and speeches where each sentence has been annotated by an independent news or fact-checking organization for whether or not the statement should be checked for veracity. The most recent and best performing system on the data considered in this paper consists of a two-layer bidirectional GRU network which acts on both word embeddings and syntactic parse tags [97]. In addition, they augment the native dataset with weak supervision from unlabelled political speeches.

Citation Needed Detection Wikipedia citation needed detection has been investigated recently in [196]. The authors present a dataset of sentences from Wikipedia labelled for whether or not they have a citation attached to them. They also released a set of sentences which have been flagged as not having a citation but needing one (i.e. *unverified*). In contrast to other check-worthiness detection domains, there are much more training data available on Wikipedia. However, the rules for what requires a citation do not necessarily capture all "checkable" statements, as "all material in Wikipedia articles must be verifiable" [196]. Given this, we view Wikipedia citation data as a set of positive and unlabelled data: statements which have attached citations are positive samples of check-worthy statements, and within the set of statements without citations there exist some positive samples (those needing a citation) and some negative samples. Based on this, this domain constitutes the most general formulation of check-worthiness among the domains we consider. Therefore, we experiment with using data from this domain as a source for transfer learning, training variants of PU learning models on it, then applying them to target data from other domains.

2.2.2 Positive Unlabelled Learning

PU learning methods attempt to learn good binary classifiers given only positive labelled and unlabelled data. Recent applications where PU learning has been shown to be beneficial include

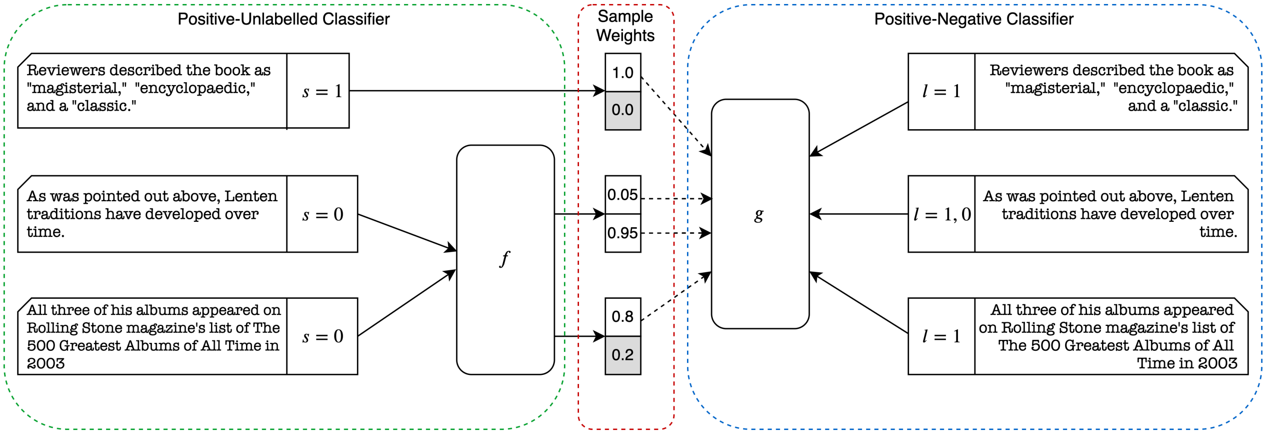

Positive-Unlabelled Classifier
Sample Weights
Positive-Negative Classifier
Reviewers described the book as "magisterial," "encyclopaedic," and a "classic."
$s = 1$
As was pointed out above, Lenten traditions have developed over time.
$s = 0$
All three of his albums appeared on Rolling Stone magazine's list of The 500 Greatest Albums of All Time in 2003
$s = 0$
f
1.0
0.0
0.05
0.95
0.8
0.2
g
$l = 1$
Reviewers described the book as "magisterial," "encyclopaedic," and a "classic."
$l = 1, 0$
As was pointed out above, Lenten traditions have developed over time.
$l = 1$
All three of his albums appeared on Rolling Stone magazine's list of The 500 Greatest Albums of All Time in 2003

Figure 2.2: High level view of *PUC*. A PU classifier (f, green box) is first learned using PU data (with s indicating if the sample is positive or unlabelled). From this the prior probability of a sample being positive is estimated. Unlabelled samples are then ranked by f (red box) and the most positive samples are converted into positives until the dataset is balanced according to the estimated prior. The model g is then trained using the duplication and weighting method of [71] as described in §2.3.2 with labels l (blue box). Greyed out boxes are negative weights which are ignored when training the classifier g, as those examples are only trained as positives.

detecting deceptive reviews online [139, 199], keyphrase extraction [221] and named entity recognition [181]. For a survey on PU learning, see [25], and for a formal definition of PU learning, see §2.3.2.

Methods for learning positive-negative (PN) classifiers from PU data have a long history [61, 58, 137], with one of the most seminal papers being from [71]. In this work, the authors show that by assuming the labelled samples are a random subset of all positive samples, one can utilize a classifier trained on PU data in order to train a different classifier to predict if a sample is positive or negative. The process involves training a PN classifier with positive samples being shown to the classifier once and *unlabelled* samples shown as *both* a positive sample and a negative sample. The loss for the duplicated samples is weighted by the confidence of a PU classifier that the sample is positive.

Building on this, du Plessis et al. [67] propose an unbiased estimator which improves the estimator introduced in [71] by balancing the loss for positive and negative classes. The work of Kiryo et al. [126] extends this method to improve the performance of deep networks on PU learning. Our work builds on the method of Elkan and Noto [71] by relabelling samples which are highly confidently positive.

2.3 Methods

The task considered in this paper is to predict if a statement makes “an assertion about the world that is checkable” [127]. As the subjectivity of annotations for existing data on claim check-worthiness detection is a known problem [127], we view the data as a set of positive and unlabelled (PU) data. In addition, we unify our approach to each of them by viewing Wikipedia data as an abundant source corpus. Models are then trained on this source corpus using variants of PU learning and transferred via fine-tuning to the other claim check-worthiness detection datasets, which are subsequently trained on as PU data. On top of vanilla PU learning, we introduce *Positive Unlabelled Conversion (PUC)* which relabels examples that are most confidently positive in the unlabelled data. A formal task definition, description of PU learning, and explanation of the *PUC* extension are given in the following sections.

2.3.1 Task Definition

The fundamental task is binary text classification. In the case of positive-negative (PN) data, we have a labelled dataset $\mathcal{D} : \{(x, y)\}$ with input features $x \in \mathbb{R}^d$ and labels $y \in \{0, 1\}$. The goal is to learn a classifier $g : x \rightarrow (0, 1)$ indicating the probability that the input belongs to the positive class. With PU data, the dataset $\mathcal{D}$ instead consists of samples $\{(x, s)\}$, where the value $s \in \{0, 1\}$ indicates if a sample is labelled or not. The primary difference from the PN case is that, unlike for the labels y, a value of $s = 0$ does not denote the sample is negative, but that the label is unknown. The goal is then to learn a PN classifier g using a PU classifier $f : x \rightarrow (0, 1)$ which predicts whether or not a sample is labelled [71].

2.3.2 PU Learning

Our overall approach is depicted in Figure 2.2. We begin with an explanation of the PU learning algorithm described in [71]. Assume that we have a dataset randomly drawn from some probability distribution $p(x, y, s)$, where samples are of the form $(x, s),\ s \in \{0, 1\}$ and $s = 1$ indicates that the sample is labelled. The variable y is unknown, but we make two assumptions which allow us to derive an estimator for probabilities involving y. The first is that:

$$p(y = 0|s = 1) = 0 \tag{2.1}$$

In other words, if we know that a sample is labelled, then that label cannot be 0. The second assumption is that labelled samples are Selected Completely At Random from the underlying distribution (also known as the SCAR assumption). Check-worthiness data can be seen as an instance of SCAR PU data; annotators tend to only label those instances which are very clearly check-worthy in *their* opinion [127]. When combined across several annotators, we assume this leads to a random sample from the total set of check-worthy statements.

Given this, a classifier $f : x \rightarrow (0, 1)$ is trained to predict $p(s = 1|x)$ from the PU data. It is then employed to train a classifier g to predict $p(y = 1|x)$ by first estimating $c = p(s = 1|y = 1)$ on a set of validation data. Considering a validation set V where $P \subset V$ is the set of positive samples in V, c is estimated as:

$$c \approx \frac{1}{|P|} \sum_{x \in P} f(x) \tag{2.2}$$

This says our estimate of $p(s = 1|y = 1)$ is the average confidence of our classifier on known positive samples. Next, we can estimate $E_{p(x,y,s)}[h(x, y)]$ for any arbitrary function h empirically from a dataset of k samples as follows:

$$E[h] = \frac{1}{k}(\sum_{(x,s=1)} h(x, 1) + \sum_{(x,s=0)} w(x)h(x, 1) + (1 - w(x))h(x, 0)) \tag{2.3}$$

$$w(x) = p(y = 1|x, s = 0) = \frac{1 - c}{c} \frac{p(s = 1|x)}{1 - p(s = 1|x)} \tag{2.4}$$

In this case, c is estimated using Equation 2.2 and $p(s = 1|x)$ is estimated using the classifier f. The derivations for these equations can be found in [71].

To estimate $p(y = 1|x)$ empirically, the unlabelled samples in the training data are duplicated, with one copy negatively labelled and one copy positively labelled. Each copy is trained on with a weighted loss $w(x)$ when the label is positive and $1 - w(x)$ when the label is negative. Labelled samples are trained on normally (i.e. a single copy with unit weight).

2.3.3 Positive Unlabelled Conversion

For *PUC*, the motivation is to relabel those samples from the unlabelled data which are very clear cut positive. To accomplish this, we start with the fact that one can also estimate the prior probability of a sample having a positive label using f. If instead of h we want to estimate $E[y] = p(y = 1)$, the following is obtained:

$$p(y = 1) \approx \frac{1}{k}(\sum_{x,s=1} 1 + \sum_{x,s=0} w(x)) \tag{2.5}$$

This estimate is then utilized to convert the most confident unlabelled samples into positives. First, all of the unlabelled samples are ranked according to their calculated weight $w(x)$. The ranked samples are then iterated through and converted into positive-only samples until the distribution of positive samples is greater than or equal to the estimate of $p(y = 1)$. Unlike in vanilla PU learning, these samples are discretized to have a positive weight of 1, and trained on by the classifier g once per epoch as positive samples along with the labelled samples. The remaining unlabelled data are trained on in the same way as in vanilla PU learning.

2.3.4 Implementation

In order to create a unified approach to check-worthiness detection, transfer learning from Wikipedia citation needed detection is employed. To accomplish this, we start with a training dataset $\mathcal{D}^s$ of statements from Wikipedia featured articles that are either labelled as containing a citation (positive) or unlabelled. We train a classifier f^s on this dataset and obtain a classifier g^s via *PUC*. For comparison, we also train models with vanilla PU learning and PN learning as baselines. The network architecture for both f^s and g^s is BERT [62], a large pretrained transformer-based [237] language model. We use the HuggingFace transformers implementation of the 12-layer 768 dimensional variation of BERT [251]. The classifier in this implementation is a two layer neural network acting on the `[CLS]` token.

From g^s, we train a classifier g^t using downstream check-worthiness detection dataset D^t by initializing g^t with the base BERT network from g^s and using a new randomly initialized final layer. In addition, we train a model f^t on the target dataset, and train g^t with *PUC* from this model to obtain the final classifier. As a baseline, we also experiment with training on just the dataset D^t without any pretraining. In the case of citation needed detection, since the data comes from the same domain we simply test on the test split of statements labelled as “citation needed” using the classifier g^s. We compare our models to the published state of the art baselines on each dataset.

For all of our models (f^s, g^s, f^t, g^t) we train for two epochs, saving the weights with the best F1 score on validation data as the final model. Training is performed with a max learning rate of 3e-5 and a triangular learning rate schedule [112] that linearly warms up for 200 training steps, then linearly decays to 0 for the rest of training. For regularization we add L2 loss with a coefficient of 0.01, and dropout with a rate of 0.1. Finally, we split the training sets into 80% train and 20% validation, and train with a batch size of 8. The code to reproduce our experiments can be found here.[2]

2.4 Experimental Results

To what degree is claim check-worthiness detection a PU learning problem, and does this enable a unified approach to check-worthiness detection? In our experiments, we progressively answer this question by answering the following: 1) is PU learning beneficial for the tasks considered? 2) Does PU citation needed detection transfer to rumour detection? 3) Does PU citation needed detection transfer to political speeches? To investigate how well the data in each domain reflects the definition of a check-worthy statement as one which “makes an assertion about the world which is checkable” and thus understand subjectivity in the annotations, we perform a dataset analysis comparing the provided labels of the top ranked check-worthy claims from the *PUC* model with the labels given by two human annotators. In all experiments, we report the mean performance of our models and standard deviation across 15 different random seeds. Additionally, we report the performance of each model ensembled across the 15 runs through majority vote on each sample.

[2] `https://github.com/copenlu/check-worthiness-pu-learning`

Method	P	R	F1	eP	eR	eF1
[196]	75.3	70.9	73.0 [76.0]*	-	-	-
BERT	78.8 ± 1.3	83.7 ± 4.5	81.0 ± 1.5	79.0	85.3	82.0
BERT + PU	**78.8 ± 0.9**	84.3 ± 3.0	81.4 ± 1.0	79.0	85.6	82.2
BERT + *PUC*	78.4 ± 0.9	**85.6 ± 3.2**	**81.8 ± 1.0**	78.6	**87.1**	**82.6**

Table 2.1: F1 and ensembled F1 score for citation needed detection training on the FA split and testing on the LQN split of [196]. The FA split contains statements with citations from featured articles and the LQN split consists of statements which were flagged as not having a citation but needing one. Listed are the mean, standard deviation, and ensembled results across 15 seeds (eP, eR, and eF1). **Bold** indicates best performance, underline indicates second best. *The reported value is from rerunning their released model on the test dataset. The value in brackets is the value reported in the original paper.

2.4.1 Datasets

Wikipedia Citations We use the dataset from [196] for citation needed detection. The dataset is split into three sets: one coming from featured articles (deemed ‘high quality’, 10k positive and 10k negative statments), one of statements which have no citation but have been flagged as needing one (10k positive, 10k negative), and one of statements from random articles which have citations (50k positive, 50k negative). In our experiments the models were trained on the high quality statements from featured articles and tested on the statements which were flagged as ‘citation needed’. The key differentiating features of this dataset from the other two datasets are: 1) the domain of text is Wikipedia and 2) annotations are based on the decisions of Wikipedia editors following Wikipedia guidelines for citing sources[3].

Twitter Rumours The PHEME dataset of rumours is employed for Twitter claim check-worthiness detection [277]. The data consists of 5,802 annotated tweets from 5 different events, where each tweet is labelled as rumourous or non-rumourous (1,972 rumours, 3,830 non-rumours). We followed the leave-one-out evaluation scheme of [276], namely, we performed a 5-fold cross-validation for all methods, training on 4 events and testing on 1. The key differentiating features of this dataset from the other two datasets are: 1) the domain of data is tweets and 2) annotations are collected from professional journalists specifically for building a dataset to train machine learning models.

Political Speeches The dataset we adopted in the political speeches domain is the same as in [97], consisting of 4 political speeches from the 2018 Clef CheckThat! competition [8] and 3 political speeches from ClaimRank [115] (2,602 statements total). We performed a 7-fold cross-validation, using 6 splits as training data and 1 as test in our experimental setup. The data from ClaimRank is annotated using the judgements from 9 fact checking organizations, and the

[3] https://en.wikipedia.org/wiki/Wikipedia:Citing_sources

Method	μP	μR	μF1	eP	eR	eF1
[276]	66.7	55.6	60.7	-	-	-
BiLSTM	62.3	56.4	59.0	-	-	-
BERT	69.9 ± 1.7	60.8 ± 2.6	65.0 ± 1.3	71.3	61.9	66.3
BERT + Wiki	69.3 ± 1.6	61.4 ± 2.6	65.1 ± 1.2	70.7	62.2	66.2
BERT + WikiPU	69.9 ± 1.3	62.5 ± 1.6	66.0 ± 1.1	**72.2**	64.6	68.2
BERT + Wiki*PUC*	**70.1** ± **1.1**	61.8 ± 1.8	65.7 ± 1.0	71.5	62.7	66.8
BERT + PU	68.7 ± 1.2	64.7 ± 1.8	66.6 ± 0.9	69.9	65.2	67.5
BERT + *PUC*	68.1 ± 1.5	65.3 ± 1.6	66.6 ± 0.9	69.1	66.3	67.7
BERT + PU + WikiPU	68.4 ± 1.2	**66.1** ± **1.2**	**67.2** ± **0.6**	69.3	67.2	68.3
BERT + *PUC* + WikiPUC	68.0 ± 1.4	66.0 ± 2.0	67.0 ± 1.3	69.4	**67.5**	**68.5**

Table 2.2: micro-F1 (μF1) and ensembled F1 (eF1) performance of each system on the PHEME dataset. Performance is averaged across the five splits of [276]. Results show the mean, standard deviation, and ensembled score across 15 seeds. **Bold** indicates best performance, underline indicates second best.

data from Clef 2018 is annotated by factcheck.org. The key differentiating features of this dataset from the other two datasets are: 1) the domain of data is transcribed spoken utterances from political speeches and 2) annotations are taken from 9 fact checking organizations gathered independently.

2.4.2 Is PU Learning Beneficial for Citation Needed Detection?

Our results for citation needed detection are given in Table 2.1. The vanilla BERT model already significantly outperforms the state of the art model from Redi et al. [196] (a GRU network with global attention) by 6 F1 points. We see further gains in performance with PU learning, as well as when using *PUC*. Additionally, the models using PU learning have lower variance, indicating more consistent performance across runs. The best performing model we see is the one trained using *PUC* with an F1 score of 82.6. We find that this confirms our hypothesis that citation data is better seen as a set of positive and unlabelled data when used for check-worthiness detection. In addition, it gives some indication that PU learning improves the generalization power of the model, which could make it better suited for downstream tasks.

2.4.3 Does PU Citation Needed Detection Transfer to Rumour Detection?

2.4.3.1 Baselines

The best published method that we compare to is the CRF from [276]. which utilizes a combination of content and social features. Content features include word vectors, part-of-speech tags, and various lexical features, and social features include tweet count, listed count, follow ratio, age, and whether or not a user is verified. The CRF acts on a timeline of tweets, making it contextual. In addition, we include results from a 2-layer BiLSTM with FastText embeddings [35]. There exist other deep learning models which have been developed for this task, including [152] and [1], but they do not publish results on the standard splits of the data and we were unable to recreate their results, and thus are omitted.

2.4.3.2 Results

The results for the tested systems are given in Table 2.2. Again we see large gains from BERT based models over the baseline from [276] and the 2-layer BiLSTM. Compared to training solely on PHEME, fine tuning from basic citation needed detection sees little improvement (0.1 F1 points). However, fine tuning a model trained using PU learning leads to an increase of 1 F1 point over the non-PU learning model, indicating that PU learning enables the Wikipedia data to be useful for transferring to rumour detection i.e. the improvement is not only from a better semantic representation learned from Wikipedia data. For *PUC*, we see an improvement of 0.7 F1 points over the baseline and lower overall variance than vanilla PU learning, meaning that the results with *PUC* are more consistent across runs. The best performing models also use PU learning on in-domain data, with the best average performance being from the models trained using PU/*PUC* on in domain data and initialized with weights from a Wikipedia model trained using PU/*PUC*. When models are ensembled, pretraining with vanilla PU learning improves over no pretraining by almost 2 F1 points, and the best performing models which are also trained using PU learning on in domain data improve over the baseline by over 2 F1 points. We conclude that framing rumour detection on Twitter as a PU learning problem leads to improved performance.

Based on these results, we are able to confirm two of our hypotheses. The first is that Wikipedia citation needed detection and rumour detection on Twitter are indeed similar tasks, and a unified approach for both of them is possible. Pretraining a model on Wikipedia provides a clear downstream benefit when fine-tuning on Twitter data, *precisely when PU/PUC is used*. Additionally, training using *PUC* on in domain Twitter data provides further benefit. This shows that *PUC* constitutes a unified approach to these two tasks.

The second hypothesis we confirm is that both Twitter and Wikipedia data are better seen as positive and unlabelled for claim check-worthiness detection. When pretraining with the data as a traditional PN dataset there is no performance gain and in fact a performance loss when the models are ensembled. PU learning allows the model to learn better representations for general claim check-worthiness detection.

To explain why this method performs better, Table 2.1 and Table 2.2 show that *PUC* improves model recall at very little cost to precision. The aim of this is to mitigate the issue of subjectivity in the annotations of check-worthiness detection datasets noted in previous work [127]. Some of the effects of this are illustrated in Table A.1 and Table A.2 in §A.1. The *PUC* models are better at distinguishing rumours which involve claims of fact about people i.e. things that people said or did, or qualities about people. For non-rumours, the *PUC* pretrained model is better

Method	MAP
[127]	26.7
[97]	30.2
BERT	33.0 ± 1.8
BERT + Wiki	**34.4 ± 2.7**
BERT + WikiPU	33.2 ± 1.7
BERT + Wiki*PUC*	31.7 ± 1.8
BERT + PU	18.8 ± 3.7
BERT + *PUC*	26.7 ± 2.8
BERT + PU + WikiPU	16.8 ± 3.5
BERT + *PUC* + Wiki*PUC*	27.8 ± 2.7

Table 2.3: Mean average precision (MAP) of models on political speeches. **Bold** indicates best performance, underline indicates second best.

at recognizing statements which describe qualitative information surrounding the events and information that is self-evident e.g. a tweet showing the map where the Charlie Hebdo attack took place.

2.4.4 Does PU Citation Needed Detection Transfer to Political Speeches?

2.4.4.1 Baselines

The baselines we compare to are the state of the art models from [97] and [127]. The model from [127] consists of InferSent embeddings [52] concatenated with POS tag and NER features passed through a logistic regression classifier. The model from [97] is a bidirectional GRU network acting on syntatic parse features concatenated with word embeddings as the input representation.

2.4.4.2 Results

The results for political speech check-worthiness detection are given in Table 2.3. We find that the BERT model initialized with weights from a model trained on plain Wikipedia citation needed statements performs the best of all models. As we add transfer learning and PU learning, the performance steadily drops. We perform a dataset analysis to gain some insight into this effect in §2.4.5.

2.4.5 Dataset Analysis

In order to understand our results in the context of the selected datasets, we perform an analysis to learn to what extent the positive samples in each dataset reflect the definition of a check-worthy claim as "an assertion about the world that is checkable". We ranked all of the statements based on the predictions of 15 *PUC* models trained with different seeds, where more positive class predictions means a higher rank (thus more check-worthy), and had two experts manually relabel the top 100 statements. The experts were informed to label the statements based on

Dataset	P	R	F1
Wikipedia	81.7	87.0	84.3
	84.8	87.0	85.9
	83.3	*87.0*	*85.1*
Twitter	87.5	82.4	84.8
	86.3	81.2	83.6
	86.9	*81.8*	*84.2*
Politics	33.8	89.3	49.0
	31.1	100.0	47.5
	32.5	*94.7*	*48.3*

Table 2.4: F1 score comparing manual relabelling of the top 100 predictions by *PUC* model with the original labels in each dataset by two different annotators. *Italics* are average value between the two annotators.

the definition of check-worthy given above. We then compared the manual annotation to the original labels using F1 score. Higher F1 score indicates the dataset better reflects the definition of check-worthy we adopt in this work. Our results are given in Table 2.4.

We find that the Wikipedia and Twitter datasets contain labels which are more general, evidenced by similar high F1 scores from both annotators ($>$ 80.0). For political speeches, we observe that the human annotators both found many more examples to be check-worthy than were labelled in the dataset. This is evidenced by examples such as *It's why our unemployment rate is the lowest it's been in so many decades* being labelled as not check-worthy and *New unemployment claims are near the lowest we've seen in almost half a century* being labelled as check-worthy in the same document in the dataset's original annotations. This characteristic has been noted for political debates data previously [127], which was also collected using the judgements of independent fact checking organizations [83]. Labels for this dataset were collected from various news outlets and fact checking organizations, which may only be interested in certain types of claims such as those most likely to be false. This makes it difficult to train supervised machine learning models for general check-worthiness detection based solely on text content and document context due to labelling inconsistencies.

2.5 Discussion and Conclusion

In this work, we approached claim check-worthiness detection by examining how to unify three distinct lines of work. We found that check-worthiness detection is challenging in any domain as there exist stark differences in how annotators judge what is check-worthy. We showed that one can correct for this and improve check-worthiness detection across multiple domains by using positive unlabelled learning. Our method enabled us to perform a structured comparison of datasets in different domains, developing a unified approach which outperforms state of the art in 2 of 3 domains and illuminating to what extent these datasets reflect a general definition of check-worthy.

Future work could explore different neural base architectures. Further, it could potentially benefit all tasks to consider the greater context in which statements are made. We would also like to acknowledge again that all experiments have only focused on English language datasets; developing models for other, especially low-resource languages, would likely result in additional challenges. We hope that this work will inspire future research on check-worthiness detection, which we see as an under-studied problem, with a focus on developing resources and models across many domains such as Twitter, news media, and spoken rhetoric.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 801199.

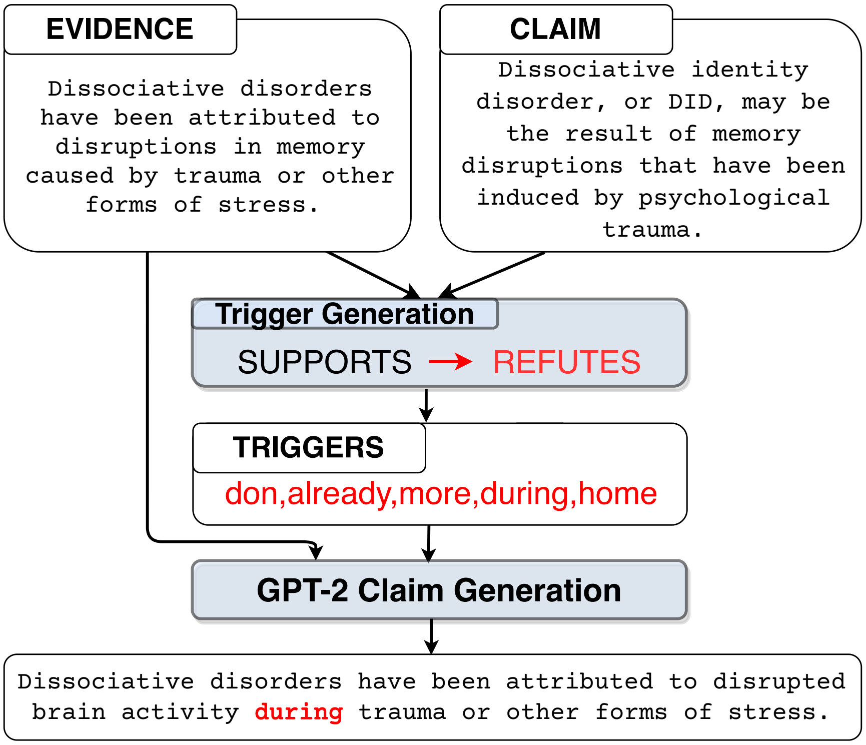

EVIDENCE
Dissociative disorders have been attributed to disruptions in memory caused by trauma or other forms of stress.
CLAIM
Dissociative identity disorder, or DID, may be the result of memory disruptions that have been induced by psychological trauma.
Trigger Generation
SUPPORTS → REFUTES
TRIGGERS
don,already,more,during,home
GPT-2 Claim Generation
Dissociative disorders have been attributed to disrupted brain activity during trauma or other forms of stress.

Figure 3.1: High level overview of our method. First, universal triggers are discovered for flipping a source to a target label (e.g. SUPPORTS → REFUTES). These triggers are then used to condition the GPT-2 language model to generate novel claims with the original label, including at least one of the found triggers.

3 Generating Label Cohesive and Well-Formed Adversarial Claims

3.1 Introduction

Adversarial examples [87, 226] are deceptive model inputs designed to mislead an ML system into making the wrong prediction. They expose regions of the input space that are outside the training data distribution where the model is unstable. It is important to reveal such vulnerabilities and correct for them, especially for tasks such as fact checking (FC).

In this paper, we explore the vulnerabilities of FC models trained on the FEVER dataset [230], where the inference between a claim and evidence text is predicted. We particularly construct *universal adversarial triggers* [241] – single n-grams appended to the input text that can shift the prediction of a model from a source class to a target one. Such adversarial examples are of particular concern, as they can apply to a large number of input instances.

However, we find that the triggers also change the meaning of the claim such that the true label is in fact the target class. For example, when attacking a claim-evidence pair with a ‘SUPPORTS’ label, a common unigram found to be a universal trigger when switching the label to 'REFUTES' is 'none'. Prepending this token to the claim drastically changes the meaning of the claim such that the new claim is in fact a valid 'REFUTES' claim as opposed to an adversarial 'SUPPORTS' claim. Furthermore, we find adversarial examples constructed in this way to be nonsensical, as a new token is simply being attached to an existing claim.

Our **contributions** are as follows. We *preserve the meaning* of the source text and *improve the semantic validity* of universal adversarial triggers to automatically construct more potent adversarial examples. This is accomplished via: 1) a *novel extension to the HotFlip attack* [70], where we jointly minimize the target class loss of a FC model and the attacked class loss of a natural language inference model; 2) a *conditional language model* trained using GPT-2 [192], which takes trigger tokens and a piece of evidence, and generates a semantically coherent new claim containing at least one trigger. The resulting triggers maintain potency against a FC model while preserving the original claim label. Moreover, the conditional language model produces semantically coherent adversarial examples containing triggers, on which a FC model performs 23.8% worse than with the original FEVER claims. The code for the paper is publicly available.[4]

3.2 Related Work

3.2.1 Adversarial Examples

Adversarial examples for NLP systems can be constructed as automatically generated text [198] or perturbations of existing input instances [117, 70]. For a detailed literature overview, see [269].

One potent type of adversarial techniques are universal adversarial attacks [82, 241] – single perturbation changes that can be applied to a large number of input instances and that cause significant performance decreases of the model under attack. [241] find universal adversarial triggers that can change the prediction of the model using the HotFlip algorithm [70].

However, for NLI tasks, they also change the meaning of the instance they are appended to, and the prediction of the model remains correct. [161] address this by exploring only perturbed instances in the neighborhood of the original one. Their approach is for instance-dependent attacks, whereas we suggest finding *universal* adversarial triggers that also preserve the original meaning of input instances. Another approach to this are rule-based perturbations of the input [201] or imposing adversarial constraints on the produced perturbations [64]. By contrast, we extend the HotFlip method by including an auxiliary Semantic Textual Similarity (STS) objective. We additionally use the extracted universal adversarial triggers to generate adversarial examples with low perplexity.

[4] https://github.com/copenlu/fever-adversarial-attacks

3.2.2 Fact Checking

Fact checking systems consist of components to identify check-worthy claims [8, 97, 254], retrieve and rank evidence documents [264, 3], determine the relationship between claims and evidence documents [37, 14, 21], and finally predict the claims' veracity [230, 13]. As this is a relatively involved task, models easily overfit to shallow textual patterns, necessitating the need for adversarial examples to evaluate the limits of their performance.

[231] are the first to propose hand-crafted adversarial attacks. They follow up on this with the FEVER 2.0 task [232], where participants design adversarial attacks for existing FC systems. The first two winning systems [172, 104] produce claims requiring multi-hop reasoning, which has been shown to be challenging for fact checking models [175]. The other remaining system [123] generates adversarial attacks manually. We instead find universal adversarial attacks that can be applied to most existing inputs while markedly decreasing fact checking performance. [172] additionally feed a pre-trained GPT-2 model with the target label of the instance along with the text for conditional adversarial claim generation. Conditional language generation has also been employed by [120] to control the style, content, and the task-specific behavior of a Transformer.

3.3 Methods

3.3.1 Models

We take a RoBERTa [94] model pretrained with a LM objective and fine-tune it to classify claim-evidence pairs from the FEVER dataset as SUPPORTS, REFUTES, and NOT ENOUGH INFO (NEI). The evidence used is the gold evidence, available for the SUPPORTS and REFUTES classes. For NEI claims, we use the system of [156] to retrieve evidence sentences. To measure the semantic similarity between the claim before and after prepending a trigger, we use a large RoBERTa model fine-tuned on the Semantic Textual Similarity Task.[5] For further details, we refer the reader to §B.1.

3.3.2 Universal Adversarial Triggers Method

The Universal Adversarial Triggers method is developed to find n-gram trigger tokens $\mathbf{t_{ff}}$, which, appended to the original input x, $f(x) = y$, cause the model to predict a target class $\widetilde{y}$: $f(t_\alpha, x) = \widetilde{y}$. In our work, we generate unigram triggers, as generating longer triggers would require additional objectives to later produce well-formed adversarial claims. We start by initializing the triggers with the token 'a'. Then, we update the embeddings of the initial trigger tokens $\mathbf{e}_\alpha$ with embeddings $\mathbf{e}_{w_i}$ of candidate adversarial trigger tokens w_i that minimize the loss $\mathcal{L}$ for the target class $\widetilde{y}$. Following the HotFlip algorithm, we reduce the brute-force optimization

[5] https://huggingface.co/SparkBeyond/roberta-large-sts-b

problem using a first-order Taylor approximation around the initial trigger embeddings:

$$\underset{w_i \in V}{arg\,min} \left[\mathbf{e}_{w_i} - \mathbf{e}_\alpha\right]^\top \nabla_{\mathbf{e}_\alpha} \mathcal{L} \tag{3.1}$$

where $\mathcal{V}$ is the vocabulary of the RoBERTa model and $\nabla_{\mathbf{e}_\alpha}\mathcal{L}$ is the average gradient of the task loss accumulated for all batches. This approximation allows for a $\mathcal{O}(|\mathcal{V}|)$ space complexity of the brute-force candidate trigger search.

While HotFlip finds universal adversarial triggers that successfully fool the model for many instances, we find that the most potent triggers are often negation words, e.g., ‘not’, ‘neither’, ‘nowhere’. Such triggers change the meaning of the text, making the prediction of the target class correct. Ideally, adversarial triggers would preserve the original label of the claim. To this end, we propose to include an auxiliary STS model objective when searching for candidate triggers. The additional objective is used to minimize the loss $\mathcal{L}'$ for the maximum similarity score (5 out of 0) between the original claim and the claim with the prepended trigger. Thus, we arrive at the combined optimization problem:

$$\underset{w_i \in V}{arg\,min}([\mathbf{e}_{w_i} - \mathbf{e}_\alpha]^\top \nabla_{\mathbf{e}_\alpha}\mathcal{L} + [\mathbf{o}_{w_i} - \mathbf{o}_\alpha]^\top \nabla_{\mathbf{o}_\alpha}\mathcal{L}') \tag{3.2}$$

where $\mathbf{o}_w$ is the STS model embedding of word w. For the initial trigger token, we use “[MASK]” as STS selects candidates from the neighborhood of the initial token.

3.3.3 Claim Generation

In addition to finding highly potent adversarial triggers, it is also of interest to generate coherent statements containing the triggers. To accomplish this, we use the HuggingFace implementation of the GPT-2 language model [192, 251], a large transformer-based language model trained on 40GB of text. The objective is to generate a coherent claim, which either entails, refutes, or is unrelated a given piece of evidence, while also including trigger words.

The language model is first fine tuned on the FEVER FC corpus with a specific input format. FEVER consists of claims and evidence with the labels `SUPPORTS`, `REFUTES`, or `NOT ENOUGH INFO` (NEI). We first concatenate evidence and claims with a special token. Next, to encourage generation of claims with certain tokens, a sequence of tokens separated by commas is prepended to the input. For training, the sequence consists of a single token randomly selected from the original claim, and four random tokens from the vocabulary. This encourages the model to only select the one token most likely to form a coherent and correct claim. The final input format is `[trigger tokens]||[evidence]||[claim]`. Adversarial claims are then generated by providing an initial input of a series of five comma-separated trigger tokens plus evidence, and progressively generating the rest of the sequence. Subsequently, the set of generated claims is pruned to include only those which contain a trigger token, and constitute the desired

Class	F1	STS	PPL
No Triggers			
All	.866	5.139	11.92 (±45.92)
S	.938	5.130	12.22 (±40.34)
R	.846	5.139	12.14 (±37.70)
NEI	.817	5.147	14.29 (±84.45)
FC Objective			
All	.602 (±.289)	4.586 (±.328)	12.96 (±55.37)
S→R	.060 (±.034)	4.270 (±.295)	12.44 (±41.74)
S→NEI	.611 (±.360)	4.502 (±.473)	12.75 (±40.50)
R→S	.749 (±.027)	4.738 (±.052)	11.91 (±36.53)
R→NEI	.715 (±.026)	4.795 (±.094)	11.77 (±36.98)
NEI→R	.685 (±.030)	4.378 (±.232)	14.20 (±83.32)
NEI→S	.793 (±.054)	4.832 (±.146)	14.72 (±93.15)
FC+STS Objectives			
All	.763 (±.123)	4.786 (±.156)	12.97 (±58.30)
S→R	.702 (±.237)	4.629 (±.186)	12.62 (±41.91)
S→NEI	.717 (±.161)	4.722 (±.152)	12.41 (±39.66)
R→S	.778 (±.010)	4.814 (±.141)	11.93 (±37.04)
R→NEI	.779 (±.009)	4.855 (±.098)	12.20 (±37.67)
NEI→R	.780 (±.078)	4.894 (±.115)	15.27 (±111.2)
NEI→S	.821 (±.008)	4.800 (±.085)	13.42 (±82.30)

Table 3.1: Universal Adversarial Trigger method performance. Triggers are generated given claims from a source class to fool the classifier to predict a target class (column *Class*, with SUPPORTS (S), REFUTES (R), NEI). The results are averaged over the top 10 triggers.

label. The latter is ensured by passing both evidence and claim through an external NLI model trained on SNLI [37].

3.4 Results

We present results for universal adversarial trigger generation and coherent claim generation. Results are measured using the original FC model on claims with added triggers and generated claims (macro F1). We also measure how well the added triggers maintain the claim's original label (semantic similarity score), the perplexity (PPL) of the claims with prepended triggers, and the semantic quality of generated claims (manual annotation). PPL is measured with a pretrained RoBERTa LM.

3.4.1 Adversarial Triggers

Table 3.1 presents the results of applying universal adversarial triggers to claims from the source class. The top-performing triggers for each direction are found in §B.2. The adversarial method with a single FC objective successfully deteriorates model performance by a margin of 0.264 F1 score overall. The biggest performance decrease is when the adversarial triggers are constructed to flip the predicted class from SUPPORTS to REFUTES. We also find that 8 out of 18 triggers from the top-3 triggers for each direction, are negation words such as ‘nothing’, ‘nobody’, ‘neither’, ‘nowhere’ (see Table B.1 in the appendix). The first of these triggers decreases the performance of the model to 0.014 in F1. While this is a significant performance drop, these triggers also flip the meaning of the text. The latter is again indicated by the decrease of the semantic similarity between the claim before and after prepending a trigger token, which is the largest for the SUPPORTS to REFUTES direction. We hypothesise that the success of the best performing triggers is partly due to the meaning of the text being flipped.

Including the auxiliary STS objective increases the similarity between the claim before and after prepending the trigger for five out of six directions. Moreover, we find that now only one out of the 18 top-3 triggers for each direction are negation words. Intuitively, these adversarial triggers are worse at fooling the FC model as they also have to preserve the label of the original claim. Notably, for the SUPPORTS to REFUTES direction the trigger performance is decreased with a margin of 0.642 compared to the single FC objective. We conclude that including the STS objective for generating Universal Adversarial triggers helps to preserve semantic similarity with the original claim, but also makes it harder to both find triggers preserving the label of the claim while substantially decreasing the performance of the model.

3.4.2 Generation

We use the method described in §3.3.3 to generate 156 claims using triggers found with the additional STS objective, and 156 claims without. 52 claims are generated for each class (26 flipping to one class, 26 flipping to the other). A different GPT-2 model is trained to generate claims for each specific class, with triggers specific to attacking that class used as input. The generated claims are annotated manually (see §B.3.3 for the procedure). The overall average claim quality is 4.48, indicating that most generated statements are highly semantically coherent. The macro F1 of the generative model w.r.t. the intended label is 58.9 overall. For the model without the STS objective, the macro F1 is 56.6, and for the model with the STS objective, it is 60.7, meaning that using triggers found with the STS objective helps the generated claims to retain their intended label.

We measure the performance of the original FC model on generated claims (Table 3.3). We compare between using triggers that are generated with the STS objective (Ex2) and without (Ex1). In both cases, the adversarial claims effectively fool the FC model, which performs 38.4% worse and 23.8% worse on Ex1 and Ex2, respectively. Additionally, the overall sentence quality increases when the triggers are found with the STS objective (Ex2). The FC model’s performance is higher on claims using triggers generated with the STS objective but still significantly worse than on the original claims. We provide examples of generated claims with their evidence in

Evidence	Triggers	Generated Claim
SUPPORTS Claims		
Since the 19th century, some Romani have also migrated to the Americas.	don,already,more,during,home	Romani have moved to the Americas during the 19th century.
Cyprus is a major tourist destination in the Mediterranean.	foreign,biggest,major,every, friends	Cyprus is a major tourist destination.
The first Nobel Prize in Chemistry was awarded in 1901 to Jacobus Henricus van't Hoff, of the Netherlands, "for his discovery of the laws of chemical dynamics and osmotic pressure in solutions."	later,already,quite,altern, whereas	Henricus Van't Hoff was already awarded the Nobel Prize.
REFUTES Claims		
California Attorney General Kamala Harris defeated Sanchez , 61.6% to 38.4%.	phys,incarn,not,occasionally, something	Kamala Harris did not defeat Sanchez, 61.6% to 38.4%.
Uganda is in the African Great Lakes region.	unless,endorsed,picks,pref, against	Uganda is against the African Great Lakes region.
Times Higher Education World University Rankings is an annual publication of university rankings by Times Higher Education (THE) magazine.	interested,reward,visit, consumer,conclusion	Times Higher Education World University Rankings is a consumer magazine.
NOT ENOUGH INFO Claims		
The KGB was a military service and was governed by army laws and regulations, similar to the Soviet Army or MVD Internal Troops.	nowhere,only,none,no,nothing	The KGB was only controlled by a military service.
The series revolves around Frank Castle, who uses lethal methods to fight crime as the vigilante "the Punisher", with Jon Bernthal reprising the role from Daredevil.	says,said,take,say,is	Take Me High is about Frank Castle's use of lethal techniques to fight crime.
The Suite Life of Zack & Cody is an American sitcom created by Danny Kallis and Jim Geoghan.	whilst,interest,applic,someone, nevertheless	The Suite Life of Zack & Cody was created by someone who never had the chance to work in television.

Table 3.2: Examples of generated adversarial claims. These are all claims which the FC model incorrectly classified.

Table 3.2.

Comparing FC performance with our generated claims vs. those from the development set of adversarial claims from the FEVER shared task , we see similar drops in performance (0.600 and 0.644 macro F1, respectively). While the adversarial triggers from FEVER cause a larger performance drop, they were manually selected to meet the label coherence and grammatical correctness requirements. Conversely, we automatically generate claims that meet these requirements.

3.5 Conclusion

We present a method for automatically generating highly potent, well-formed, label cohesive claims for FC. We improve upon previous work on universal adversarial triggers by determining how to construct valid claims containing a trigger word. Our method is fully automatic, whereas previous work on generating claims for fact checking is generally rule-based or requires manual intervention. As FC is only one test bed for adversarial attacks, it would be interesting to test

Target	F1	Avg Quality	# Examples
FC Objective			
Overall	0.534	4.33	156
SUPPORTS	0.486	4.79	39
REFUTES	0.494	4.70	32
NEI	0.621	3.98	85
FC+STS Objectives			
Overall	0.635	4.63	156
SUPPORTS	0.617	4.77	67
REFUTES	0.642	4.68	28
NEI	0.647	4.44	61

Table 3.3: FC performance for generated claims.

this method on other NLP tasks requiring semantic understanding such as question answering to better understand shortcomings of models.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 801199.

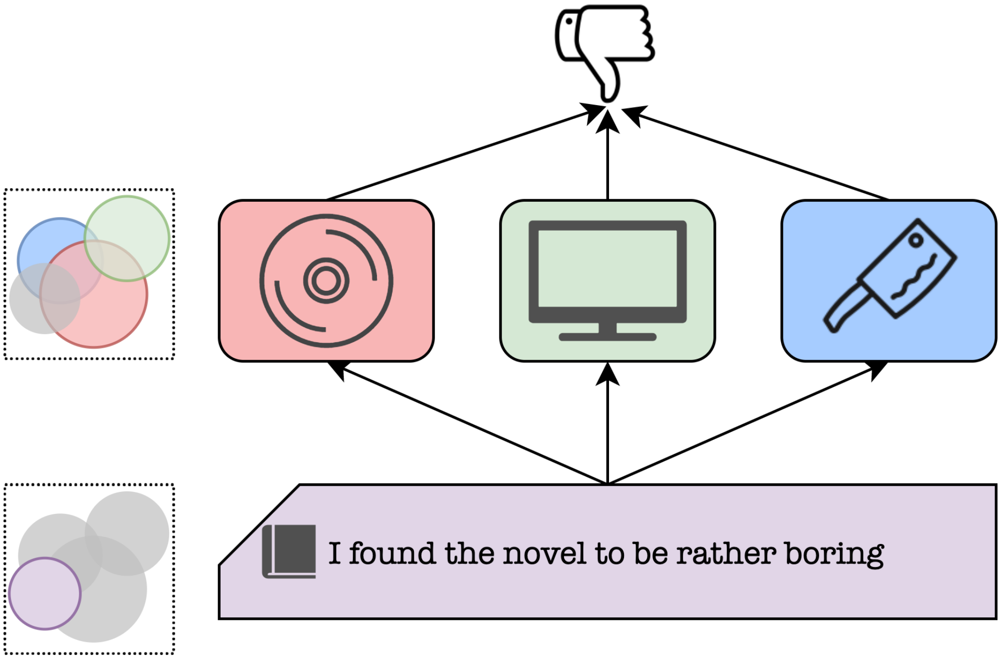


Figure 4.1: In multi-source domain adaptation, a model is trained on data drawn from multiple parts of the underlying distribution. At test time, the model must make predictions on data from a potentially non-overlapping part of the distribution.

4 Transformer Based Multi-Source Domain Adaptation

4.1 Introduction

Machine learning practitioners are often faced with the problem of evolving test data, leading to mismatches in training and test set distributions. As such, the problem of *domain adaptation* is of particular interest to the natural language processing community in order to build models which are robust this shift in distribution. For example, a model may be trained to predict the sentiment of product reviews for DVDs, electronics, and kitchen goods, and must utilize this learned knowledge to predict the sentiment of a review about a book (Figure 4.1). This paper is concerned with this setting, namely *unsupervised multi-source domain adaptation*.

Multi-source domain adaptation is a well studied problem in deep learning for natural language processing. Prominent techniques are generally based on data selection strategies and representation learning. For example, a popular representation learning method is to induce domain invariant representations using unsupervised target data and domain adversarial learning [80]. Adding to this, mixture of experts techniques attempt to learn both domain specific and global shared representations and combine their predictions [92, 142, 153]. These methods have been primarily studied using convolutional nets (CNNs) and recurrent nets (RNNs) trained from scratch, while the NLP community has recently begun to rely more and more on large pretrained transformer (LPX) models e.g. BERT [62]. To date there has been some preliminary investigation of how LPX models perform under domain shift in the single source-single target setting [153, 96, 203, 94]. What is lacking is a study into the effects of and best ways to apply classic multi-source domain adaptation techniques with LPX models, which can give insight into possible avenues for improved application of these models in settings where there is domain shift.

Given this, we present a study into unsupervised multi-source domain adaptation techniques for large pretrained transformer models. Our main research question is: do mixture of experts and domain adversarial training offer any benefit when using LPX models? The answer to this is not immediately obvious, as such models have been shown to generalize quite well across domains and tasks while still learning representations which are not domain invariant. Therefore, we experiment with four mixture of experts models, including one novel technique based on attending to different domain experts; as well as domain adversarial training with gradient reversal. Surprisingly, we find that, while domain adversarial training helps the model learn more domain invariant representations, this does not always result in increased target task performance. When using mixture of experts, we see significant gains on out of domain rumour detection, and some gains on out of domain sentiment analysis. Further analysis reveals that the classifiers learned by domain expert models are highly homogeneous, making it challenging to learn a better mixing function than simple averaging.

4.2 Related Work

Our primary focus is multi-source domain adaptation with LPX models. We first review domain adaptation in general, followed by studies into domain adaptation with LPX models.

4.2.1 Domain Adaptation

Domain adaptation approaches generally fall into three categories: *supervised* approaches (e.g. [56, 76, 134]), where both labels for the source and the target domain are available; *semi-supervised* approaches (e.g. [66, 261]), where labels for the source and a small set of labels for the target domain are provided; and lastly *unsupervised* approaches (e.g. [31, 80, 225, 146]), where only labels for the source domain are given. Since the focus of this paper is the latter, we restrict our discussion to unsupervised approaches. A more complete recent review of unsupervised domain adaptation approaches is given in [129].

A popular approach to unsupervised domain adaptation is to induce representations which are invariant to the shift in distribution between source and target data. For deep networks, this can be accomplished via domain adversarial training using a simple gradient reversal trick [80]. This has been shown to work in the multi-source domain adaptation setting too [142]. Other popular representation learning methods include minimizing the covariance between source and target features [225] and using maximum-mean discrepancy between the marginal distribution of source and target features as an adversarial objective [92].

Mixture of experts has also been shown to be effective for multi-source domain adaptation. [124] use attention to combine the predictions of domain experts. [92] propose learning a mixture of experts using a point to set metric, which combines the posteriors of models trained on individual domains. Our work attempts to build on this to study how multi-source domain adaptation can be improved with LPX models.

4.2.2 Transformer Based Domain Adaptation

There are a handful of studies which investigate how LPX models can be improved in the presence of domain shift. These methods tend to focus on the data and training objectives for single-source single-target unsupervised domain adaptation. The work of [153] shows that curriculum learning based on the similarity of target data to source data improves the performance of BERT on out of domain natural language inference. Additionally, [96] demonstrate that domain adaptive fine-tuning with the masked language modeling objective of BERT leads to improved performance on domain adaptation for sequence labelling. [203] offer similar evidence for task adaptive fine-tuning on aspect based sentiment analysis. [94] take this further, showing that significant gains in performance are yielded when progressively fine-tuning on in domain data, followed by task data, using the masked language modeling objective of RobERTa. Finally, [144] explore whether domain adversarial training with BERT would improve performance for clinical negation detection, finding that the best performing method is a plain BERT model, giving some evidence that perhaps well-studied domain adaptation methods may not be applicable to LPX models.

What has not been studied, to the best of our knowledge, is the impact of domain adversarial training via gradient reversal on LPX models on natural language processing tasks, as well as if mixture of experts techniques can be beneficial. As these methods have historically benefited deep models for domain adaptation, we explore their effect when applied to LPX models in this work.

4.3 Methods

This work is motivated by previous research on domain adversarial training and mixture of domain experts for domain adaptation. In this, the data consists of K source domains $\mathcal{S}$ and a target domain $\mathcal{T}$. The source domains consist of labelled datasets $D_s, s \in \{1, ..., K\}$ and the target domain consists only of unlabelled data U_t. The goal is to learn a classifier f, which generalizes well to $\mathcal{T}$ using only the labelled data from $\mathcal{S}$ and optionally unlabelled data from $\mathcal{T}$. We consider a base network $f_z, z \in \mathcal{S} \cup \{g\}$ corresponding to either a domain specific network

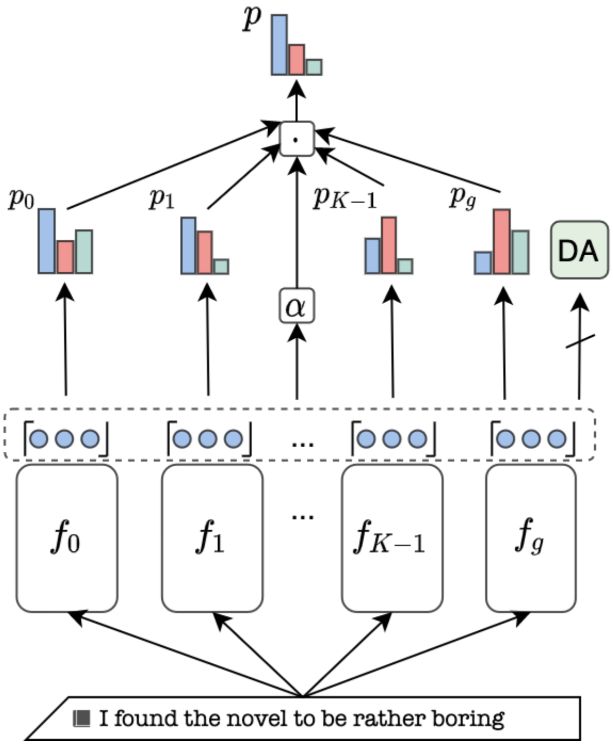


Figure 4.2: The overall approach tested in this work. A sample is input to a set of expert and one shared LPX model as described in §4.3.1. The output probabilities of these models are then combined using an attention parameter alpha (§4.3.1.1, §4.3.1.2, §4.3.1.3, §4.3.1.4). In addition, a global model f_g learns domain invariant representations via a classifier DA with gradient reversal (indicated by the slash, see §4.3.2).

or a global shared network. These f_z networks are initialized using LPX models, in particular DistilBert [211].

4.3.1 Mixture of Experts Techniques

We study four different mixture of expert techniques: simple averaging, fine-tuned averaging, attention with a domain classifier, and a novel sample-wise attention mechanism based on transformer attention [237]. Prior work reports that utilizing mixtures of domain experts and shared classifiers leads to improved performance when having access to multiple source domains [92, 142]. Given this, we investigate if mixture of experts can have any benefit when using LPX models.

Formally, for a setting with K domains, we have set of K different LPX models $f_k, k \in \{0...K-1\}$ corresponding to each domain. There is also an additional LPX model f_g corresponding to a global shared model. The output predictions of these models are $p_k, k \in \{0...K-1\}$ and p_g, respectively. Since the problems we are concerned with are binary classification, these are single values in the range $(0, 1)$. The final output probability is calculated as a weighted combination of a set of domain expert probabilities $\bar{\mathcal{K}} \subseteq \mathcal{S}$ and the probability from the global shared model. Four methods are used for calculating the weighting.

4.3.1.1 Averaging

The first method is a simple averaging of the predictions of domain specific and shared classifiers. The final output of the model is

$$p_A(x, \bar{\mathcal{K}}) = \frac{1}{|\bar{\mathcal{K}}|+1} \sum_{k \in \bar{\mathcal{K}}} p_k(x) + p_g(x) \tag{4.1}$$

4.3.1.2 Fine Tuned Averaging

As an extension to simple averaging, we fine tune the weight given to each of the domain experts and global shared model. This is performed via randomized grid search evaluated on validation data, after the models have been trained. A random integer between zero and ten is generated for each of the models, which is then normalized to a set of probabilities α_F. The final output probability is then given as follows.

$$p_F(x) = \sum_{k \in \bar{\mathcal{K}}} p_k(x) * \alpha_F^{(k)}(x) + p_g(x) * \alpha_F^{(g)}(x) \tag{4.2}$$

4.3.1.3 Domain Classifier

It was recently shown that curriculum learning using a domain classifier can lead to improved performance for single-source domain adaptation [153] when using LPX models. Inspired by this, we experiment with using a domain classifier as a way to attend to the predictions of domain expert models. First, a domain classifier f_C is trained to predict the domain of an input sample x given $\mathbf{r}_g \in \mathbb{R}^d$, the representation of the `[CLS]` token at the output of a LPX model. From the classifier, a vector α_C is produced with the probabilities that a sample belongs to each source domain.

$$\alpha_C = f_C(x) = \text{softmax}(\mathbf{W}_C \mathbf{r}_g + b_C) \tag{4.3}$$

where $\mathbf{W}_C \in \mathbb{R}^{d \times K}$ and $b_C \in \mathbb{R}^K$. The domain classifier is trained before the end-task network and is held static throughout training on the end-task. For this, a set of domain experts f_k are trained and their predictions combined through a weighted sum of the attention vector α_C.

$$p_C(x) = \sum_{k \in S} p_k(x) * \alpha_C^{(k)}(x) \tag{4.4}$$

where the superscript (k) indexes into the α_C vector. Note that in this case we only use domain experts and not a global shared model. In addition, the probability is always calculated with respect to each source domain.

4.3.1.4 Attention Model

Finally, a novel parameterized attention model is learned which attends to different domains based on the input sample. The attention method is based on the scaled dot product attention applied in transformer models [237], where a global shared model acts as a query network attending to each of the expert and shared models. As such, a shared model f_g produces a vector $\mathbf{r}_g \in \mathbb{R}^d$, and each domain expert produces a vector $\mathbf{r}_k \in \mathbb{R}^d$. First, for an input sample x, a probability for the end task is obtained from the classifier of each model yielding probabilities p_g and $p_k, k \in 0...K-1$. An attention vector α_X is then obtained via the following transformations.

$$\mathbf{q} = \mathbf{g}\mathbf{Q}^T \tag{4.5}$$

$$\mathbf{k} = \begin{bmatrix} \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_K \\ \mathbf{r}_g \end{bmatrix} \mathbf{K}^T \tag{4.6}$$

$$\alpha_X = \text{softmax}(\mathbf{q}\mathbf{k}^T) \tag{4.7}$$

where $\mathbf{Q} \in \mathbb{R}^{d\times d}$ and $\mathbf{K} \in \mathbb{R}^{d\times d}$. The attention vector α_X then attends to the individual predictions of each domain expert and the global shared model.

$$p_X(x, \bar{\mathcal{K}}) = \sum_{k\in\bar{\mathcal{K}}} p_k(x) * \alpha_X^{(k)}(x) + p_g(x) * \alpha_X^{(g)}(x) \tag{4.8}$$

To ensure that each model is trained as a domain specific expert, a similar training procedure to that of [92] is utilized, described in §4.3.3.

4.3.2 Domain Adversarial Training

The method of domain adversarial adaptation we investigate here is the well-studied technique described in [80]. It has been shown to benefit both convolutional nets and recurrent nets on NLP problems [142, 91], so is a prime candidate to study in the context of LPX models. Additionally, some preliminary evidence indicates that adversarial training might improve LPX generalizability for single-source domain adaptation [153].

To learn domain invariant representations, we train a model such that the learned representations maximally confuse a domain classifier f_d. This is accomplished through a min-max objective between the domain classifier parameters θ_D and the parameters θ_G of an encoder f_g. The objective can then be described as follows.

$$\mathcal{L}_D = \max_{\theta_D} \min_{\theta_G} -d \log f_d(f_g(x)) \tag{4.9}$$

where d is the domain of input sample x. The effect of this is to improve the ability of the classifier to determine the domain of an instance, while encouraging the model to generate maximally confusing representations via minimizing the negative loss. In practice, this is accomplished by training the model using standard cross entropy loss, but reversing the gradients of the loss with respect to the model parameters θ_G.

4.3.3 Training

Our training procedure follows a multi-task learning setup in which the data from a single batch comes from a single domain. Domains are thus shuffled on each round of training and the model is optimized for a particular domain on each batch.

For the attention based (§4.3.1.4) and averaging (§4.3.1.1) models we adopt a similar training algorithm to [92]. For each batch of training, a meta-target t is selected from among the source domains, with the rest of the domains treated as meta-sources $\mathcal{S}' \in \mathcal{S} \setminus \{t\}$. Two losses are then calculated. The first is with respect to all of the meta-sources, where the attention vector is calculated for only those domains. For target labels y_i and a batch of size N with samples from a single domain, this is given as follows.

$$\mathcal{L}_s = -\frac{1}{N} \sum_i y_i \log p_X(x, \mathcal{S}') \tag{4.10}$$

The same procedure is followed for the averaging model p_A. The purpose is to encourage the model to learn attention vectors for out of domain data, thus why the meta-target is excluded from the calculation.

The second loss is with respect to the meta-target, where the cross-entropy loss is calculated directly for the domain expert network of the meta-target.

$$\mathcal{L}_t = -\frac{1}{N} \sum_i y_i \log p_t(x) \tag{4.11}$$

This allows each model to become a domain expert through strong supervision. The final loss of the network is a combination of the three losses described previously, with λ and γ hyperparameters controlling the weight of each loss.

$$\mathcal{L} = \lambda \mathcal{L}_s + (1 - \lambda)\mathcal{L}_t + \gamma \mathcal{L}_D \tag{4.12}$$

For the domain classifier (§4.3.1.3) and fine-tuned averaging (§4.3.1.2), the individual LPX models are optimized directly with no auxiliary mixture of experts objective. In addition, we experiment with training the simple averaging model directly.

4.4 Experiments and Results

We focus our experiments on text classification problems with data from multiple domains. To this end, we experiment with sentiment analysis from Amazon product reviews and rumour detection from tweets. For both tasks, we perform cross-validation on each domain, holding out a single domain for testing and training on the remaining domains, allowing a comparison of each method on how well they perform under domain shift. The code to reproduce all of the experiments in this paper can be found here[6].

Sentiment Analysis Data The data used for sentiment analysis come from the legacy Amazon Product Review dataset [30]. This dataset consists of 8,000 total tweets from four product categories: books, DVDs, electronics, and kitchen and housewares. Each domain contains 1,000 positive and 1,000 negative reviews. In addition, each domain has associated unlabelled data. Following previous work we focus on the transductive setting [92, 272] where we use the same 2,000 out of domain tweets as unlabelled data for training the domain adversarial models. This data has been well studied in the context of domain adaptation, making for easy comparison with previous work.

Rumour Detection Data The data used for rumour detection come from the PHEME dataset of rumourous tweets [277]. There are a total of 5,802 annotated tweets from 5 different events labelled as rumourous or non-rumourous (1,972 rumours, 3,830 non-rumours). Methods which have been shown to work well on this data include context-aware classifiers [276] and positive-unlabelled learning [254]. Again, we use this data in the transductive setting when testing domain adversarial training.

4.4.1 Baselines

What's in a Domain? We use the model from [142] as a baseline for sentiment analysis. This model consists of a set of domain experts and one general CNN, and is trained with a domain adversarial auxiliary objective.

Mixture of Experts Additionally, we present the results from [92] representing the most recent state of the art on the Amazon reviews dataset. Their method consists of domain expert

[6] `https://github.com/copenlu/xformer-multi-source-domain-adaptation`

Method	Sentiment Analysis (Accuracy)					Rumour Detection (F1)					
	D	E	K	B	macroA	CH	F	GW	OS	S	μF1
[142]	77.9	80.9	80.9	77.1	79.2	-	-	-	-	-	-
[92]	87.7	89.5	90.5	87.9	88.9	-	-	-	-	-	-
[276]	-	-	-	-	-	63.6	**46.5**	70.4	69.0	61.2	60.7
Basic	89.1	89.8	90.1	89.3	89.5	66.1	44.7	71.9	61.0	63.3	62.3
Adv-6	88.3	89.7	90.0	89.0	89.3	65.8	42.0	66.6	61.7	63.2	61.4
Adv-3	89.0	89.9	90.3	89.0	89.6	65.7	43.2	72.3	60.4	62.1	61.7
Independent-Avg	88.9	**90.6**	90.4	**90.0**	**90.0**	66.1	45.6	71.7	59.4	63.5	62.2
Independent-Ft	88.9	90.3	**90.8**	**90.0**	**90.0**	65.9	45.7	72.2	59.3	62.4	61.9
MoE-Avg	**89.3**	89.9	90.5	89.9	89.9	**67.9**	45.4	**74.5**	62.6	**64.7**	**64.1**
MoE-Att	88.6	90.0	90.4	89.6	89.6	65.9	42.3	72.5	61.2	63.3	62.2
MoE-Att-Adv-6	87.8	89.0	90.5	88.3	88.9	66.0	40.7	69.0	63.8	63.7	61.8
MoE-Att-Adv-3	88.6	89.1	90.4	88.9	89.2	65.6	42.7	73.4	60.9	61.0	61.8
MoE-DC	87.8	89.2	90.2	87.9	88.8	66.5	40.6	70.5	**70.8**	62.8	63.8

Table 4.1: Experiments for sentiment analysis in (D)VD, (E)lectronics, (K)itchen and housewares, and (B)ooks domains and rumour detection for different events ((C)harlie(H)ebdo, (F)erguson, (G)erman(W)ings, (O)ttawa(S)hooting, and (S)ydneySiege) using leave-one-out cross validation. Results are averaged across 5 random seeds. The results for sentiments analysis are in terms of accuracy and the results for rumour detection are in terms of F1.

classifiers trained on top of a shared encoder, with predictions being combined via a novel metric which considers the distance between the mean representations of target data and source data.

[276] Though not a domain adaptation technique, we include the results from [276] on rumour detection to show the current state of the art performance on this task. The model is a CRF, which utilizes a combination of content and social features acting on a timeline of tweets.

4.4.2 Model Variants

A variety of models are tested in this work. Therefore, each model is referred to by the following.

Basic Basic DistilBert with a single classification layer at the output.

Adv-X DistilBert with domain adversarial supervision applied at the X'th layer (§4.3.2).

Independent-Avg DistilBert mixture of experts averaged but trained individually (not with the algorithm described in §4.3.3).

Independent-FT DistilBert mixture of experts averaged with mixing attention fine tuned after training (§4.3.1.2), trained individually.

MoE-Avg DistilBert mixture of experts using averaging (§4.3.1.1).

MoE-Att DistilBert mixture of experts using our novel attention based technique (§4.3.1.4).

MoE-Att-Adv-X DistilBert mixture of experts using attention and domain adversarial supervision applied at the X'th layer.

MoE-DC DistilBert mixture of experts using a domain classifier for attention (§4.3.1.3).

4.4.3 Results

Our results are given in Table 4.1. Similar to the findings of [144] on clinical negation, we see little overall difference in performance from both the individual model and the mixture of experts model when using domain adversarial training on sentiment analysis. For the base model, there is a slight improvement when domain adversarial supervision is applied at a lower layer of the model, but a drop when applied at a higher level. Additionally, mixture of experts provides some benefit, especially using the simpler methods such as averaging.

For rumour detection, again we see little performance change from using domain adversarial training, with a slight drop when supervision is applied at either layer. The mixture of experts methods overall perform better than single model methods, suggesting that mixing domain experts is still effective when using large pretrained transformer models. In this case, the best mixture of experts methods are simple averaging and static grid search for mixing weights, indicating the difficulty in learning an effective way to mix the predictions of domain experts. We elaborate on our findings further in §4.5. Additional experiments on domain adversarial training using Bert can be found in Table C.1 in §C.1, where we similarly find that domain adversarial training leads to a drop in performance on both datasets.

4.5 Discussion

We now discuss our initial research questions in light of the results we obtained, and provide explanations for the observed behavior.

4.5.1 What is the Effect of Domain Adversarial Training?

We present PCA plots of the representations learned by different models in Figure 4.3. These are the final layer representations of 500 randomly sampled points for each split of the data. In the ideal case, the representations for out of domain samples would be indistinguishable from the representations for in domain data.

In the case of basic DistilBert, we see a slight change in the learned representations of the domain adversarial models versus the basic model (Figure 4.3 top half, a-c) for sentiment analysis. When the attention based mixture of experts model is used, the representations of

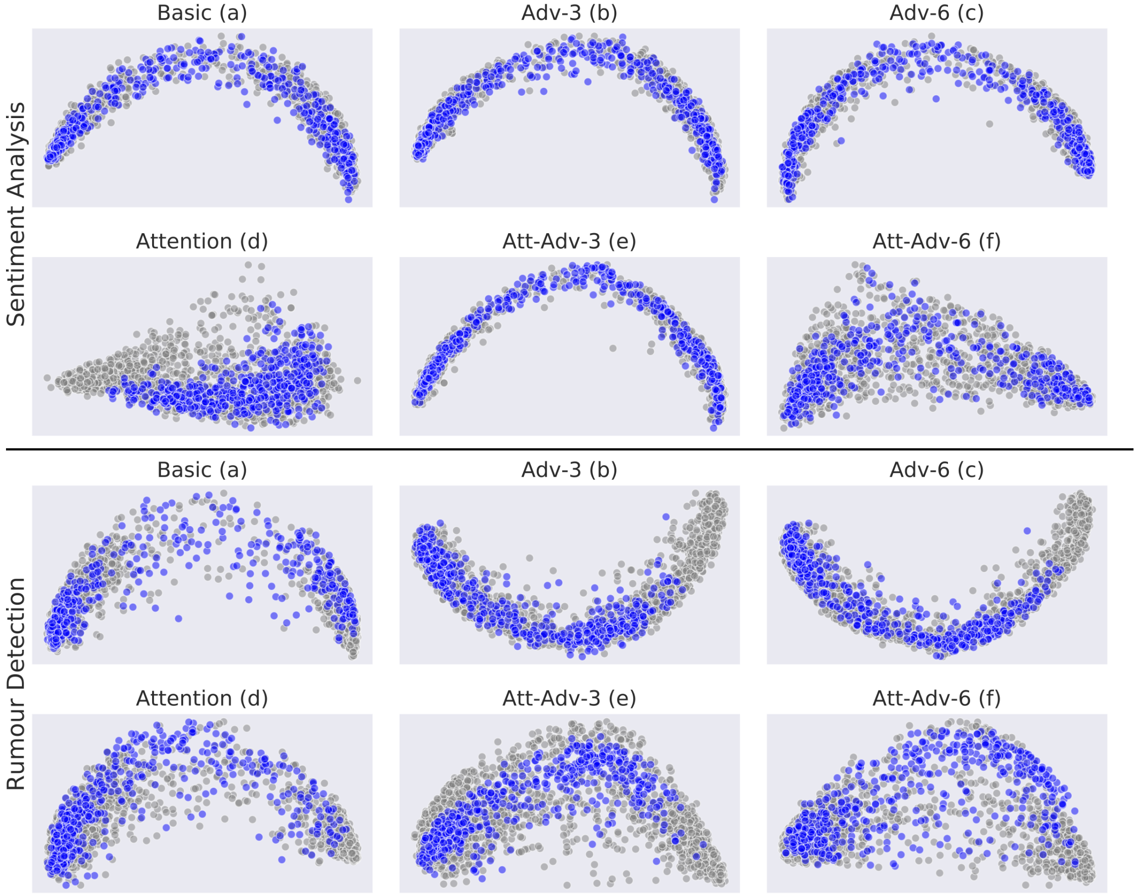


Figure 4.3: Final layer DistilBert embeddings for 500 randomly selected examples from each split for each tested model for sentiment data (top two rows) and rumour detection (bottom two rows). The blue points are out of domain data (in this case from Kitchen and Housewares for sentiment analysis and Sydney Siege for rumour detection) and the gray points are in domain data.

out of domain data cluster in one region of the representation space (d). With the application of adversarial supervision, the model learns representations which are more domain agnostic. Supervision applied at layer 6 of DistilBert (plot f) yields a representation space similar to the version without domain adversarial supervision. Interestingly, the representation space of the model with supervision at layer 3 (plot e) yields representations similar to the basic classifier. This gives some potential explanation as to the similar performance of this model to the basic classifier on this split (kitchen and housewares). Overall, domain adversarial supervision has some effect on performance, leading to gains in both the basic classifier and the mixture of experts model for this split. Additionally, there are minor improvements overall for the basic case, and a minor drop in performance with the mixture of experts model.

The effect of domain adversarial training is more pronounced on the rumour detection data for the basic model (Figure 4.3 bottom half, a), where the representations exhibit somewhat

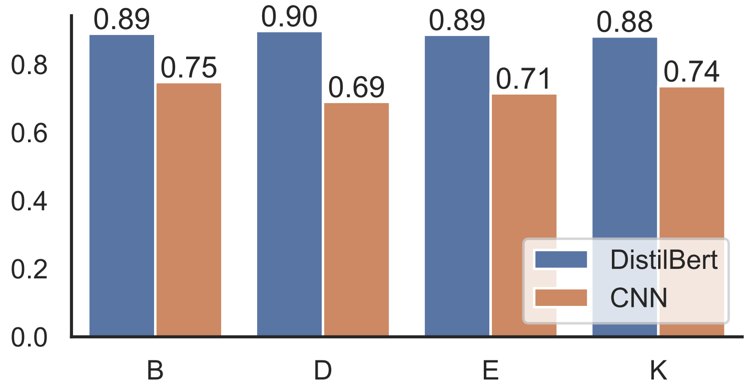


Figure 4.4: Comparison of agreement (Krippendorff's alpha) between domain expert models when the models are either DistilBert or a CNN. Predictions are made on unseen test data by each domain expert, and agreement is measured between their predictions ((B)ooks, (D)VD, (E)lectronics, and (K)itchen). The overall agreement between the DistilBert experts is greater than the CNNs, suggesting that the learned classifiers are much more homogenous.

less variance when domain adversarial supervision is applied. Surprisingly, this leads to a slight drop in performance for the split of the data depicted here (Sydney Siege). For the attention based model, the variant without domain adversarial supervision (d) already learns a somewhat domain agnostic representation. The model with domain adversarial supervision at layer 6 (f) furthers this, and the classifier learned from these representations perform better on this split of the data. Ultimately, the best performing models for rumour detection do not use domain supervision, and the effect on performance on the individual splits are mixed, suggesting that domain adversarial supervision can potentially help, but not in all cases.

4.5.2 Is Mixture of Experts Useful with LPX Models?

We performed experiments with several variants of mixture of experts, finding that overall, it can help, but determining the optimal way to mix LPX domain experts remains challenging. Simple averaging of domain experts (§4.3.1.1) gives better performance on both sentiment analysis and rumour detection over the single model baseline. Learned attention (§4.3.1.4) has a net positive effect on performance for sentiment analysis and a negative effect for rumour detection compared to the single model baseline. Additionally, simple averaging of domain experts consistently outperforms a learned sample by sample attention. This highlights the difficulty in utilizing large pretrained transformer models to learn to attend to the predictions of domain experts.

Comparing agreement To provide some potential explanation for why it is difficult to learn to attend to domain experts, we compare the agreement on the predictions of domain experts of one of our models based on DistilBert, versus a model based on CNNs (Figure 4.4). CNN models are chosen in order to compare the agreement using our approach with an approach which has been shown to work well with mixture of experts on this data [92]. Each CNN consists of an embedding layer initialized with 300 dimensional FastText embeddings [35], a series of 100 dimensional convolutional layers with widths 2, 4, and 5, and a classifier. The end performance is on par with previous work using CNNs [142] (78.8 macro averaged accuracy, validation accuracies of the individual models are between 80.0 and 87.0). Agreement is measured using Krippendorff's alpha [131] between the predictions of domain experts on test data.

We observe that the agreement between DistilBert domain experts on test data is significantly higher than that of CNN domain experts, indicating that the learned classifiers of each expert are much more similar in the case of DistilBert. Therefore, it will potentially be more difficult for a mixing function on top of DistilBert domain experts to gain much beyond simple averaging, while with CNN domain experts, there is more to be gained from mixing their predictions. This effect may arise because each DistilBert model is highly pre-trained already, hence there is little change in the final representations, and therefore similar classifiers are learned between each domain expert.

4.6 Conclusion

In this work, we investigated the problem of multi-source domain adaptation with large pretrained transformer models. Both domain adversarial training and mixture of experts techniques were explored. While domain adversarial training could effectively induce more domain agnostic representations, it had a mixed effect on model performance. Additionally, we demonstrated that while techniques for mixing domain experts can lead to improved performance for both sentiment analysis and rumour detection, determining a beneficial mixing of such experts is challenging. The best method we tested was a simple averaging of the domain experts, and we provided some evidence as to why this effect was observed. We find that LPX models may be better suited for data-driven techniques such as that of [94], which focus on inducing a better prior into the model through pretraining, as opposed to techniques which focus on learning a better posterior with architectural enhancements. We hope that this work can help inform researchers of considerations to make when using LPX models in the presence of domain shift.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 801199.

5 Multi-View Knowledge Distillation from Crowd Annotations for Out-of-Domain Generalization

5.1 Introduction

One of the primary concerns in supervised machine learning is how to define, collect, and use labels as training data for a given task. There are a multitude of tradeoffs associated with this decision, including the cost, the number of labels to collect, the time to collect those labels, the accuracy of those labels with respect to the task under consideration, and how well those labels enable model generalization. These tradeoffs are made based on how the labels are collected (e.g. crowdsourcing, expert labeling, distant supervision) and how they are trained on in practice, for example as one-hot categorical labels (hard labeling) or as a distribution over possible classes (soft labeling).

A large body of literature exists which examines all facets of this question [235]. Recent work has focused on utilizing soft-labeling schemes for classification tasks as a method for improving both model accuracy and uncertainty estimation [183, 234, 78]. When using soft-labels, models are trained to minimize the divergence between their predictive distribution and a distribution over the labels obtained from crowd annotations [234]. While this has been shown to potentially improve model generalization for vision tasks [183], little work has systematically compared how different soft-labeling schemes affect out-of-distribution performance and uncertainty estimation in NLP. We seek to fill this gap in this work, providing an in-depth study into soft-labeling techniques and best practices for improving model generalization and uncertainty estimation across eight methods, 4 NLP tasks, and and 7 datasets.

Soft-labeling methods have been compared in both [78] and [235] for an in-domain testing setting. These studies are primarily focused on identifying the best methods for learning from these soft distributions within a particular domain without going in great depth about which methods for obtaining soft-labels lead to best performance. As such, no clear best method emerges when comparing across soft-labeling approaches in the in-domain setting, making it difficult to decide which aggregation technique to use for a given task. Additionally, these studies do not examine the out of domain test setting, where the benefits of soft-labeling have been made clear in the computer vision literature [183]. Here, we demonstrate that soft-labels which are aggregated across multiple soft-labeling techniques mitigate changes in performance for these different techniques across different tasks for out of domain data. To do this, we propose four multi-view aggregation methods to generate aggregated soft-labels, including three novel methods based on the Jensen-Shannon centroid and temperature scaling.

In sum, this work makes the following contributions:

1. A comprehensive comparison of soft-labeling techniques for learning from crowd anno-

tations for 4 NLP tasks across 7 datasets in the out-of-domain test setting, including text classification (recognizing textual entailment, medical relation extraction, and toxicity detection) and sequence tagging (part-of-speech tagging);

2. Novel methods for aggregating different views of soft-labels derived from crowd-annotations;
3. Insights and suggestions into best practices and tradeoffs for different soft-labeling methods in terms of performance and uncertainty estimation.

5.2 Related Work

Learning from Crowd-Sourced Labels An efficient way to collect training data for a new task is to ask crowd annotators on platforms such as Amazon Mechanical Turk to manually annotate training data. How to select an appropriate training signal from these noisy crowd labels has a rich set of literature (e.g. see the survey from [177]). There exist a multitude of studies on selecting the best label from a set of crowd annotations, many focusing on Bayesian methods to learn a latent distribution over the true class for each sample influenced by factors such as annotator behavior [110, 57] and item difficulty [43]. While selecting a single true label to use as training is the dominant paradigm in machine learning, this discards potentially useful information regarding uncertainty over classes inherent in many tasks, for example where items can be especially difficult or ambiguous [88]. Given this, recent work has looked into how to learn directly from crowd-annotations [235]. The work of [183] was one of the first, which demonstrated that learning directly from crowd annotations treated as soft-labels using the softmax function leads to better out of distribution performance in computer vision. This line of work has been followed by [234] and [78] in NLP, looking at the use of the KL divergence as an effective loss on the soft labels. The survey of [235] provides an extensive set of experiments on different methods for learning from crowd labels on a vast array of datasets. What has not been done is a systematic comparison of different soft-labeling methods in the out of domain setting, where we reveal that the selection of the best method is not obvious. We fill this gap in this work, and propose new methods for aggregating soft-labels which yield more consistent and robust performance without requiring new annotations or learning methods.

Knowledge Distillation Knowledge distillation seeks to build compact but robust models by training them on the probability distribution learned by a much larger teacher network [18, 105]. The goal is to impart the “dark knowledge” contained in the distribution learned by the larger network, which can indicate the degree to which a particular example resembles each class if the soft-targets from the classifier are well calibrated. Oftentimes the distribution of the larger network will be smoothed via some temperature in order to accomplish this [105], as has also been done in learning from crowd labels [233], or ensembled with multiple large classifiers [105, 4]. [4] demonstrate that the data used to train an ensemble should constitute a

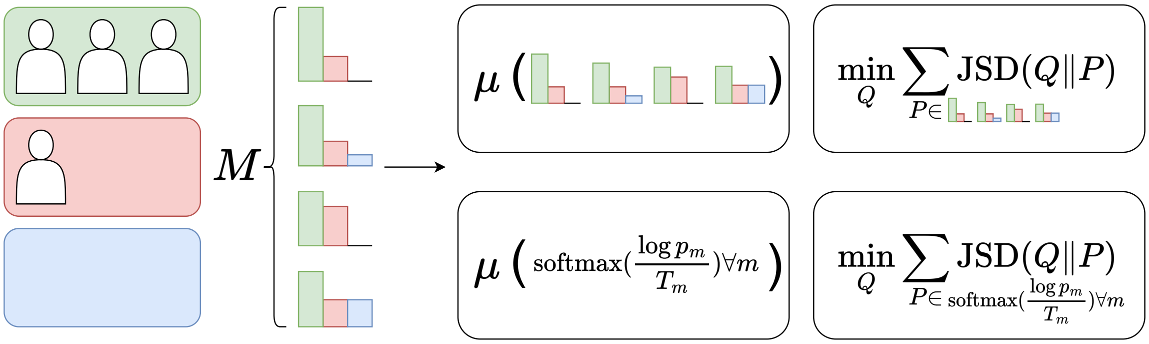

M
$\mu\,($ $)$
$\min_{Q} \sum_{P\in} \mathrm{JSD}(Q\|P)$
$\mu\left(\mathrm{softmax}(\frac{\log p_m}{T_m})\forall m\right)$
$\min_{Q} \sum_{P\in\,\mathrm{softmax}(\frac{\log p_m}{T_m})\forall m} \mathrm{JSD}(Q\|P)$

Figure 5.1: We experiment with four methods for aggregating soft labels in this work: Averaging, the Jensen-Shannon centroid, temperature scaling, and a hybrid approach.

multi-view structure (i.e. multiple different features in the data are predictive of a particular class) in order for ensembling to improve the test set performance of the final distilled model. Inspired by this, we develop several methods for aggregating multiple views of crowd-sourced labels in order to obtain a distribution that can induce robust classifiers in the out-of-domain setting. As such, “multi-view” in this work is defined as multiple posterior distributions over true classes from crowd annotations that are explained by different factors e.g. annotator behavior or raw number of votes per class.

5.3 Methods

We build upon a rich literature around the topic of learning from crowd annotations. As opposed to hard labels, soft-labeling schemes give us a distribution over the possible classes in a given dataset. In this, more difficult or ambiguous classes can have their probability mass distributed over multiple classes, which can help regularize a downstream classifier. This may also reflect potential “dark knowledge” [105], or the relative similarity of an input sample to different classes, learnable from the crowd annotations. Multiple soft-labeling schemes have been demonstrated to provide good training signals on different NLP tasks, but none of these methods are consistently best across tasks [235]. Given this, we start with several well-studied methods for learning from crowd-labels, described in subsubsection 5.3.1 [234, 78, 110, 57], adding to this literature by providing a systematic analysis of their performance and best practices when considering generalization to out of domain data, which is under-explored in NLP. Then, we propose several methods for aggregating the distributions over class labels produced by each of these methods in subsubsection 5.3.2, which we will demonstrate produce robust soft-labels across tasks.

5.3.1 Soft Labeling Methods

We experiment with two widely used normalization schemes for obtaining soft labels, as well as two methods based on Bayesian models of annotation.

Standard Normalization The standard normalization scheme presented in [234] obtains soft-labels for a given sample by transforming a set of crowd-sourced labels into a probability distribution. This is done by normalizing the number of votes given to each label by the total number of annotations for a given sample, as described in Equation 5.1.

$$p_{stand}(i, y) = \frac{c_{i,y}}{\sum_{\hat{y}} c_{i,\hat{y}}} \tag{5.1}$$

where $c_{i,y}$ is the number of votes label y received for item i.

Softmax Normalization The standard normalization scheme does not distribute probability mass to any label which receives no votes from any annotator. The works of [183, 78] propose to use the softmax function directly from label vote counts as a way to obtain soft labels for a given sample, as in Equation 5.2.

$$p_{soft}(i, y) = \frac{e^{c_{i,y}}}{\sum_{\hat{y}} e^{c_{i,\hat{y}}}} \tag{5.2}$$

This can potentially help to further regularize a model.

Dawid & Skene A common method for aggregating crowd-sourced labels into a single ground-truth label is to treat the true label as a latent variable to be learned from annotations. Several models have been proposed in the literature to accomplish this [57, 110, 43], often accounting for other aspects of the annotation problem such as annotator competence and item difficulty. One such method is the Dawid and Skene model [57], a highly popular method across fields for aggregating labels from crowd-annotations, which focuses in particular on modeling the true class based on each annotator's ability to correctly identify true instances of a given class. In other words, the model is designed to explain away inconsistencies of individual annotators, which may be desirable for use as a training signal when gold labels are unavailable. To obtain a soft label for a given sample i from this model, we use the posterior distribution of the latent variable c_i which models the true class for a given instance.[7]

MACE Multi-Annotator Competence Estimation (MACE, [110])[8] is another Bayesian method popular in NLP which focuses specifically on explaining away poor performing annotators. It does this by learning to differentiate between annotators which likely follow the global labeling strategy of selecting the true underlying label from those which follow a labeling strategy which deviates from this e.g. spamming a single label for every example. To do this, it learns a

[7] Implementation: `https://github.com/sukrutrao/Fast-Dawid-Skene`

[8] Implementation: `https://github.com/dirkhovy/MACE`

distribution over the true label for each sample, as well as the likelihood that each annotator is faithfully labeling each sample. For extensive details on both the Dawid and Skene and MACE models, as well as several other Bayesian annotation models, see the survey by [177].

5.3.2 Combining Soft Labels

In order to acquire labels which capture the multiple views of the annotations learned by individual soft-labeling methods, we develop and investigate novel methods for combining the soft-labels obtained from these different views. This is inexpensive, requiring zero additional annotations, and we will demonstrate that it is robust across tasks.

The goal for a single example x_i is as follows: given a set of categorical distributions $p_m(y_i|x_i)$ with $m \in \{1...M\}$ for M different soft-labeling methods, produce a categorical distribution $p(y_i|x_i) = f(p_{1:M}(y_i|x_i))$ which will serve as a soft target for example x_i. The function f should be selected to capture the different aspects of the problem which each distribution p_m models. For example, MACE produces a posterior probability over the true class which explains away bad annotator behavior, while the model of Dawid and Skene produces a posterior which best explains each annotator's individual annotation biases. Our hypothesis is that combining several different models (i.e. different **views** of the crowd-sourced annotations) will yield labels that can induce more robust classifiers.

Averaging The most basic model to acquire an ensembled probability distribution is to take an average of the individual probabilities $p_{1:M}$. More formally, the averaging function f_a takes the following form:

$$f_a(p_{1:M}(y_i|x_i)) = \frac{1}{M}\sum_m p_m(y_i|x_i) \tag{5.3}$$

This effectively yields a distribution which is the center of mass of the given distributions $p_{1:M}$.

Jensen-Shannon Centroid The Jensen-Shannon centroid (JSC) is the minimizer of the sum of the Jensen-Shannon divergences (JSD) between a proposed distribution Q and a set of probability distributions $p_{1:M}$. It is defined as:

$$f_c(p_{1:M}(y_i|x_i)) =_Q \sum_m \text{JS}(p_m\|Q) \tag{5.4}$$

where JS$(P\|Q)$ is the JSD, a symmetric version of the Kullback-Leibler divergence (KLD), defined as follows for discrete probability distributions:

$$\text{JS}(P\|Q) = \frac{1}{2}\text{KLD}(P\|S) + \frac{1}{2}\text{KLD}(Q\|S) \tag{5.5}$$

$$S = \frac{1}{2}(P + Q)$$

$$\mathsf{KLD}(P \| Q) = \sum_j P^{(j)} \log \frac{P^{(j)}}{Q^{(j)}} \tag{5.6}$$

In other words, the JSC is a probability distribution which has the least average total divergence to the average between itself and each probability distribution in a set of probability distributions.

Finding the JSC can be done efficiently using methods from convex optimization. In particular, we use the ConCave-Convex procedure (CCCP, [267]) developed in [171]. The full derivation and definition of the method can be found in [171] Equations 94-104 and Algorithm 1, but at a high level, we can define a categorical distribution p with K classes using the natural parameter θ consisting of $K-1$ components as:

$$p = \{\theta_{1:(K-1)}, 1 - \sum_{k=1}^{K-1} \theta_k\}$$

The negative entropy of this distribution is then calculated in terms of θ as follows:

$$F(\theta) = -H(\theta) = \sum_{k=1}^{K-1} \theta_k \log \theta_k + (1 - \sum_{k=1}^{K-1} \theta_k) \log(1 - \sum_{k=1}^{K-1} \theta_k) \tag{5.7}$$

which has partial derivatives and inverse gradient:

$$\frac{\partial}{\partial \theta_k} = \log \frac{\theta_k}{1 - \sum_{k=1}^{K-1} \theta_k} \tag{5.8}$$

$$\theta_k = (\nabla F^{-1}(\eta))_k = \frac{e^{\eta_k}}{1 + \sum_{k=1}^{K-1} e^{\eta_k}} \tag{5.9}$$

The JSD between two categorical distributions p_1 and p_2 under this view can then be calculated in terms of the negative entropy F defined in Equation 5.7:

$$\mathsf{JS}(\theta_1 \| \theta_2) = \frac{F(\theta_1) + F(\theta_2)}{2} - F(\frac{\theta_1 + \theta_2}{2})$$

Finally, the hyperparameterless update rule used to find the locally optimum JSC of a set of probability distributions $p_{1:M}$ using their natural parameters $\theta_{1:M}$ is defined in terms of Equation 5.8 and Equation 5.9:

$$\theta^{(t+1)} = (\nabla F)^{-1}(\frac{1}{M} \sum_m F(\frac{\theta_m + \theta^{(t)}}{2})) \tag{5.10}$$

where $\theta^{(0)} = [f_a(p_{1:M})]_{1:K-1}$. This optimization procedure is efficient and improves linearly to a local minimum.

Temperature Scaling One approach in knowledge distillation is to scale the softmax output of the larger teacher network prior to using it to produce soft labels to teach the smaller student network. Here, we develop a method for optimizing a temperature parameter for each distribution in our ensemble based on the JSD between each distribution.

For each soft-labeling method p_m, we optimize a temperature parameter $T_m, m \in \{1...M\}$ which softens each distribution produced by that method. The optimization procedure for a given parameter T_m is then given in Equation 5.11 and Equation 5.12

$$\tilde{p}_k(y_i|x_i) = \text{softmax}(\frac{l_k(y_i|x_i)}{T_k}) \tag{5.11}$$

$$\mathcal{L}(T_m) = \frac{1}{M-1}\sum_{k!=m} \text{JSD}(\tilde{p}_m\|\tilde{p}_k) \tag{5.12}$$

where l_k are the log-probabilities for a given sample and JSD is calculated as in Equation 5.5. In practice, since the JSD is symmetric, we only need to calculate the loss for the $\frac{M(M-1)}{2}$ combinations of distributions in the ensemble. Since optimizing this loss directly will encourage the temperature to scale to infinity, as the loss will be 0 when a large enough temperature drives all distributions to be uniform, we add a regularization loss on the temperature parameters in order to discourage them from being exceedingly large. The final loss (assuming averaging the JSD over a batch of samples) is given in Equation 5.13.

$$\mathcal{L} = \frac{1}{Z}\sum_{j=1}^{M}\sum_{k=j+1}^{M} \text{JSD}(\tilde{p}_j\|\tilde{p}_k) + \lambda T_j^2 \tag{5.13}$$

where λ is a regularization constant and $Z = \frac{M(M-1)}{2}$. Finally, after optimizing for the temperature parameters T_m, we aggregate the distributions by averaging over the temperature scaled ensemble.

$$f_t(p_{1:M}) = f_a(\tilde{p}_{1:M}) \tag{5.14}$$

Hybrid Finally, we develop a hybrid approach where we first temperature scale the distributions in the ensemble via Equation 5.13, followed by finding the JSC as in Equation 5.4.

$$f_h(p_{1:M}) = f_c(\tilde{p}_{1:M}) \tag{5.15}$$

5.3.3 Learning From Soft labels

Learning from soft labels requires methods which minimize the divergence between the probability distribution produced by a classifier and the distribution over labels. To do this, we adopt the strategies used in knowledge distillation and learning from crowd-sourced labels, using the

Kullback-Leibler divergence as a loss function between the probability distribution produced by a given classifier and the soft labels produced by the methods in subsubsection 5.3.1 and subsubsection 5.3.2.

5.4 Experimental Setup

We focus our experiments on the following research questions:

- **RQ1**: Which methods for learning from crowd-sourced labels are most robust in out-of-domain settings?
- **RQ2**: Does aggregating multiple views of crowd annotations lead to more robust out-of-domain performance?
- **RQ3**: Which soft-labeling methods lead to better uncertainty estimation?

Our experiments focus on the out-of-domain setting, where there is distribution shift between training data and test data. We use pairs of datasets which capture the same high level tasks such that the training data has both gold and crowd-annotations available while the testing data has only gold annotations. Thus, to perform well on the test set, a model must be able to generalize across these two factors: input data sourced from different corpora and/or labels acquired from different sources. Additionally, two of our experiments employ limited datasets for training, giving a very difficult out of domain setting. We choose this setup in order to understand the impact of crowd-sourced soft targets on model generalization, whereas in the in-domain setting where train and test data use gold labels obtained from the same source, performance is dominated by the use of expert labels. In other words, when does the knowledge contained in soft-targets confer benefits over expert annotations in NLP?

For all experiments we use RoBERTa as our base network [148] with the same training hyperparameters in order to provide a stable comparison across different soft-labeling techniques. Additionally, this allows us to observe how the same soft-labeling techniques on the same network perform on different tasks. For the soft-labeling experiments (labeled “KLD”) we only use soft labels obtained using one of the crowd-labeling methods described in subsubsection 5.3.1 and subsubsection 5.3.2 and trained using the KL divergence as the loss. Additionally, we experiment with the multi-task learning setup used in [78, 235], where the model is trained on both gold labels and soft crowd-sourced labels (labeled “Gold + KLD”). This allows us to differentiate performance between when gold annotations are available vs. not, which is clearly beneficial in the in-domain test setting where the same method of acquiring gold labels is used for test data [78], but not necessarily in the out-of-domain setting [183]. The tasks and datasets used in our experiments are described in the following paragraphs.

Recognizing Textual Entailment (RTE) The first task we consider is recognizing textual entailment (RTE). In the RTE task, a model must predict whether a hypothesis is entailed (i.e. supported) by a given premise. For training, we use the Pascal RTE-1 dataset [53] with crowd-sourced labels from [219]. The dataset consists of 800 premise-hypothesis pairs annotated by 164 different annotators with 10 annotations per pair. As an out-of-domain test set, we use the Stanford Natural Langauge Inference dataset (SNLI) [37], where we transform the task into binary classification by collapsing the "neutral" and "contradiction" classes into a single class.

Medical Relation Extraction (MRE) Medical relation extraction (MRE) seeks to predict what relations hold between different biomedical entities in sentences extracted from biomedical papers. The MRE dataset used for training in this work is the crowd-sourced dataset from [69], which collected crowd annotations from 3,984 sentences from PubMed abstracts [243] annotated by at least 15 annotators for 14 different UMLS [33] relations. Here we focus on the 975 sentence subset which also received expert annotations, specifically for the "cause" relationship. As such, we follow previous work [235] and frame the task as a binary classification problem, where a positive label indicates the "cause" relation exists. For testing, we use the causal claim-strength dataset curated from [257], which contains 1,126 sentences from news articles and scientific papers related to health science labeled for causal claim strength (statement of no relation, correlational, conditional causal, and causal). We convert the dataset to a binary classification problem by combining the "conditional causal" and "causal" classes into the positive class and the "correlational" and "no relation" classes into the negative class.

Part-of-Speech Tagging (POS) The POS tagging task is a sequence tagging task, where the goal is to predict the correct part-of-speech for each token in a sentence. For training data, we use the Gimpel dataset from [85] with the crowd-sourced labels provided by [111] mapped to the universal POS tag set in [184]. The dataset consists of 1000 tweets (17,503 tokens) labeled with Universal POS tags and annotated by 177 annotators. Each token received at least 5 annotations. We use the publicly available sample of the Penn Treebank POS dataset [157] accessed from NLTK [28] as our out-of-domain test set, which consists of 3,914 sentences from Wall Street Journal articles (100,676 tokens).

Toxicity Detection Finally, to measure performance on a highly subjective task, we use the toxicity detection dataset created as a part of the Google Jigsaw unintended bias in toxicity classification competition[9]. The dataset we use comes from [89], which annotated 25,500 comments from the original Civil Comments dataset. The pool of annotators is specifically selected and split into multiple rating pools based on self-indicated identity group membership

[9] https://www.kaggle.com/competitions/jigsaw-unintended-bias-in-toxicity-classification

(African American and LGBTQ). This study revealed that one's identity group correlates with shifts in perception of which comments are toxic, thus demonstrating the subjectivity of the task. We randomly split the dataset into training and test, and for the test data we use the annotations in the original crowd-sourcing task; in other words, using a completely separate annotator pool that isn't selected based on identity groups.

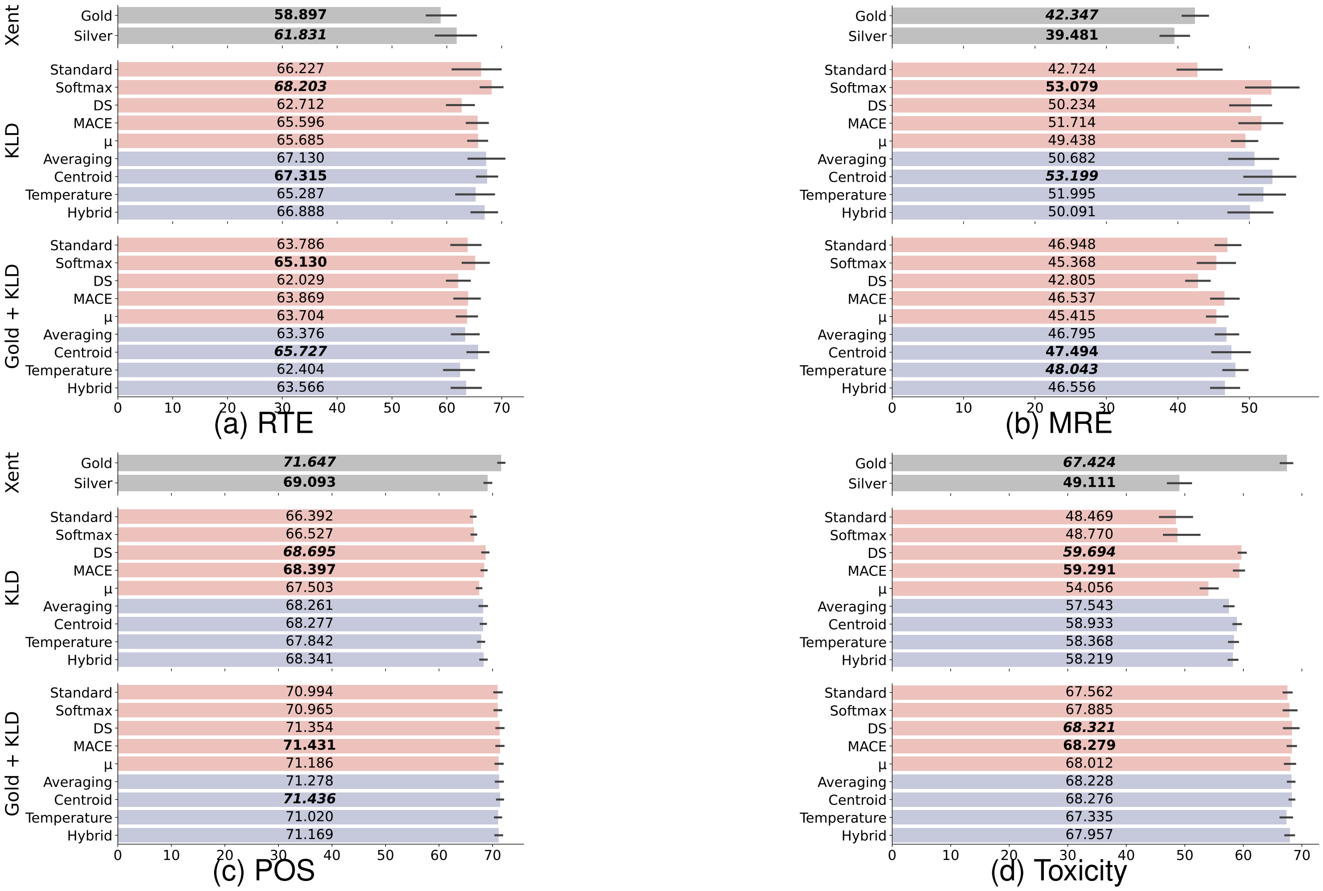
5.5 Results and Discussion

We evaluate the performance of each soft-labeling method across two metrics: F1 score and calibrated log-likelihood (CLL, [7]). F1 score provides an indication of the predictive ability for the classifiers trained on the soft-labels, while CLL describes how well calibrated each classifier is in terms of its predictive uncertainty. We use the CLL in order to obtain a fair comparison between methods, as it first performs temperature scaling on a held-out portion of the test set and measures the temperature-scaled negative log-likelihood on the rest of the test set, averaging results over 5 splits. Formal definitions of each metric can be found in subsection D.1. Additionally, we show the average performance of the individual methods (μ) to illustrate how the aggregation methods compare with the average individual performance, as well as results using only expert annotations (Gold) and only majority vote (Silver). We will first discuss general observations from our results, and based on this will provide answers for the research questions proposed in subsection 5.4.

5.5.1 Raw Performance

Overall In general, we see that the RTE and MRE datasets are much more difficult to generalize from than the POS and Jigsaw tasks, as reflected in the high variance of the results. Additionally, gold labels in these two settings yield worse performance than simply training on soft labels, as opposed to the in-domain setting reported in [235] where gold labels are needed for high performance, indicating that it may be more beneficial to use only crowd-sourced labels in the out-of-domain setting. POS tagging sees the best performance when using only gold labels, contrasting with results reported in [235] which show that adding soft labels with gold labels improves performance in the in-domain setting. The Toxicity task on the other hand benefits from both gold and crowd labels. This may be due to the input data coming from the same distribution, even though the gold labels come from disjoint sets of annotators. Additionally, all experiments using only soft-labels, with the exception of POS tagging, perform better than using hard labels obtained with majority vote.

Soft Labels Looking towards which soft-labeling method provides the best performance in the absence of gold labels, it is inconsistent across tasks. This was also seen in the survey by [235]. Where the aggregation methods provide the most benefit is in their **consistency** of performance.



(a) RTE

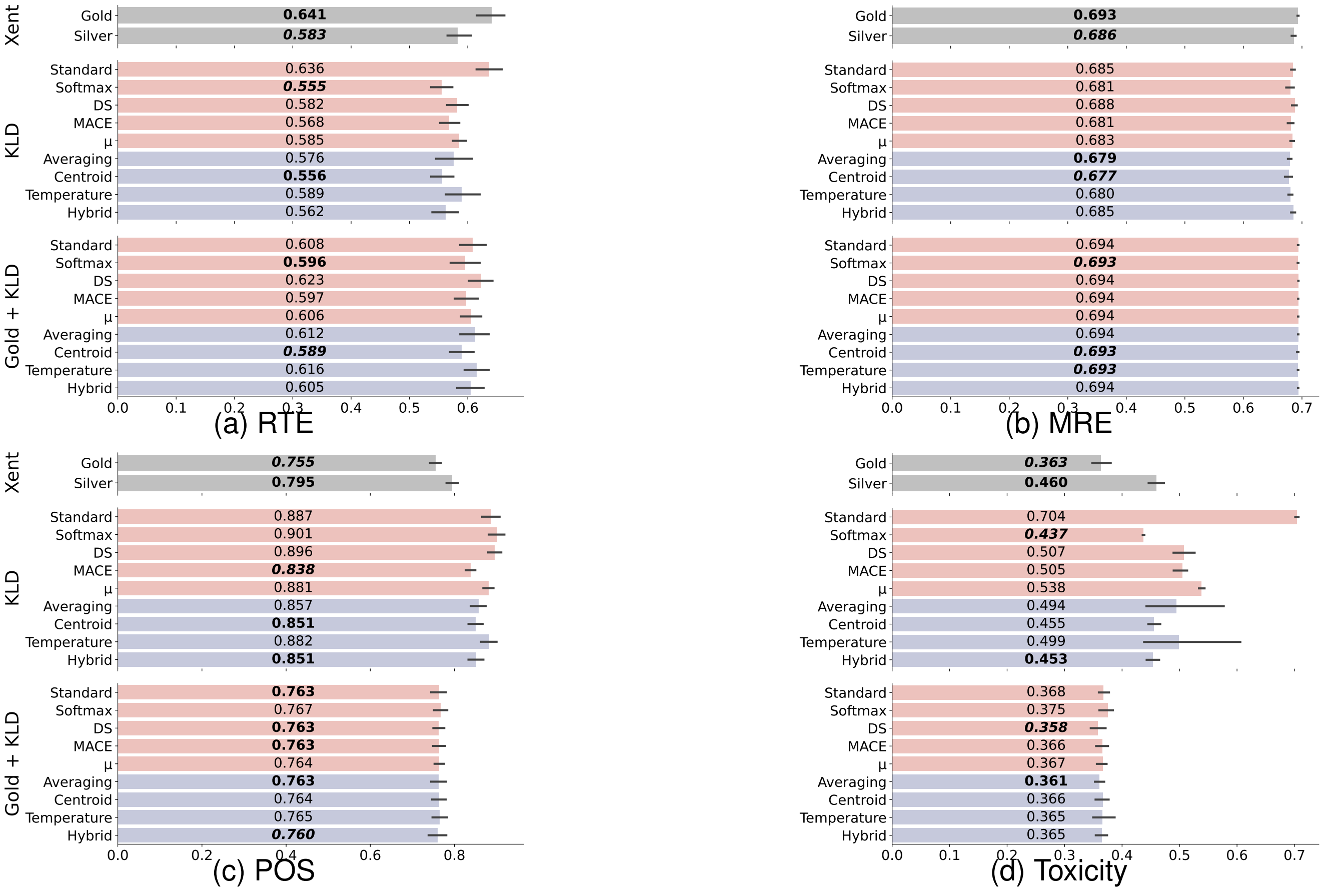
(b) MRE

(c) POS

(d) Toxicity

Figure 5.2: F1 scores for the (a) RTE, (b) MRE, (c) POS, and (d) Toxicity datasets on out of domain test sets for each given method. Results are averaged across 20 random seeds (5 seeds for Toxicity detection). Grey are hard labels only, red are individual methods and blue are aggregation methods. Best results within each setting (Xent, KLD, Gold + KLD) are given in ***bold italics***, second best results in **bold**.

In particular, the aggregation method using the JSC (Centroid in Figure 5.2) yields best or near-best performance across datasets, while the hybrid method works slightly better on POS tagging. This is despite fluctuations in performance for the unaggregated methods across tasks. For example, the softmax method works well for RTE and MRE, but worse for POS tagging and much worse for the highly subjective Toxicity detection task. The Bayesian methods show the opposite behavior, working well for POS tagging and Toxicity detection (potentially due to there being far more annotations from which to learn), but much worse for RTE and MRE. Looking at the average performance of the individual soft-labeling methods vs. the aggregation methods, we see that aggregation consistently performs better than average. Aggregation is also resistant to low-performing constituent distributions, as can be seen in the toxicity experiment where both the standard and softmax distributed labels produce significantly worse classifiers than those trained on labels from either Bayesian method, while each aggregation method remains close to the best performers. Finally, we also see that temperature scaling does not benefit performance



(a) RTE

(b) MRE

(c) POS

(d) Toxicity

Figure 5.3: Calibrated log-likelihood scores (lower is better) for the (a) RTE, (b) MRE, (c) POS, and (d) Toxicity datasets on out of domain test sets for each given method. Results are averaged across 20 random seeds (5 seeds for Toxicity detection). Grey are hard labels only, red are individual methods and blue are aggregation methods. Best results within each setting (Xent, KLD, Gold + KLD) are given in ***bold italics***, second best results in **bold**.

in this setting, and robust performance is achieved with the JSC alone.

Gold + KLD Adding gold labels for the RTE and MRE tasks leads to worse performance, potentially due to the limited amount of labeled data. This adds further evidence to the literature that soft labels can provide benefits over expert annotations for out-of-domain performance [183]. In terms of raw performance, gold labels are sufficient to obtain best performance for POS tagging, with soft labels not conferring benefits in the out-of-domain setting. This may be explained by the observation that the gold annotations for the POS dataset [85] were collected by researchers correcting labels for tweets pre-tagged by a tagger trained on Wall Street Journal articles (as in PTB), while the crowd-sourced annotations we use from [111] are annotated from scratch with minimal context, only seeing three words at a time. As such, while there is a significant difference between the source of input data between train and test, there may be less difference in terms of gold labels. For the toxicity detection task, all methods perform within reasonable ranges of each other, with the Bayesian methods and basic averaging conferring slightly better performance.

5.5.2 Uncertainty Estimation

Overall We see that uncertainty estimation as measured using calibrated log-likelihood can be improved with the addition of soft-labels in all cases except for POS tagging. The benefits are again more pronounced for the RTE and MRE tasks, where training data is limited. We also see inconsistency from the individual soft labeling methods across tasks, while the aggregation methods (and particularly the JSC) offer much more consistent uncertainty estimation which is better or approximately equal to the performance of the best performing individual method.

Soft Labels When looking at soft-labels only, the JSC aggregation method provides the most consistent results across tasks, with either the best or second best performance. The hybrid method also offers good uncertainty estimation, especially in the large-data regime of POS tagging and Toxicity detection, though less so for MRE.

Gold + KLD As with the raw performance results, including gold labels in a multi-task setup yields better uncertainty estimation when labeled data is abundant; otherwise using only soft-labels yields better uncertainty estimation.

5.5.3 Research Questions

RQ1: Best methods for OOD performance. In the out-of-domain setting, we find that among individual soft-labeling techniques, no consistent and clear best performer arises. Aggregating the soft-labels appears to mitigate these fluctuations in performance; in particular, aggregating using the JSC of the individual distributions, which leads to consistently best or near-best performance on all tasks.

RQ2: Does aggregation help? We find that aggregating multiple views of crowd-labels sometimes leads to better performance in the out of distribution setting, but will generally be at least approximately as good as the best performing individual methods regardless of poor performance induced by some individual methods. This is illustrated by the observation that on all tasks in both the multi-task and single-task settings, at least one individual soft labeling method leads to noticeably poorer performance than the best individual methods, while aggregating across the soft-labeling methods using the JSC is consistently high performing.

Dataset	Avg	Centroid	Temp	Hybrid
RTE	-	0.993**	-	-
MRE	-	0.998**	-	-
POS	-	0.987**	-	-
Jigsaw	-	0.996**	-	-

Table 5.1: Pearon correlation between the JS divergence of different aggregation methods and individual methods correlated with the individual method's average JS divergence to each other individual method. We only report correlation scores that are significant at p < 0.05 (- indicates no significance).

RQ3: Uncertainty estimation from soft-labeling. We find that in the absence of hard-labels, different individual soft-labeling methods are inconsistent in their uncertainty estimation across tasks. Again, aggregating these different views of the crowd-sourced labels mitigates these fluctuations. As with raw performance, we find that the Jenson-Shannon centroid is a sensible and consistent choice across tasks in the out-of-distribution setting,

5.5.4 Analysis

We briefly analyze the difference in JSD (Equation 5.5) between the aggregation methods and each individual method on each dataset. We first ask if the performance of an aggregation method correlates with its average JSD to the distribution generated by the best performing individual method. Looking at the Pearson correlation on each dataset, we find that a statistically significant correlation appears for the RTE dataset (RTE: 0.962, p < 0.05), while the scores obtained for other datasets were not significant. As such, it suggests the possibility that performance is correlated with distance to the best performing distribution, though more experiments would be needed for statistically significant scores on the other datasets.

We next explore how the JSD of each aggregation method to each individual method relates to how close the distribution produced by that individual method is to all other individual methods. This is to understand if there is a relationship between how close the individual methods are and the distribution obtained from each aggregation method. In other words, we correlate the following values:

$$\text{JSD}(Q \| p_m), \frac{1}{M-1} \sum_{k!=m} \text{JSD}(p_m \| p_k)$$

where Q is the distribution produced by one of the aggregation methods and p_m is the distribution of individual method m. Our hypothesis is that the JSC aggregation method produces a distribution which is closer to the distributions in the ensemble which are more similar to each other, as opposed to simple averaging which may be influenced by more divergent distributions. Our hypothesis for the JSC is confirmed by the results in Table 5.1, where the average JSD of the distributions produced by the JSC aggregation method to those of a given individual method is highly correlated with the average JSD of that individual method to all other individual methods. This suggests that aggregating using the JSC will lead to distributions closer to the hubs of an ensemble, where many of the individual distributions are similar. This may be desirable if those different views are representative of the problem one is modeling; the downside is the potential to ignore divergent views of the data which could be informative. We leave further exploration of this tradeoff to future work.

5.6 Conclusion

In this work we present a systematic comparison of soft-labeling techniques from crowd-sourced labels and demonstrate their utility on out-of-domain performance for several text-classification tasks. The out-of-domain setting allows us to observe how well learning from crowd-sourced soft-labels enables generalization to unseen domains of data, potentially reflecting the “dark knowledge” imparted by these labels. Given than no consistent best performing model appears, we propose four novel methods for aggregating multiple views of crowd-sourced labels into a combined distribution, demonstrating that doing so leads to consistently robust performance across tasks despite fluctuations in performance shown by the constituent views. Concretely, we show that using the JSC between the constituent distributions yields consistently high raw performance and good uncertainty estimation, and is resilient to fluctuations in performance of the individual methods. This constitutes a low-cost solution to acquiring reliable soft-labels from crowd-annotations which oftentimes outperform expert labels on out-of-domain data.

Acknowledgements

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 801199.

6 CITEWORTH: Cite-Worthiness Detection for Improved Scientific Document Understanding

6.1 Introduction

Building effective NLP systems from scientific text is challenging due to the highly domain-specific and diverse nature of scientific language, and a lack of abundant sources of labelled data to capture this. While large scale repositories of extracted, structured, and unlabelled plain-text scientific documents have recently been introduced [149], most datasets for downstream tasks such as named entity recognition [140] and citation intent classification [48] remain limited in size and highly domain specific. This begs the question: what useful training signals can be automatically extracted from massive unlabelled scientific text corpora to help improve systems for scientific document processing?

Scientific documents contain much inherent structure (sections, tables, equations, citations, etc.), which can facilitate creating large labelled datasets. Some recent examples include using paper field [26], the section to which a sentence belongs [48], and the cite-worthiness of a sentence [48, 223] as a training signal.

Cite-worthiness detection is the task of identifying *citing sentences*, i.e. sentences which contain a reference to an external source. It has useful applications, such as in assistive document editing, and as a first step in citation recommendation [75]. In addition, cite-worthiness has been shown to be useful in helping to improve the ability of models to learn other tasks [48]. We also hypothesize that there is a strong domain shift between how different fields use citations, and that such a dataset is useful for studying domain adaptation problems with scientific text.

However, constructing such a dataset to be of high quality is surprisingly non-trivial. Building a dataset for cite-worthiness detection involves extracting sentences from a scientific document, labelling whether each sentence contains a citation, and removing all citation markers. As a form of distant supervision, this naturally comes with the hazard of adding spurious correlations, such as poorly removed citation text causing ungrammatical sentences and hanging punctuation, which can trivially indicate a cite-worthy or non-cite-worthy sentence. Additionally, the task itself is quite difficult to learn, as different fields employ citations differently, and whether or not a sentence contains a citation depends on factors such as the context in which it appears. Given this, we present CITEWORTH, a rigorously curated dataset for cite-worthiness detection in English. CITEWORTH contains rich metadata, such as authors and links to cited papers, and all data is provided in *full paragraphs*: every sentence in a paragraph is labelled in order to provide sentence *context*. We offer the dataset to the research community to facilitate further research on cite-worthiness detection and related scientific document processing tasks.

Using CITEWORTH, we ask the following primary research questions:

RQ1: How can a dataset for cite-worthiness detection be automatically curated with low noise (§6.3)?

RQ2: What methods are most effective for automatically detecting cite-worthy sentences (§6.4)?

RQ3: How does domain affect learning cite-worthiness detection (§6.5)?

RQ4: Can large scale cite-worthiness data be used to perform transfer learning to downstream scientific text tasks (§6.6)?

We demonstrate that CITEWORTH is of high quality through a manual evaluation, that there are large differences in how models generalize to data from different fields, and that sentence context leads to significant performance improvements on cite-worthiness detection. Additionally, we find that cite-worthiness is a useful task for transferring to downstream scientific text tasks, in particular citation intent classification, for which we offer performance improvements over the current state-of-the-art model SciBERT [26].

In sum, our **contributions** are as follows:

- CITEWORTH, a dataset of 1.2M rigorously cleaned sentences from scientific papers labelled for cite-worthiness, balanced across 10 diverse scientific fields.
- A method for cite-worthiness detection which considers the entire paragraph a sentence resides in, improving by 5 F1 points over the state of the art model for scientific document processing, SciBERT [26].
- A thorough analysis of the problem of cite-worthiness detection, including explanations of predictions and insight into how scientific domain affects performance.
- New state of the art on citation intent detection via transfer learning from joint citation detection and language model fine-tuning on CITEWORTH, with improved performance over SciBERT on several other tasks.

6.2 Related Work

6.2.1 Cite-Worthiness Detection

Cite-worthiness detection is the task of identifying *citing sentences*, i.e. sentences which contain a reference to an external source. The reasons for citing are varied, e.g. to give credit to existing ideas or to provide evidence for a claim being made. [223] perform cite-worthiness detection using SVMs with features such as unigrams, bigrams, presence of proper nouns, and the classification of previous and next sentences. They create a dataset from the ACL Anthology Reference corpus (ACL-ARC, [29]), using heuristics to remove citation markers. [75] document the performance of convolutional recurrent neural nets on a larger set of three datasets coming from ACL-ARC, arXiv CS [74], and Scholarly Dataset 2.[10] Datasets from these studies suffer from high class imbalance, are limited to only one or a few domains, and little analysis of the datasets is performed to understand the quality of the data or what aspects of the problem are difficult or easy. Additionally, no study to date has considered how sentence context can affect learning to perform cite-worthiness detection.

In addition to being a useful task in itself, cite-worthiness detection is useful for other tasks in scientific document understanding. In particular, it has been shown to help improve performance on the closely related task of citation intent classification [119] when used as an auxiliary task in a multi-task setup [48]. However, cite-worthiness detection has not been studied in a transfer learning setup as a pretraining task for multiple scientific text problems. In this work, we seek to understand to what extent cite-worthiness detection is a transferable task.

Scientific Document Understanding Numerous problems related to scientific document understanding have been studied previously. Popular tasks include named entity recognition [140, 122, 65, 150] and linking [259], keyphrase extraction [12, 15], relation extraction [130, 150], dependency parsing [121], citation prediction [107], citation intent classification [119, 48], summarization [50], and fact checking [239].

Datasets for scientific document understanding tasks tend to be limited in size and restricted to only one or a few fields, making it difficult to build models with which one can study cross-domain performance and domain adaptation. Here, we curate a large dataset of cite-worthy sentences spanning 10 different fields, showing that such data is both useful for studying domain adaptation and for transferring to related downstream scientific document understanding tasks.

6.3 RQ1: CiteWorth Dataset Construction

The first research question we ask is: How can a dataset for cite-worthiness detection be automatically curated with low noise? To answer this, we start with the S2ORC dataset of extracted plain-text scientific articles [149]. It consists of data from 81.1M English scientific articles, with full structured text for 8.1M articles. S2ORC uses ScienceParse[11] to parse PDF documents and Grobid[12] to extract structured data from text. As such, the data also includes rich metadata, e.g. Microsoft Academic Graph (MAG) categories, linked citations, and linked figures and tables. Throughout this work, a "citation span" denotes a span containing citation text (e.g. "[2]"), and a "citation marker" is any text that trivially indicates a citation, such as the phrase "is shown in." A citation span is also a type of citation marker. It is important to remove all citation markers from the dataset to prevent the model learning to use these signals for prediction.

[10] http://www.comp.nus.edu.sg/~sugiyama/SchPaperRecData.html

[11] https://github.com/allenai/scienceparse

[12] https://github.com/kermitt2/grobid

6.3.1 Data Filtering

Given the size of S2ORC, we first reduce the candidate set of data to papers where all of the following are available.

- Abstract
- Body text
- Bibliography
- Tables and figures
- Venue information
- Inbound citations
- Microsoft Academic Graph categories

Filtering based on these criteria results in 5,494,387 candidate papers from which to construct the dataset. After filtering the candidate set of papers, we perform the following checks on the sentences in the body text.

1. Citation spans are parenthetical author-year or bracketed-numerical form.
2. Citation spans are at the end of a sentence.
3. All possible citation spans have been extracted by S2ORC.
4. No citation markers are left behind after removing citation spans from the text.
5. Sentence starts with a capital letter, ends with '.', '!', or '?', and is at least 20 characters long.

The detailed steps of extracting and labelling sentences based on these criteria are given in §6.3.2. With the first two criteria, we restrict the scope of cite-worthy sentences to being only those whose citation span comes at the end of a sentence, and whose citation format is parenthetical author-year form or bracketed-numerical form. In other words, cite-worthy sentences in our data are constrained to those of the following forms.

> This result has been shown in previous work (Author1 et al., ####, ...).
>
> This result has been shown in previous work [#-#].

In this, we ignore citation sentences which contain inline citations, such as "The work of Authors et al. (####) has shown this in previous work", as well as any sentence with a citation format that does not match the two we have selected.

Curating cite-worthy sentences as such helps prevent spurious correlations in the data. Removing citations in the middle of a sentence runs the risk of rendering the sentence ungrammatical (for example, the above sample would turn into "The work *of has* shown this in previous work"), providing a signal to machine learning models. While there are cases where inline citations could potentially be removed in their entirety and not destroy the sentence structure, this is beyond the scope of this paper and left to future work.

Biology
Wood Frogs (Rana sylvatica) are a charismatic species of frog common in much of North America. They breed in explosive choruses over a few nights in late winter to early spring. *The incidence in Wood Frogs was associated with a die-off of frogs during the breeding chorus in the Sylamore District of the Ozark National Forest in Arkansas~~ (Trauth et al., 2000)~~.*

Computer Science
Land use or cover change is a direct reflection of human activity, such as land use, urban expansion, and architectural planning, on the earth's surface caused by urbanization~~ [1].~~ Remote sensing images are important data sources that can efficiently detect land changes. *Meanwhile, remote sensing image-based change detection is the change identification of surficial objects or geographic phenomena through the remote observation of two or more different phases~~ [2]~~.*

Table 6.1: Excerpts from training samples in CiteWorth from the Biology and Computer Science fields. Green sentences are cite-worthy sentences, from which citation markers are removed during dataset construction.

Metric	#
Total sentences	1,181,793
Total number of tokens	34,170,708
Train sentences	945,426
Dev sentences	118,182
Test sentences	118,185
Total cite-worthy	375,388 (31.76%)
Total non-cite-worthy	806,405 (68.24%)
Min char length	21
Max char length	1,447
Average char length	152
Median char length	142

Table 6.2: Various statistics of the CiteWorth dataset.

6.3.2 Extracting Cite-Worthy Sentences in Context

As we are interested in using sentence context for prediction, we perform extraction at the *paragraph level*, ensuring that all of the sentences in a given paragraph meet the checks given in §6.3.1. As such, our dataset construction pipeline for a given paper begins by first extracting all paragraphs from the body text which belong to sections with titles coming from a constrained list of permissible titles (e.g. "Introduction," "Methods," "Discussion") . The full list is provided in §E.1.

For a given paragraph, we first word and sentence tokenize the text with SciSpacy [169]. Each sentence is then checked for containing citations using the provided citation spans in the S2ORC dataset. In some cases, the sentence contains citations which were missed by S2ORC; these are checked using regular expressions (see §E.2). If a match is found the paragraph is ignored, as we only consider paragraphs where all citations have been extracted by S2ORC. Otherwise, the location and format of the citation is checked, again using regular expressions (see §E.2). If the citation is not at the end of the sentence, the paragraph is ignored. We then remove the citation text using the provided citation spans for all sentences which pass the above checks.

Simply removing the citation span runs the risk of leaving other types of citation markers, such as hanging punctuation and prepositional phrases e.g. “This was shown by the work of ~~Author et al. (####)~~.” To mitigate this, we remove all hanging punctuation at the end of a sentence that is not a period, exclamation point, or question mark, and check for possible hanging citations using the regular expression provided in §E.2. The regular expression checks for many common prepositional phrases and citation markers occurring as the last phrase of a sentence such as “see,” “of,” “by,” etc.

To handle issues with sentence tokenization, we also ensure that the first character of each sentence is a capital letter, and that the sentence ends with a period, exclamation point, or question mark. If all criteria are met for all sentences in a paragraph, the paragraph is added to the dataset. Finally, we build a dataset which is diverse across domains by evenly sampling paragraphs from the following 10 MAG categories, ensuring that each paragraph belongs to exactly one category: Biology, Medicine, Engineering, Chemistry, Psychology, Computer Science, Materials Science, Economics, Mathematics, and Physics. Example excerpts from the dataset are presented in Table 6.1, and the statistics for the final dataset are given in Table 6.2.[13]

6.3.3 Manual Evaluation

In order to provide some measure of the general quality of CITEWORTH, we perform a manual evaluation of a sample of the data. We annotate the data for whether or not citation markers are completely removed, and for whether or not the sentences are well-formed, containing no obvious extraction artifacts. We sample 500 cite-worthy sentences and 500 non-cite-worthy sentences randomly from the data. Additionally, we compare to a baseline where the only heuristic used is to remove citation spans based on the provided spans in the S2ORC dataset. We again sample 500 cite-worthy and 500 non-cite-worthy sentences for annotation. The two sets are shuffled together and given to an independent expert annotator with a PhD in computer science for labelling. The annotator is instructed to label if the sentences are complete and have no hanging punctuation or obvious extraction errors, and if there are any textual indicators that the sentences contain a citation. The results for the manual annotation can be seen in Table 6.3.

We see that the CITEWORTH data are of a much higher quality than removing citation markers based only on the citation spans. Overall, our heuristics improve on extraction quality by 6.83% absolute and on removing markers of citations by 5.32% absolute. This results in 1.1% of the sample data containing sentence cleaning issues, and 1.9% having trivial markers indicating a citation is present. We argue that this is a strong indicator of the quality of the data for supervised learning.

[13] The full dataset can be downloaded from this repository: `https://github.com/copenlu/cite-worth`

Method	Extracted Correct	Markers Removed
Baseline	92.07	92.78
Ours	**98.90**	**98.10**

Table 6.3: Results of manually annotating 1000 random sentences (per method) from CITEWORTH and a naive baseline which only removes citations based on provided citation spans . "Extracted Correct" are results for correctly extracting the sentences (i.e. that sentences are tokenized correctly and are grammatical), and "Markers Removed" are results for successfully removing citation markers. The data curated using our method has 6% fewer errors in terms of extraction and removal of citation markers, and less than 2% of the samples have some form of citation marker.

6.4 RQ2: System Evaluation

Next, we ask: what methods are most effective for performing cite-worthiness detection? To answer this and characterize the difficulty of the problem, we run a variety of baseline models on CITEWORTH. The hyperparameters selected for each model, as well as hyperparameter sweep information, are given in Appendix G.2.6.

Logistic Regression As a simple baseline, we use a logistic regression model with TF-IDF input features.

[75] The convolutional recurrent neural network (CRNN) model from [75]. They additionally use oversampling to deal with class imbalance.

Transformer We additionally train a Transformer model from scratch [237], tuning the model hyperparameters on a subset of the training data via randomized grid search.

BERT We use a pretrained BERT model [62] due to the strong performance of large pretrained Transformer models on downstream tasks.

SciBERT SciBERT [26] is a BERT model pretrained on a large corpus of scientific text from Semantic Scholar [6], and is therefore potentially better suited to fine-tuning on scientific cite-worthiness detection.

SciBERT + PU Learning We experiment with SciBERT trained using positive-unlabelled (PU) learning [71] which has been shown to significantly improve performance on citation needed detection in Wikipedia and rumour detection on Twitter [254]. The intuition behind PU learning is to assume that cite-worthy data is labelled and non-cite-worthy data is unlabelled, containing some cite-worthy examples. This is to mitigate the subjectivity involved in adding citations to

Method	P	R	F1
Logistic Regression	$46.65_{0.00}$	$64.88_{0.00}$	$54.28_{0.00}$
[75]	$49.57_{0.96}$	$65.56_{2.61}$	$56.41_{0.34}$
Transformer	$47.92_{0.78}$	$71.59_{1.74}$	$57.39_{0.10}$
BERT	$55.04_{0.66}$	$69.02_{1.33}$	$61.23_{0.21}$
SciBERT-no-weight	$\mathbf{65.94}_{0.37}$	$51.62_{0.53}$	$57.91_{0.30}$
SciBERT	$57.03_{0.50}$	$68.08_{1.03}$	$62.06_{0.15}$
SciBERT + PU	$49.46_{0.83}$	$\mathbf{82.12}_{1.40}$	$61.73_{0.27}$
Longformer-Solo	$57.21_{0.25}$	$68.00_{0.41}$	$62.14_{0.02}$
Longformer-Ctx	$59.92_{0.28}$	$77.15_{0.49}$	$\mathbf{67.45}_{0.06}$

Table 6.4: F1 performance of baselines on the test set of CITEWORTH. Results are averaged across 5 seeds, with standard deviations given in the subscripts.

sentences. Technically, this involves training a classifier on the positive-unlabeled data which will predict the probability that a sample is labeled, and using this to estimate the probability that a sample is positive given that it is unlabeled. One then trains a second model where positive samples are trained on normally and unlabeled samples are duplicated and trained on twice, once as positive and once as negative data, weighed by the first model's estimate of the probability that the sample is positive.

Longformer-Ctx Finally, we test our novel contextualized prediction model based on Longformer [27]. Longformer is a Transformer based language model which uses a sparse attention mechanism to scale better to longer documents. We process an entire paragraph at a time, separating each sentence with a `[SEP]` token. Each `[SEP]` token representation at the output of Longformer is then passed through a network with one hidden layer and a classifier. As a control, we also experiment with Longformer using only single sentences as input (Longformer-Solo).

Due to the imbalance in the distribution of classes, the loss for each of the models is weighted. For comparison, we include results for SciBERT without weighting the loss function. The results for our baseline models on the test set of the dataset are given in Table 6.4.[14]

Our results indicate that context is critical, resulting in the best F1 score of 67.45 (Longformer-Ctx) and a 5.31 point improvement over the next best model. Using class weighting is also highly important, resulting in another increase of over 4 F1 points. Compared to not using class weights, PU learning performs significantly better, and leads to the highest recall of all models under test. Additionally, language model pre-training is useful, as BERT, SciBERT, and Longformer all perform significantly better than a Transformer trained from scratch and the model from [75].

To gain some insight into what the model learns, we visualize the most salient features from SciBERT for selected easy and hard examples. We use the single-sentence model instead of the paragraph model for simplicity. "Easy" samples are defined as those which the model

[14] The code for all experiments can be found here: `https://github.com/copenlu/cite-worth`

predicted correctly with high confidence, and “hard” examples are defined as those for which the model had low confidence in its prediction. We use the InputXGradient method [125], specifically the variant using L2 normalization over neurons to get a pre-embedding score, as it has been recently shown to have the best overall agreement with human rationales versus several other explainability techniques [9]. The method works by calculating the gradient of the output with respect to the input, then multiplies this with the input. In the examples below “C” refers to an example whose gold label is cite-worthy, and “N” refers to an example whose gold label is non-cite-worthy.

The model is able to pick up on obvious markers of cite-worthy and non-cite-worthy sentences for the following correctly classified examples, such as that a sentence refers to a preprint or to different sections within the paper itself:

C: [CLS] in this note , we follow the approach to the en ##och ##s conjecture outlined in the preprint . [SEP]

N: [CLS] conclusions are provided in section 4 . [SEP]

We also see that the dataset contains many relatively difficult instances, as we show in the following incorrectly classified examples. E.g., the model observes “briefly discussed” as an indicator that an instance is non-cite-worthy when it is in fact cite-worthy, and that “described earlier” and “previous work” signal that a sentence is cite-worthy when it is in fact labelled as non-cite-worthy.

C: [CLS] some approaches for the solution as well as their limitations are briefly discussed . [SEP]

N: [CLS] this simple and fast technique for the production of snps was described earlier in our previous work . [SEP]

We hypothesize that in such instances, context can help the most in disambiguating which sentences in a paragraph should be labelled as cite-worthy. Additionally, other information such as the section in which a sentence resides could help. E.g., to correctly label the fourth statement above as “non-cite-worthy”, it may help to see that the last sentence of the paragraph is “In our previously published work, it was reported that SNPs were joined together by the heat treatment, and this process led to increase in the sizes of SNPs which finally resulted in sharper

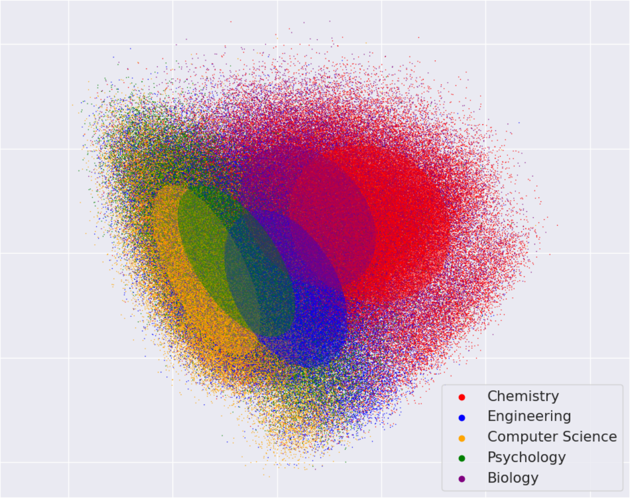


Figure 6.1: Visualizing the BERT embeddings for 5 of the 10 domains from CITEWORTH using the method by [2]. Clustering is performed using Gaussian Mixture Models.

XRD peaks" which is a cite-worthy sentence. Additionally, it may help to know that it resides in the "Discussion" section of the paper.

6.5 RQ3: Domain Evaluation

We next ask: how does domain affect learning to perform cite-worthiness? To answer this, we study the relationships between cite-worthiness data from different fields and how the Longformer-Ctx model performs in a cross-domain setup. For ease of analysis we limit the scope of fields to 5 of the 10 fields in the dataset: Chemistry, Engineering, Computer Science, Psychology, and Biology.

First, we visualize the embedding space for data from each of these domains using the method of [2]. In this, the data is passed through BERT (specifically the base, uncased variant) and the output representations for each token in a sentence are average pooled. These representations are visualized in 2D space via PCA in Figure 6.1. It is clear that similar fields occupy closer space, with 'engineering' and 'computer science' sharing closer representations, as well as 'biology' and 'chemistry'. We perform clustering on this data using a Gaussian mixture model similarly to [2], finding that domains form somewhat distinct clusters with a cluster purity of 57.61. This demonstrates that the data in different fields are drawn from different distributions, thus differences could exist in a model's ability to perform cite-worthiness detection on out of domain data.

To test this, we perform a cross-validation experiment using the 5 selected fields, training on one field and testing on another for all 25 combinations. The results for the 5x5 train/test setup using Longformer-Ctx are given in Table 6.5.

Not surprisingly, the best performance for each split occurs when training on data from the

Test Train	Ch	E	CS	P	B
Ch	**67.58**	58.41	56.86	62.35	68.23
E	66.62	**60.25**	60.11	64.02	68.07
CS	65.05	59.36	**61.99**	63.85	66.72
P	65.49	58.03	56.69	**65.10**	68.27
B	66.59	58.80	58.22	64.54	**69.12**
σ	0.90	0.78	2.02	0.92	0.77
ρ	0.87	0.86	0.76	0.67	0.79

Table 6.5: F1 performance on different domain adaptation settings for the fields (Ch)emistry, (E)ngineering, (C)omputer (S)cience, (P)sychology, and (B)iology. Out-of-domain tests use the entire set of data from that field, while in domain tests use 80% of data for training, 10% for validation, and 10% for test. σ is the standard deviation of performance of different train domains on the given test domain, and ρ is Pearson correlation between performance and Euclidean distance from the train domain cluster to the test domain cluster.

same field. We also observe high variance in the maximum performance for each field (σ = 3.32), and between different fields on the same test data, despite large pretrained Transformer models being relatively invariant across domains [255]. This suggests stark differences in how different fields employ citations. Additionally, we observe a strong (inverse) correlation between distance in the embedding space and performance on different domains, showing that using more similar data for training helps on out-of-domain performance [2].

6.6 RQ4: Cite-Worthiness for Transfer Learning

The final question we ask is: to what extent is cite-worthiness detection transferable to downstream tasks in scientific document understanding? To answer this, we fine tune SciBERT on the task of cite-worthiness detection as well as masked language modeling (MLM) on CITEWORTH, followed by fine-tuning on several document understanding tasks. We use SciBERT in order to have a direct comparison with previous work [26]. The tasks we evaluate on come from [26] and are categorized as follows.

- Named Entity Recognition (NER)/PICO: These tasks involve labelling the spans of different types of entities in a document.
- Relation Extraction (REL): This task involves labelling a sequence for the relationship between two entities.
- Text classification (CLS): Finally, we test on several text classification tasks (citation intent classification and paper field classification), where the goal is to classify a sentence into one or more categories.

We compare five variants of pre-training and fine-tuning, given as follows.

Dataset	Reference	Task	Base	LM	Cite	LMCite
BC5CDR	[140]	NER	$89.84_{0.18}$	$\mathbf{90.03_{0.11}}$	$89.73_{0.25}$	$90.02_{0.79}$
JNLPBA	[122]	NER	$77.02_{0.36}$	$77.13_{0.53}$	$76.97_{0.44}$	$\mathbf{77.15_{0.58}}$
NCBI-Disease	[65]	NER	$\mathbf{88.79_{0.35}}$	$88.53_{0.58}$	$88.66_{0.57}$	$88.31_{0.43}$
SciERC	[150]	NER	$67.08_{0.50}$	$66.64_{0.47}$	$67.12_{0.46}$	$\mathbf{67.48_{0.45}}$
EBM-NLP	[174]	PICO	$76.61_{0.21}$	$\mathbf{76.69_{0.28}}$	$76.55_{0.88}$	$76.41_{0.32}$
ChemProt	[130]	REL	$83.17_{0.43}$	$\mathbf{83.26_{0.90}}$	$82.70_{1.06}$	$83.16_{0.63}$
SciERC	[150]	REL	$80.21_{0.81}$	$\mathbf{80.68_{1.04}}$	$80.00_{1.73}$	$80.58_{0.96}$
ACL-ARC	[119]	CLS	$71.82_{2.93}$	$70.95_{2.25}$	$\mathbf{73.68_{2.75}}$	$72.92_{3.76}$
SciCite	[48]	CLS	$84.83_{0.65}$	$85.18_{0.47}$	$85.32_{0.16}$	$\mathbf{85.35_{0.29}}$
PaperField	[26]	CLS	$65.48_{0.18}$	$\mathbf{65.57_{0.27}}$	$65.46_{0.24}$	$65.42_{0.48}$
Average			78.386	78.466	78.619	**78.680**

Table 6.6: Performance on various downstream scientific document understanding tasks as presented by [26]. The metrics used are the same as in their paper: NER is span-level F1, PICO is token level F1, relation extraction is macro-F1, and ChemProt is micro-F1. All runs are averaged across 5 seeds. Subscripts are the standard deviation for 5 runs.

Base The original SciBERT model.

LM SciBERT with MLM fine tuning on CiteWorth.

Cite SciBERT fine-tuned for the task of cite-worthiness detection. The classifier is a pooling layer on top of the `[CLS]` representation of SciBERT, followed by a classification layer.

LMCite SciBERT with MLM fine tuning and cite-worthiness detection. The two tasks are trained jointly i.e. on each batch of training, the model incurs a loss for both MLM and cite-worthiness detection which are summed together.

The results for all experiments are given in Table 6.6. Note that the reported results for SciBERT are on re-running the model locally for fair comparison. We first observe that incorporating our dataset into fine-tuning tends to improve model performance across all tasks to varying degrees, with the exception of NER on the NCBI-Disease corpus. The tasks where cite-worthiness as an objective has the most influence are the two citation intent classification tasks (ACL-ARC and SciCite). We see average improvements of 1.8 F1 points for the ACL-ARC dataset (including 2 points F1 improvement over the minumum and maximum model performance of SciBERT) and 0.5 F1 points on SciCite. The best average performance is from the model which incorporates both MLM and cite-worthiness as an objective, which we call CiteBert.[15]

For other tasks, fine-tuning the language model on CiteWorth data tends to be sufficient for improving performance, though the margin of improvement tends to be minimal. This is in line with previous work reporting that language model fine-tuning on in-domain data leads to improvements on end-task fine-tuning [94]. CiteWorth is relatively small compared to the

[15] We release two CiteBert models available from the HuggingFace model hub: `copenlu/citebert` and `copenlu/citebert-cite-only`.

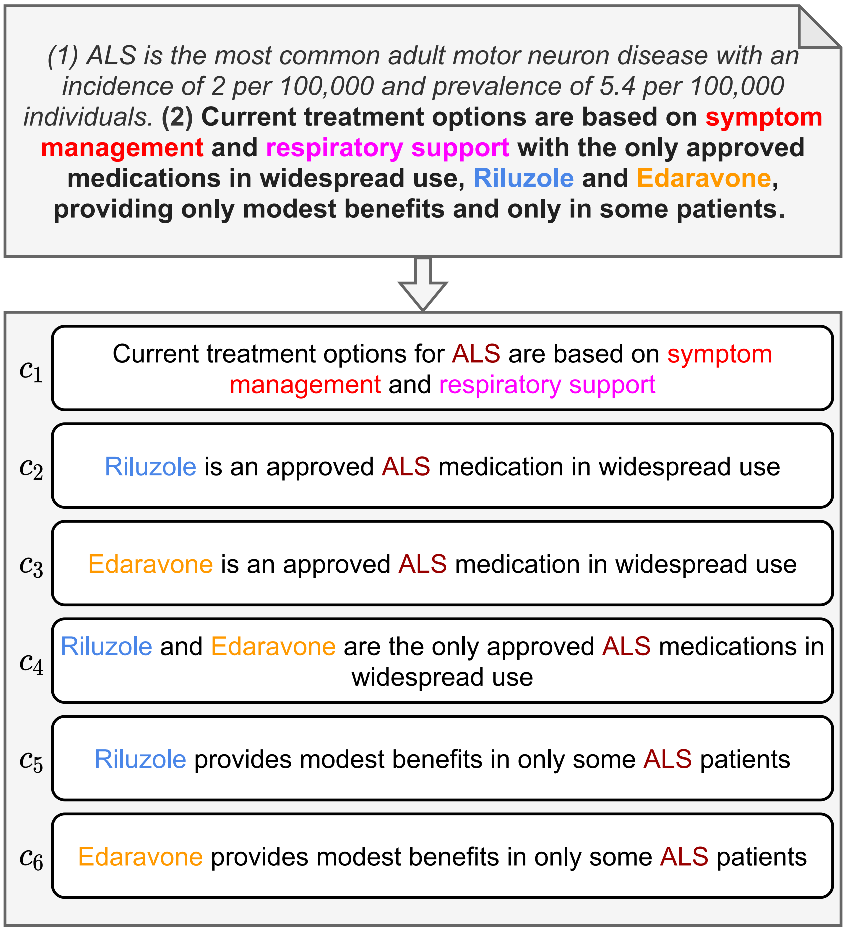
corpus on which SciBERT is originally trained (30.7M tokens for the train and dev splits on which we train versus 3.1B), so one could potentially see further improvements by incorporating more data or including cite-worthiness as an auxiliary task during language model pre-training. However, this is outside the scope of this work.

6.7 Conclusion

In this work, we present an in-depth study into the problem of cite-worthiness detection in English. We rigorously curate CiteWorth, a high-quality dataset for cite-worthiness detection; present a paragraph-level contextualized model which improves by 5.31 F1 points on the task of cite-worthiness detection over the existing state-of-the-art; show that CiteWorth is a good testbed for studying domain adaptation in scientific text; and show that in a transfer-learning setup one can achieve state of the art results on the task of citation intent classification using this data. In addition to studying cite-worthiness and transfer learning, CiteWorth is suitable for use in downstream natural language understanding tasks. As we retain the S2ORC metadata with the data, one could potentially use the data to study joint cite-worthiness detection and citation recommendation. Additionally, one could explore other useful problems such as modeling different authors' writing styles and incorporating the author network as a signal. We hope that the data and accompanying fine-tuned models will be useful to the research community working on problems in the space of scientific language processing.

Acknowledgements

The research documented in this paper has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 801199.



(1) ALS is the most common adult motor neuron disease with an incidence of 2 per 100,000 and prevalence of 5.4 per 100,000 individuals. **(2) Current treatment options are based on symptom management and respiratory support with the only approved medications in widespread use, Riluzole and Edaravone, providing only modest benefits and only in some patients.**

c_1 Current treatment options for ALS are based on symptom management and respiratory support

c_2 Riluzole is an approved ALS medication in widespread use

c_3 Edaravone is an approved ALS medication in widespread use

c_4 Riluzole and Edaravone are the only approved ALS medications in widespread use

c_5 Riluzole provides modest benefits in only some ALS patients

c_6 Edaravone provides modest benefits in only some ALS patients

Figure 7.1: A complex excerpt from [159] (top) and the set of valid claims that can be generated from the bolded sentence (c1-c6).

7 Generating Scientific Claims for Zero-Shot Scientific Fact Checking

7.1 Introduction

Scientific documents contain complex assertions about scientific processes, making it difficult to automate important tasks such as claim extraction and scientific fact checking. Additionally, the collection of manually annotated labels to train models on tasks with scientific data is time consuming and expensive due to the need for domain expertise [50, 15, 136, 239, 63]. As such, methods which require less manual annotation are especially useful in this domain. This work addresses this challenge by exploring how automatic generation of scientific claims can assist with dataset creation and zero-shot fact checking in the biomedical domain.

Being able to reduce scientific text to atomic assertions has numerous possible applications, and is known to be helpful for scientific communication and machine processing of scientific concepts [133]. Claim generation can enable zero-shot fact checking, reducing the need for expert-labeled data [176], and can be used to expand existing datasets such as [239] and [206] without additional manual annotation. In this work we focus on the use of claim generation in scientific fact checking, demonstrating that claim generation enables zero-shot biomedical fact checking.

Generating scientific claims involves distilling a complex scientific sentence into one or more valid claims (see examples in Figure 7.1). As in previous work, we focus on biomedical claims as biomedical literature has long been a major focus in scientific natural language processing, as well as scientific fact checking [206, 239, 128]. While in [239], claims were rewritten by domain experts from complex citation sentences (citances), we propose methods for automatically generating claims and claim negations from this source.

Similar to other generation tasks, evaluating the quality of generated output requires multiple judgements beyond the fluency of the generated text, e.g., whether each claim is faithful to the source sentence, and is understandable on its own [209]. However, there are also other quality attributes that are important to assess specifically for scientific claims, such as whether each claim is atomic or check-worthy [254]. Given this, we propose a set of manual evaluation criteria and annotation guidelines for evaluating claim generation (§7.5.2).

Additionally, when generating claims to build datasets for tasks such as fact checking, a major challenge is creating refuted claims as negative training instances. Previous work has proposed automatic ways of generating refutations based on negating existing claims or creating claim variants via entity-replacement [176] and text-infilling using a pre-trained masked language model [206]. We improve upon this by introducing Knowledge Base Informed Negations (KBIN), a principled method to generate refutations that performs entity-replacement using the relations and learned embeddings of entities in a domain-specific knowledge base.

Contributions In sum, our contributions are:

- The first study on scientific claim generation, comparing both unsupervised (CLAIMGEN-ENTITY) and fully supervised (CLAIMGEN-BART) generation on biomedical text.
- KBIN, a novel method for generating refuted scientific claims which produces more convincing negations than previous work.
- Application of our claim generation methods on zero-shot scientific fact checking resulting in 90% of the performance of a model trained on in-domain manually written claims. Additionally, a rigorous evaluation study showing that CLAIMGEN-BART and KBIN produce significantly higher quality claims and more convincing negations than previous work.

7.2 Preliminaries

Valid Claims In this work, we define a *valid claim* as one which is fluent, atomic, de-contextualized, and accurately reflects the meaning of the original sentence. Fluency is concerned with a claim being a generally well-formed English sentence, and atomicity with a claim being a "verifiable statement expressing a finding about one aspect of a scientific entity or process, which can be verified from a single source" [239]. De-contextualilzation is concerned with a sentence being interpretable on its own, requiring none of the original surrounding text to resolve aspects of the sentence such as pronouns, abbreviations, etc., and can be handled by either directly de-contextualizing a sentence [46] or by ensuring that all of the context sentences are available to a model [240]. Check-worthy claims in the wild may not be fluent, atomic, or de-contextualized, however it is useful to generate such claims as they have been shown to be useful for automated processing of science concepts [133] and scientific fact checking [239].

Scientific Claim Generation At a high level, scientific claim generation is the task of distilling one or more *valid claims* from one or more sentences concerned with a scientific fact. More specifically, the task is defined as: given a scientific sentence s and optionally additional context sentences X, generate one or more claims $c_i \in C$ which are valid and entailed by s and X. In the context of fact checking, we must generate claims which are either *supported* or *refuted* by the literature, as well as those for which *not enough information* is present to make a veracity judgement, in order that they may be paired with appropriate evidence documents to serve as training data for fact checking systems. As such, we require methods which can take the claims in C which are entailed by the source sentence and generate negations to acquire *refuted* claims.

7.3 Generating Supported Claims

We experiment with two generation methods designed to produce claims which are *supported* by the source sentence. The first method is an entity-centric unsupervised method adapted from [176] which requires no <sentence, claim> pairs (CLAIMGEN-ENTITY). We also introduce a new method that uses BART [138] trained on a small set of <sentence, claim> pairs to directly generate claims (CLAIMGEN-BART). For each sample i, we refer to the input source sentence as s_i, the context sentences as $x_l^{(i)} \in X_i$ and the output claims as C_i consisting of k claims $\{c_1^{(i)} \dots c_k^{(i)}\}$ Following [239], we use citation sentences as unlabelled sentences for generation since these provide a natural link to an evidence document. Various components of our modeling pipelines take advantage of models pretrained on datasets for NER, NLI, QA, and fact-checking. We provide an overview of these datasets in §G.2.7.

7.3.1 CLAIMGEN-ENTITY

We adapt the entity-centric method presented in [176] as an unsupervised claim generation approach. This method has been tested on general domain fact checking, but has not been used for science claim generation and zero-shot scientific fact checking. In particular, we re-implement the base method used for generating supported claims and adapt it to the biomedical domain, substituting in a domain specific model for named-entity recognition. The method consists of the following steps for a given sample i:

1. Run named entity recognition (NER) on the input text to obtain a set of named entities E_i.
2. For each named entity $e_j^{(i)}$, generate a question $q_j^{(i)}$ about that entity which can be answered from s_i.
3. From $q_j^{(i)}$, generate the declarative form of the question to obtain claim $c_j^{(i)}$.

Named Entity Recognition For NER, we employ scispaCy [169], a spaCy[16] pipeline for scientific NLP. The NER model is trained on the MedMentions dataset [162], which consists of 4,392 PubMed abstracts exhaustively annotated for mentions of UMLS entities [33].

Question Generation For question generation, we use BART trained on questions from SQuAD [194]. As input for training, we encode a concatenation of the context and answer text from a given SQuAD question, and train the model to decode the question. During inference, we concatenate the source sentence s_i and an entity $e_j^{(i)}$ and sample a question $q_j^{(i)}$ for this pair using beam search.

Question to Claim Finally, as in [176], we use a second BART model to generate declarative claims from questions. We train the model on the QA2D dataset [60], which contains declarative full sentences paired with questions and their answer from SQuAD. The model is trained by encoding a concatenation of the question and answer, and decoding the full declarative sentence. At inference time, we concatenate and encode $q_j^{(i)}$ and $e_j^{(i)}$, and use beam search at the decoder to generate a claim $c_j^{(i)}$.

7.3.2 ClaimGen-BART

We introduce a fully-supervised model for claim generation based on BART trained on <citance, claim> pairs. For this, we use the manual citance re-writes released by the SciFact authors,[17] which consist of citances from scientific papers rewritten as one or more atomic claims which are directly entailed by the citance.

For training, we encode the citance, as well as the sentences immediately before and after the citance (the context), and train the decoder to generate claims directly. We choose to encode the context as well to help *de-contextualize* generated claims. We concatenate the citance and context using a double pipe (i.e. $X_i||s_i$), and train the encoder to generate one claim at a time.

[16] https://spacy.io/

[17] https://github.com/allenai/scifact/blob/master/doc/claims-with-citances.md

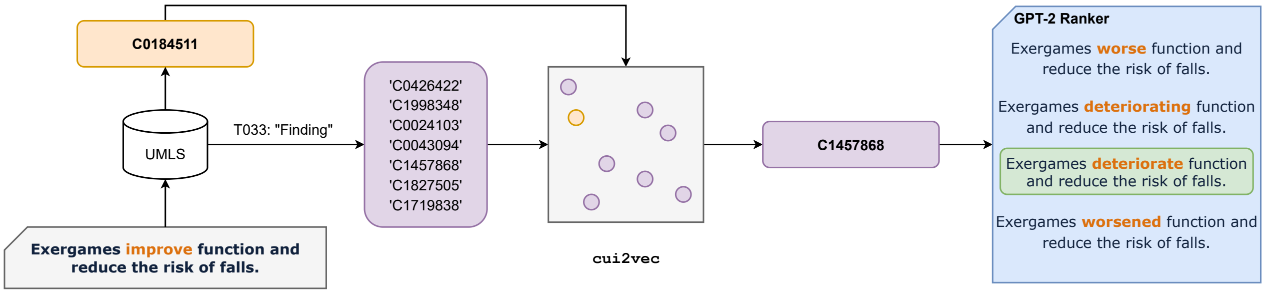

C0184511
UMLS
T033: "Finding"
'C0426422'
'C1998348'
'C0024103'
'C0043094'
'C1457868'
'C1827505'
'C1719838'
cui2vec
C1457868
GPT-2 Ranker
Exergames worse function and reduce the risk of falls.
Exergames deteriorating function and reduce the risk of falls.
Exergames deteriorate function and reduce the risk of falls.
Exergames worsened function and reduce the risk of falls.
Exergames improve function and reduce the risk of falls.

Figure 7.2: KBIN method. We start with NER and linking to UMLS using scispaCy. We then find the most similar concepts with the same type using `cui2vec`, replace the entity in the source sentence using the canonical name and aliases of similar entities, and rank them using GPT-2. Finally, from the highest ranked replacements, we select the claim which maximizes contradiction with the original claim using an external NLI model.

Algorithm 7.1 KBIN algorithm

```
function GetNegation($c$, KB, $V$, $N$)
    $E \leftarrow$ NER$(c)$
    $\bar{C} \leftarrow []$
    for $e_j$ in $E$ do
        $u_j \leftarrow$ LINK$(e_j)$
        $R \leftarrow$ KB.siblings$(u_j)$
        filter$(R,$ KB.type$(u_j))$
        $dist \leftarrow$ cosdist$(V[u_j], V[R])$
        for $r$ in argsort$(dist)[: N]$ do
            $A \leftarrow$ KB.aliases$(R[r])$
            $T \leftarrow$ replace$(c, e_j, a)$ for $a$ in $A$
            $\bar{C}$.add(rank_perplexity$(T)[0]$)
        end for
    end for
    return rank_contradiction$(c, \bar{C})[0]$
end function
```

We use top-k sampling to generate multiple claims, with k set to the number of noun chunks in the original source citance.[18]

7.4 Knowledge Base Informed Negations

ClaimGen-Entity and ClaimGen-BART only produce claims which are entailed by the source sentence. Additionally, we are interested in producing claim variants which are directly refuted by the original sentence, as these negations are needed when building fact checking datasets and for training fact checking models. Work in [239] created these negations manually, and some work has begun to explore automatically generating these negations for scientific claims [206].

[18] We use scispaCy to identify noun chunks

To this end, we leverage the availability of large curated biomedical knowledge bases to develop a principled approach to claim variant generation. In particular, we use the UMLS metathesaurus [33], which unifies hundreds of different ontologies in biomedicine, as a source of term replacements for negations.

We provide an overview of the KBIN algorithm in Algorithm 7.1 and Figure 7.2. KBIN works by first performing NER on an input claim c, obtaining entities $\{e_1, \ldots, e_n\} \in E$. For each entity e_j in E, we link the entity to its unique concept u_j in UMLS using the scispaCy entity linker. If the entity is linked, we select all concepts which are siblings to u_j in the concept hierarchy, and which have the same semantic type (e.g. "Clinical Drug"). We rank all selected concepts by their cosine distance to the entity concept using pre-trained UMLS concept vectors, retaining the top 20 closest concepts. For this, we use `cui2vec` [24], which contains pre-trained concept vectors for 108,477 concepts from UMLS trained on medical documents from diverse sources.

For each of the related concepts, we generate candidate claim variants by replacing the entity text in the original claim with the canonical name and aliases of the related concept from UMLS. We rank all replacement sentences by their perplexity using a pre-trained GPT-2 model [192], keeping the sentence with least perplexity for each replacement. Finally, from among these most fluent sentences, we select the replacement which maximizes the NLI prediction of *contradiction* with the original claim. For this, we use a RoBERTa model [148] pre-trained on MNLI [250].

7.5 Experiments

We investigate three primary research questions:

RQ1 Do automatically generated claims enable zero-shot scientific fact checking?

RQ2 What is the percentage of high-quality claims generated using our methods?

RQ3 How does KBIN compare with previous work for claim negation in terms of generating contradictions?

For **RQ1**, we use CLAIMGEN-ENTITY and CLAIMGEN-BART generated claims to train a fact checking model, evaluating on the SciFact dataset [239] and comparing to relevant baselines. To answer **RQ2** and **RQ3**, we design annotation criteria and perform manual evaluations with a group of expert annotators (details in §7.5.2).

7.5.1 RQ1: Fact Checking Performance

SciFact Task The SciFact fact verification task consists of: given a claim c and a corpus of scientific abstracts D, retrieve evidence abstracts from D, predict if the claim is *supported* or *refuted* by those documents or if there is *not enough information (NEI)* to make a prediction, and optionally determine what the rationale sentences are that explain the prediction. Here we focus on the oracle abstract setting of the task, in which gold abstracts are provided to the model and there is no retrieval component. This setup exists in the scientific fact checking literature [206], and allows us to focus on one component of the fact checking pipeline for evaluating the impacts of claim generation.

Creating Training Data for the Zero-shot Setting We require a set of claim-abstract pairs for training where the abstract either supports, refutes, or does not provide evidence for the given claim. We exploit citation relationships to generate claims paired with potential evidence, using citances from the CiteWorth dataset [256] as source citances for generation. *Supports* claims are produced by directly pairing a generated claim with the abstracts of documents cited by the source citance. For *refutes* claims, we negate a generated claim using KBIN and pair it with the same abstract. For claims labelled *NEI*, we pair the generated claim or negated claim with the abstract of the source document of the citance; the source document is related to the claim but presumably does not directly support or refute the claim given the need for a citation.

Experimental Setup In our experimental setup, we use LongChecker [240], a Longformer [27] model adapted for scientific fact checking. The model forms its input by concatenating a claim with its evidence abstract, inserting separator tokens between sentences, and uses a classification head to predict the veracity label from the representation of the `[CLS]` token.

We explore several different setups for our training data. As a baseline, we experiment with pre-training only on FEVER claims [230], which are general domain fact checking data based on Wikipedia. We also include an experiment where we manually tune a threshold for the prediction of *NEI* on the SciFact training data, as we saw that the model tends to overpredict this label without any fine-tuning on in-domain data. We also provide an upper bound on performance by fine-tuning on the in-domain train split of SciFact. Finally, we experiment with both CLAIMGEN-ENTITY and CLAIMGEN-BART as sources of training data generated from CiteWorth citances, pairing both with KBIN for negations. We note that though CLAIMGEN-BART requires manually re-written claims as training data for generating *supports* claims, it does not use any claims paired with evidence manually labelled for veracity, thus making it zero-shot for the SciFact fact-checking task. In all cases we test on the SciFact dev split. Hyperparameter information, including number of training instances, is given in §G.2.6, and code and data will be released upon paper acceptance. In all cases, results are reported as macro-F1.

Results Our results on SciFact are given in Table 7.1. With an upper bound of 77.70 F1, we see that a model fine-tuned on automatically generated claims is able to achieve within 90% of the performance of a model trained on in-domain manually written claims. This is also invariant to the method used to generate claims, as both CLAIMGEN-ENTITY and CLAIMGEN-BART produce similar results. Additionally, both methods provide significant gains over pre-training on

Method	P	R	F1
FEVER only	86.21	11.96	21.01
FEVER + thresh	69.15	66.51	67.80
SciFact (Upper Bound)	77.88	77.51	77.70
ClaimGen-entity	**72.86**	69.38	**71.08**
ClaimGen-BART	64.09	**79.43**	70.94

Table 7.1: Results for veracity prediction on the SciFact dataset using different sources of training data.

Metric	Labels
Fluency	3 - The claim contains no grammatical errors and its meaning can be understood 2 - The claim contains some grammatical errors but is still understandable 1- The claim contains many grammatical errors and cannot be understood
De-Contextualized	1 - The claim is interpretable on its own and requires no context; the addition of the original context does not alter the meaning of the claim 0 - The claim cannot be interpreted in a meaningful way without the original context
Atomicity	1 - The claim is about a single entity/process (atomic) 0 - The claim is non-atomic and can be broken down into multiple claims
Faithfulness	5 - The claim is correct and fully supported and complete with respect to the original sentence and context 4 - The claim is correct with respect to the original sentence and context but leaves out information from the original sentence and context 3 - The claim is related to the original sentence and does not contain incorrect information but is not explicitly stated in the original sentence 2 - The claim contains explicitly incorrect information relative to the original sentence and context 1 - The claim has nothing to do with the original sentence

Table 7.2: Claim quality evaluation metrics and their possible values

FEVER only, especially when no threshold on *NEI* claims is used but also when re-calibrating the model to predict *NEI* less often.

7.5.2 RQ2: Claim Quality Evaluation

Next, we explore if there are differences between our methods in terms of claim quality and the percentage of valid claims. For this, we ask three expert annotators to manually assess generated claims along a number of quality criteria. One annotator has undergraduate training in the life sciences and graduate training in computer science; the other two annotators have undergraduate training in the life sciences and materials science respectively. We define a set of criteria for evaluation, given in Table 7.2. These criteria are inspired by the AIDA (Atomic, Independent, Declarative, and Absolute) framework for scientific claims introduced in [133]. They are also based on similar human evaluation criteria used to assess generation quality for related tasks [209]. We develop an initial set of guidelines for the annotators and conduct two rounds of pilot annotations to improve instructions and increase agreement. For the final evaluation,

Method	Fluency	De-Con. (%)	Atomic (%)	Faithfulness	# Gen	# Accept	P
ClaimGen-Entity	2.51	55.63	**85.28**	3.54	893	111	12.43
ClaimGen-BART	**2.74**	**84.35**	80.65	**4.15**	156	69	**44.23**
α (236 claims)	82.74	64.53	58.71	53.01	-	-	-

Table 7.3: Average annotation score, agreement, and claim yield for each category. De-contextualized is only annotated if fluency > 1; atomicity and faithfulness are only annotated if fluency > 1 and de-contextualized == 1. # Gen are the total claims generated by the method, and # Accept are the number of acceptable claims generated.

Citance	Generated	Fl,D,A,Fa
Due to its geographic position and geological history, the island of Sardinia is characterized by a remarkable richness of endemic species and represents one of the most prominent biodiversity hotspots in the Mediterranean basin.	The island of Sardinia is characterized by a remarkable richness of endemic species.	3,1,1,5
Frequently reported symptom-eliciting chemicals and environmental agents include fragranted products, motor-vehicle exhaust fumes, cleaning agents, freshly printed papers or magazines, and smoke from wood burners.	Frequently reported symptom-eliciting chemicals and environmental agents are fragranted products.	3,1,1,5
The herbicide inhibits EPSPS (5-enolpyruvylshikimate-3-phosphate synthase) in the shikimate pathway, which has a key role in the biosynthesis of aromatic amino acids and is required for survival of the plant.	The herbicide inhibits EPSPS in the shikimate pathway.	3,1,1,5
Experimental models of OA, such as the intra-articular injection of monosodium acetate (MIA), are associated with joint pathology and pain behaviour comparable to clinical OA.	OA is associated with joint pathology and pain behaviour comparable to clinical OA.	3,1,0,4

Table 7.4: Sample generated claims with their ratings for (Fl)uency, (D)e-Contextualized, (A)tomicity, (Fa)ithfulness

we generate claims on a set of 100 citances sampled from the CiteWorth dataset [256], which contains citations in context for over 1M citances spanning 10 domains.

We limit the citances to those from papers in biology and medicine to match the domain of SciFact. Annotator agreement is measured as Krippendorff's α [131] on 236 claims for each category except fluency, where we measure the percentage of claims where all annotators agree.[19] The annotators then assess 1,049 total claims (including the 236 shared claims). Each annotator rates all criteria for an individual claim, starting with fluency, then de-contextualized, then atomicity, then faithfulness. We are mainly interested in claim quality and yield, so annotators only annotate "de-contextualized" if the claim is legible (fluency > 1), and only annotate "atomicity" and "faithfulness" if the claim is also de-contextualized (so one is able to discern meaning from the claim). This results in the following rules for acceptable claims based on the definitions for the labels in each category: Fluency $>$ 1 AND De-Contextualized $=$ 1 AND Atomicity $=$ 1 AND Faithfulness $>$ 3. An acceptable claim is thus legible, meaningful, represents a single

[19] Fluency agreement is measured in terms of agreement percentage as most ratings are the same (3), thus any disagreements have an oversized influence on α.

Method	R-1	R-2	R-L
Entity	47.12	27.63	42.30
BART	**56.58**	**40.12**	**53.38**

Table 7.5: ROUGE score between generated and manually written reference claims in the SciFact dataset

aspect of a scientific entity or process, and accurately reflects the information presented in the original citance.

The results of claim quality annotation are given in Table 7.3. Note that these are on claims generated by CLAIMGEN-ENTITY and CLAIMGEN-BART (see examples in Table 7.4), and thus are only *supports* claims. We first note that inter-annotator agreement is very high for fluency and moderate across all other criteria. Generated claims are quite fluent across methods, with a small minority of instances being illegible. Unsurprisingly, CLAIMGEN-BART improves over CLAIMGEN-ENTITY across all categories except for atomicity. This intuitively makes sense as CLAIMGEN-ENTITY directly produces claims which are about a single entity. CLAIMGEN-ENTITY yields a higher number of claims per citance as it generates one claim for every entity in the sentence, but the precision of acceptable claims is much lower than that of CLAIMGEN-BART. Thus, there is a tradeoff between the two methods between the number of claims generated and their acceptability. While higher yield could lead to higher coverage of claims in the original text, this study is left to future work.

Next, we examine the similarity between generated claims and manually written claims from SciFact. We generate claims for each source citance s_i in the SciFact dev split, and calculate the ROUGE score [145] between each generated claim $c_j^{(i)}$ and each manually written claim $d_k^{(i)}$. From this, we take an average of the max ROUGE score for each generated claim. Formally, given $|C|$ claims we calculate:

$$score = \frac{1}{|C|}\sum_i \sum_j \max_k \text{ROUGE}(c_j^{(i)}, d_k^{(i)})$$

Our evaluation results are given in Table 7.5. Both methods produce claims which have high overlap with the reference claims, though claims generated directly using BART are significantly closer to the reference claims than those generated using CLAIMGEN-ENTITY. Finally, we note the these scores are in the range of state-of-the-art models used for paraphrase generation, establishing a solid baseline for this task [270].

7.5.3 RQ3: Negation Evaluation

Finally, we perform a manual evaluation to compare KBIN against other methods of negation generation. Annotators evaluate negations based on Fluency and Entailment. We adopt the

Original Claim	Method	Generated Negation
Tonic signaling from the SCFV prevents constitutive stimulation.	Entity replace	Tonic signaling from the SCFV under care of respiratory physician (finding) constitutive stimulation.
	[206]	Tonic signaling from the inflammatory stimulation.
	KBIN	Tonic signaling from the SCFV accelerates constitutive stimulation.
Activation of the RAC1 homolog CED-10 kills viable cells in SRGP-1 mutant *Caenorhabditis Elegans*.	Entity replace	Activation of the LASS4 homolog CED-10 kills viable cells in SRGP-1 mutant *Caenorhabditis Elegans*.
	[206]	Activation of the RAC1 homolog CED-10 kills viable cells in SRGP-1 *Helicobacter Elegans*.
	KBIN	Activation of the RAC1 homolog CED-10 mediate viable cells in SRGP-1 mutant *Caenorhabditis Elegans*.

Table 7.6: Example negations generated using three methods. Span replacements are highlighted in red. In addition to replacing noun phrases, KBIN also has the ability to replace verb phrases as shown in these examples.

		Entailment		
Method	Fluency	3	2	1
Entity replace	83	1	81	1
[206]	83	10	64	9
KBIN	**93**	**15**	75	3

Table 7.7: Results for manual annotation of claim negations on 100 negations for each method. Fluent claims received annotations other than “SKIP”.

definitions used to annotate the SNLI corpus [37], in which the annotator is given an original claim (premise) and a generated negation (hypothesis) and asked to select from among the following options, including a SKIP option for Fluency:

3 The hypothesis is DEFINITELY FALSE given the premise

2 The hypothesis MIGHT BE TRUE given the premise

1 The hypothesis is DEFINITELY TRUE given the premise

SKIP The hypothesis contains a lot of grammatical errors and cannot be understood

We compare KBIN to two baselines. The first baseline replaces a single entity in the claim with a random entity of the same type, similar to the method in [176]. The second is the proposed negation generation method in [206]. The method is based on extracting keywords using YAKE [41] (an unsupervised method based on statistical text features), replacing those keywords using text infilling with a pre-trained language model, and selecting the replacement with the highest contradiction score using a model pre-trained for NLI. We generate negations for 100 claims using all three methods. For annotation, generated negations from all three methods are aggregated and the order of negation method randomized for each of the 100 claims.

Example negations generated by all three methods are given in Table 7.6 and annotation results for fluency and entailment are given in Table 7.7. First, KBIN produces more fluent claims than both baselines. Additionally, KBIN produces more convincing negations on average than both baselines. We observe that the most common operation performed by all three methods is to replace a noun phrase. KBIN has the benefit of being able to replace many entity types corresponding to concepts found in UMLS, which also include verb phrases that encode relations. Finally, KBIN improves over the baseline from [206] by producing fewer claims which are directly entailed by the source claim, i.e., that maintain the original meaning and do not negate the original claim.

7.5.4 Further Analysis

To give further insight into the quality of claims generated using our methods, we perform an experiment where we train and test models for scientific fact checking using claims only. This "claim-only" experiment helps us assess whether the negation process introduces data artifacts that can be leveraged by the model to predict veracity. We present results from training on claims generated using ClaimGen-BART and KBIN, compared against training on the original SciFact training data (which has manually written negations), along with random and majority baselines, in Figure 7.3.

We observe that there are likely some dataset artifacts in the original SciFact claims that lead to model performance well above the majority and random baselines.[20] This phenomenon has been observed in general domain natural language inference datasets as well [185]. Training on claims generated using our methods results in performance that is much more proximal to random performance on the SciFact dev set, indicating that the label-associated bias in the original training data is not present and a possible domain shift between the original SciFact claims and our generated claims. This can further explain some of the performance gap we observe between zero-shot fact-checking and the upper bound of training on manually labeled training data (Table 7.1).

7.6 Related Work

Scientific Fact Checking Our work follows a line of recent literature on scientific fact checking [239]. The goal of this task is to determine the veracity of claims related to scientific topics by retrieving appropriate documents from scientific literature, finding evidentiary sentences from those documents, and determining whether claims are supported, refuted, or there is not

[20] It is difficult to fully separate the contributions of data artifacts and model performance in this setting, i.e., there is no situation which guarantees *no* undesirable data artifacts. Performance ought to be better than a random baseline in this theoretical setting, due to the pretrained language model likely having had some exposure to the content of the claims during pretraining.

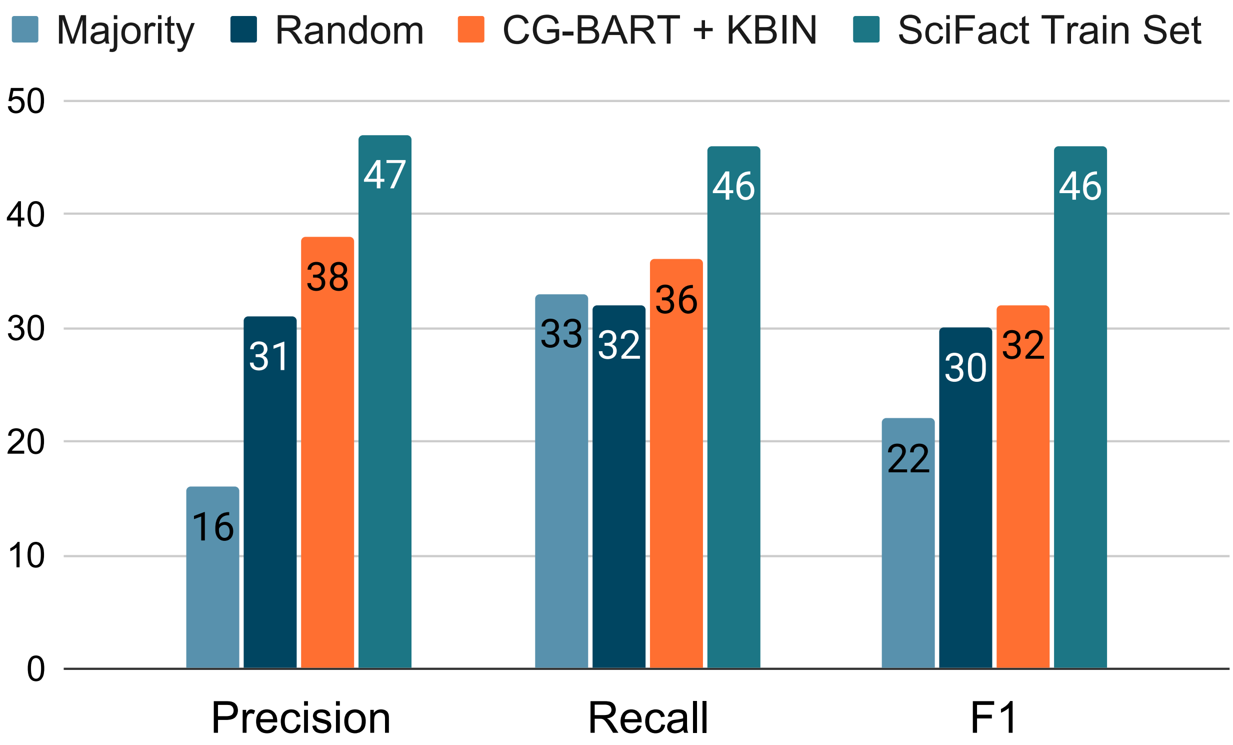


Figure 7.3: Fact checking performance of models trained only on claims (i.e. no evidence). Training on our generated claims result in performance closer to random (indicating fewer data artifacts) than training on the original SciFact claims.

enough evidence to make a judgement. The task closely resembles the task of general domain fact-checking [230, 13]. Well-performing systems on this task use large language models to perform neural document retrieval [189] or multi-task learning of rationale prediction and stance prediction [141, 240]. Recent work on general domain fact checking has also introduced methods for adversarial generation of claims which are particularly difficult to fact-check [232, 11], and for performing the task without any labeled data [176]. Our proposed methods extend zero-shot fact checking to the scientific domain, demonstrating that one can achieve 90% of the inference performance of state-of-the-art systems without domain-specific labeled data.

Generating Training Data Our work is also related to methods for the automatic generation of training data. Generation of synthetic data has been used for multiple tasks, for example question answering [68, 200], knowledge-base completion [207], and fact-checking [176]. Most similar to our setting, the COVID-Fact dataset [206] contains claims related to COVID-19 crawled from Reddit, and is constructed semi-automatically. Claims which are supported by evidence are extracted from Reddit and verified by human annotators, while negations of these claims are generated automatically via masked language model infilling. KBIN improves upon the negation method proposed in this work by leveraging in-domain structured knowledge via UMLS.

7.7 Conclusion

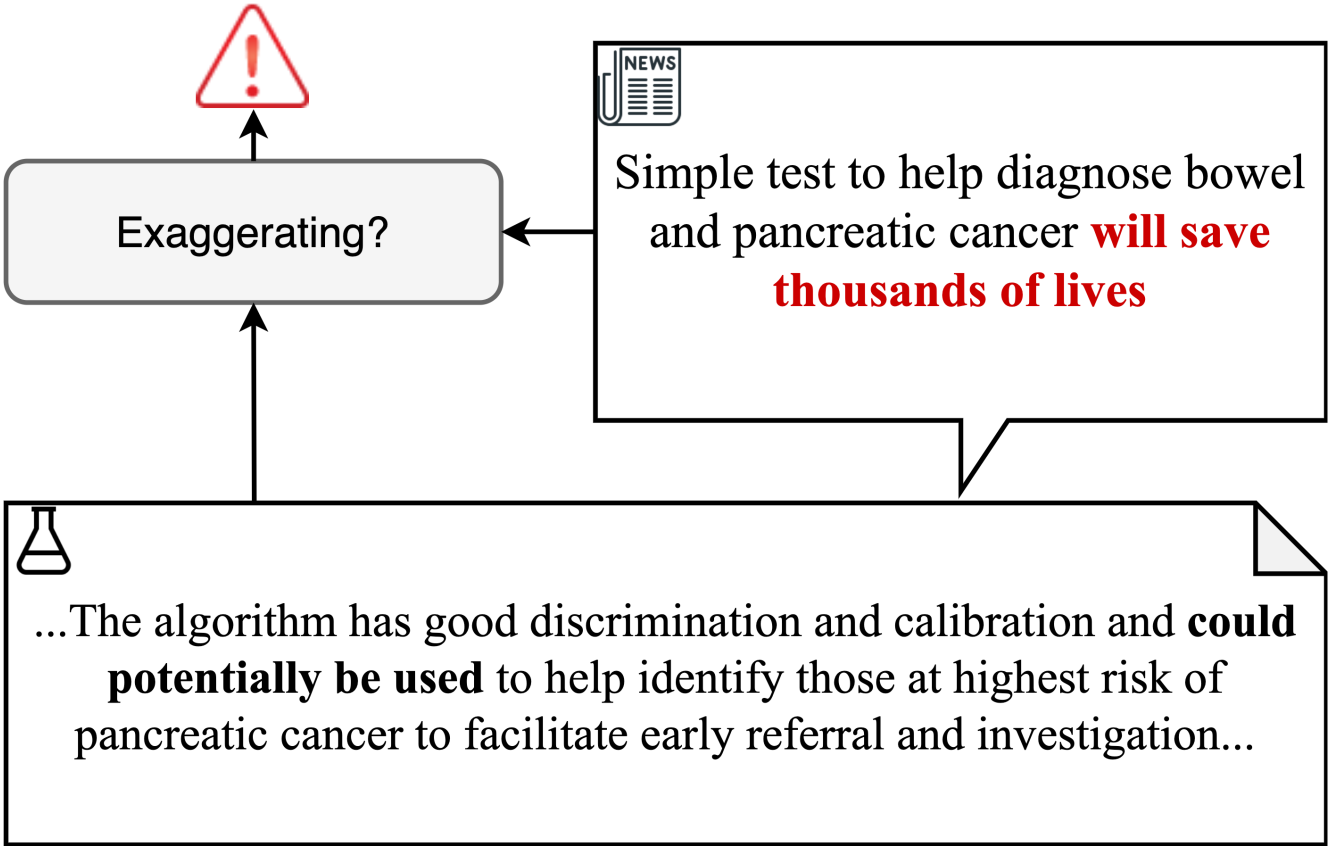
In this work, we propose the task of scientific claim generation, presenting ClaimGen-BART, ClaimGen-entity, and KBIN to perform the task. We demonstrate that generated claims can be used to train a model for zero-shot scientific fact checking and obtain within 90% of the performance of a model trained on human-written claims. Through a rigorous user study we demonstrate that ClaimGen-BART produces higher quality claims than ClaimGen-entity, and that KBIN produces more fluent and more convincing negations than previous work. Work remains to improve claim generation quality and assess the impacts of generated claims in other domains of science, as well as how generated claims can be used in the evidence retrieval component of fact checking systems. We hope that our methods will be used to facilitate future work by enabling faster creation of training datasets and improving the performance of models on the timely and important task of scientific fact checking.

Acknowledgements

This project is supported in part by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 801199, and by the United States National Science Foundation Grant OIA-2033558. We thank Doug Downey, Hannaneh Hajishirzi, the reviewers, and members of the Semantic Scholar research team for their valuable feedback.

Ethical Considerations

Automated scientific fact checking has great potential value to the scientific community, as well as for addressing phenomenon such as the propagation of scientific misinformation. Our aim in releasing models for scientific claim generation is to improve the generalizability of science fact checking systems in domains with less training resources. When training our fact checking models with generated or synthetic data, there are questions regarding the veracity of the generated data and whether a model trained on inferred labels could produce trustworthy judgments. We hope that by introducing this task and models, we will enable the community to study such questions, while contributing to data curation in a domain in which such curation would normally require significant manual efforts and cost.



Exaggerating?
NEWS
Simple test to help diagnose bowel and pancreatic cancer will save thousands of lives
...The algorithm has good discrimination and calibration and could potentially be used to help identify those at highest risk of pancreatic cancer to facilitate early referral and investigation...

Figure 8.1: Scientific exaggeration detection is the problem of identifying when a news article reporting on a scientific finding has exaggerated the claims made in the original paper. In this work, we are concerned with predicting exaggeration of the main finding of a scientific abstract as reported by a press release.

8 Semi-Supervised Exaggeration Detection of Health Science Press Releases

8.1 Introduction

Factual and honest science communication is important for maintaining public trust in science [168, 164], and the "dominant link between academia and the media" are press releases about scientific articles [224]. However, multiple studies have demonstrated that press releases have a significant tendency to sensationalize their associated scientific articles [224, 38, 253, 252]. In this paper, we explore how natural language processing can help identify exaggerations of scientific papers in press releases.

While [224] and [38] performed manual analyses to understand the prevalence of exaggeration in press releases of scientific papers from a variety of sources, recent work has attempted to expand this using methods from NLP [265, 266, 143]. These works focus on the problem of automatically detecting the difference in the strength of causal claims made in scientific articles and press releases. They accomplish this by first building datasets of main claims taken from PubMed abstracts and (unrelated) press releases from EurekAlert[21] labeled for their strength. With this, they train machine learning models to predict claim strength, and analyze unlabelled data using these models. This marks an important first step toward the goal of automatically identifying exaggerated scientific claims in science reporting.

However, existing work has only partially attempted to address this task using NLP. Particularly, there exists no standard benchmark data for the exaggeration detection task with **paired** press releases and abstracts i.e. where the data consist of tuples of the form (press release, abstract) and the press release is written about the paired scientific paper. Collecting paired data labeled for exaggeration is critical for understanding how well any solution performs on the task, but is challenging and expensive as it requires domain expertise [224]. The focus of this work is then to curate a standard set of benchmark data for the task of scientific exaggeration detection, provide a more realistic task formulation of the problem, and develop methods effective for solving it using limited labeled data. To this end, we present MT-PET, a multi-task implementation of Pattern Exploiting Training (PET, [213, 214]) for detecting exaggeration in health science press releases. We test our method by curating a benchmark test set of data from the expert annotated data of [224] and [38], which we release to help advance research on scientific exaggeration detection.

Contributions In sum, we introduce:

- A new, more realistic task formulation for scientific exaggeration detection.
- A curated set of benchmark data for testing methods for scientific exaggeration detection consisting of 563 press release/abstract pairs.
- MT-PET, a multi-task extension of PET which beats strong baselines on scientific exaggeration detection.

8.2 Problem Formulation

We first provide a formal definition of the problem of scientific exaggeration detection, which guides the approach described in §8.3. We start with a set of document pairs $\{(t, s) \in \mathcal{D}\}$, where s is a source document (e.g. a scientific paper abstract) and t is a document written about the source document s (e.g. a press release for the paper). The goal is to predict a label $l \in \{0, 1, 2\}$ for a given document pair (t, s), where 0 implies the target document *undersells* source document, 1 implies the target document accurately reflects the source document, and 2 implies the target document *exaggerates* the source document.

[21] https://www.eurekalert.org/

Two realizations of this formulation are investigated in this work. The first (defined as **T1**) is an *inference* task consisting of labeled document pairs used to learn to predict l directly. In other words, we are given training data of the form (t, s, l) and can directly train a model to predict l from both t and s. The second (defined as **T2**) is as a *classification* task consisting of a training set of documents $d \in \mathcal{D}'$ from **both** the source and the target domain, and a classifier is trained to predict the *claim strength* l' of sentences from these documents. In other words, we don't require **paired** documents (t, s) at train time. At test time, these classifiers are then applied to document pairs (t, s) and the predicted claim strengths (l'_s, l'_t) are compared to get the final label l. Previous work has used this formulation to estimate the prevalence of *correlation to causation* exaggeration in press releases [266], but have not evaluated this on paired labeled instances.

Following previous work [266], we simplify the problem by focusing on detecting when the *main finding* of a paper is exaggerated. The first step is then to identify the main finding from s, and the sentence describing the main finding in s from t. In our semi-supervised approach, we do this as an intermediate step to acquire unlabeled data, but for all labeled training and test data, we assume the sentences are already identified and evaluate on the sentence-level exaggeration detection task.

8.3 Approach

One of the primary challenges for scientific exaggeration detection is a lack of labeled training data. Given this, we develop a semi-supervised approach for few-shot exaggeration detection based on pattern exploiting training (PET, [213, 214]). Our method, multi-task PET (MT-PET, see Figure 8.2), improves on PET by using multiple complementary cloze-style QA tasks derived from different source tasks during training. We first describe PET, followed by MT-PET.

8.3.1 Pattern Exploiting Training (PET)

PET [213] uses the masked language modeling objective of pretrained language models to transform a task into one or more cloze-style question answering tasks. The two primary components of PET are *patterns* and *verbalizers*. *Patterns* are cloze-style sentences which mask a single token e.g. in sentiment classification with the sentence "We liked the dinner" a possible pattern is: "We liked the dinner. It was `[MASK]`." *Verbalizers* are single tokens which capture the meaning of the task's labels in natural language, and which the model should predict to fill in the masked slots in the provided patterns (e.g. in the sentiment analysis example, the verbalizer could be `Good`).

Given a set of *pattern-verbalizer pairs (PVPs)*, an ensemble of models is trained on a small labeled seed dataset to predict the appropriate verbalizations of the labels in the masked slots. These models are then applied on unlabeled data, and the raw logits are combined as a weighted

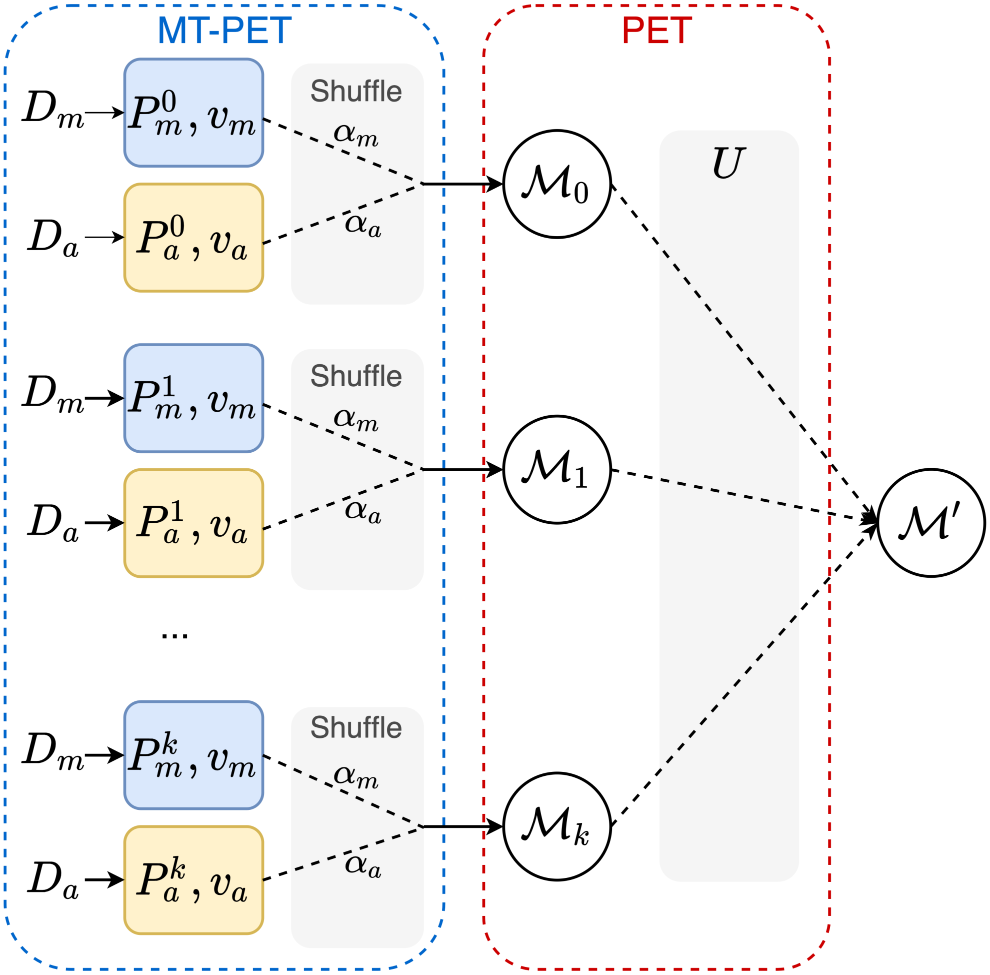


Figure 8.2: MT-PET design. We define pairs of complementary pattern-verbalizer pairs for a main task and auxiliary task. These PVPs are then used to train PET on data from both tasks.

average to provide soft-labels for the unlabeled data. A final classifier is then trained on the soft labeled data using a distillation loss based on KL-divergence.

8.3.2 Notation

We adopt the notation in the original PET paper [213] to describe MT-PET. In this, we have a masked language model $\mathcal{M}$ with a vocabulary V and mask token `[MASK]` $\in V$. A pattern is defined as a function $P(x)$ which transforms a sequence of input sentences $\mathbf{x} = (s_0, ..., s_{k-1}), s_i \in V^*$ to a phrase or sentence which contains exactly one mask token. Verbalizers $v(x)$ map a label in the task's label space $\mathcal{L}$ to a set of tokens in the vocabulary V which $\mathcal{M}$ is trained to predict.

For a given sample $\mathbf{z} \in V^*$ containing exactly one mask token and $w \in V$ corresponding to a word in the language model's vocabulary, $M(w|\mathbf{z})$ is defined as the unnormalized score that the language model gives to word w at the masked position in $\mathbf{z}$. The score for a particular label as given in [213] is then

$$s_{\mathbf{p}}(l|\mathbf{x}) = M(v(l)|P(\mathbf{x})) \tag{8.1}$$

For a given sample, PET then assigns a score s for each label based on all of the verbalizations of that label. When applied to unlabeled data, this produces soft labels from which a final model $\mathcal{M}'$ can be trained via distillation using KL-divergence.

8.3.3 MT-PET

In the original PET implementation, PVPs are defined for a single target task. MT-PET extends this by allowing for auxiliary PVPs from related tasks, adding complementary cloze-style QA tasks during training. The motivation for the multi-task approach is two-fold: 1) complementary cloze-style tasks can potentially help the model to learn different aspects of the main task; in our case, the similar tasks of exaggeration detection and claim strength prediction; 2) data on related tasks can be utilized during training, which is important in situations where data for the main task is limited.

Concretely, we start with a main task T_m with a small labeled dataset $(x_m, y_m) \in D_m$, where $y_m \in \mathcal{L}_m$ is a label for the instance, as well as an auxiliary task T_a with labeled data $(x_a, y_a) \in D_a, y_a \in \mathcal{L}_a$. Each pattern $P_m^i(x)$ for the main task has a corresponding complementary pattern $P_a^i(x)$ for the auxiliary task. Additionally, the labels in $\mathcal{L}_a$ have their own verbalizers $v_a(x)$. Thus, with k patterns, the full set of PVP tuples is given as

$$\mathcal{P} = \{((P_m^i, v_m), (P_a^i, v_a)) | 0 \leq i < k\}$$

Finally, a large set of unlabeled data U for the *main task only* is available. MT-PET then trains the ensemble of k masked language models using the pairs defined for the main and auxiliary task. In other words, for each individual model both the main PVP (P_m, v_m) and auxiliary PVP (P_a, v_a) are used during training.

For a given model $\mathcal{M}_i$ in the ensemble, on each batch we randomly select one task $T_c, c \in \{m, a\}$ on which to train. The PVP for that task is then selected as (P_c^i, v_c). Inputs (x_c, y_c) from that dataset are passed through the model, producing raw scores for each label in the task's label space.

$$s_{\mathbf{p}_c^i}(\cdot|\mathbf{x}_c) = \{\mathcal{M}_i(v_c(l)|P_c^i(\mathbf{x}_c))|\forall\, l \in \mathcal{L}_c\} \tag{8.2}$$

The loss is calculated as the cross-entropy between the task label y_c and the softmax of the score s normalized over the scores for all label verbalizations [213], weighted by a term α_c.

$$q_{\mathbf{p}_c^i} = \frac{e^{s_{\mathbf{p}_c^i}(\cdot|\mathbf{x}_c)}}{\sum_{l \in \mathcal{L}_c} e^{s_{\mathbf{p}_c^i}(l|\mathbf{x}_c)}} \tag{8.3}$$

$$L_c = \alpha_c * \frac{1}{N} \sum_n H(y_c^{(n)}, q_{\mathbf{p}_c^i}^{(n)}) \tag{8.4}$$

where N is the batch size, n is a sample in the batch, H is the cross-entropy, and α_c is a

Name	Pattern
$P_T^0(x)$	Scientists claim a. || Reporters claim b.The reporters claims are `[MASK]`
$P_T^1(x)$	Academic literature claims a. || Popular media claims b. The media claims are `[MASK]`
$P_T^0(x)$	[Reporters|Scientists] say a. The claim strength is `[MASK]`
$P_T^1(x)$	[Academic literature|Popular media] says a. The claim strength is `[MASK]`

Table 8.1: Patterns for both **T1** (exaggeration detection) and **T2** (claim strength prediction)

Pattern	Label	Verbalizers
P_T^0	Downplays	preliminary, competing, uncertainties
	Same	following, explicit
	Exaggerates	mistaken, wrong, hollow, naive, false, lies
P_T^1	Downplays	hypothetical, theoretical, conditional
	Same	identical
	Exaggerates	mistaken, wrong, premature, fantasy, noisy, artifical
P_T^*	NA	sufficient, enough, authentic, medium
	Correlational	inferred, estimated, calculated, borderline, approximately, variable, roughly
	Cond. Causal	cautious, premature, uncertain, conflicting, limited
	Causal	touted, proven, replicated, promoted, distorted

Table 8.2: Verbalizers for PVPs from both **T1** and **T2**. Verbalizers are obtained using PETAL [212], starting with the top 10 verbalizers per label and then manually filtering out words which do not make sense with the given labels.

hyperparameter weight given to task c.

MT-PET then proceeds in the same fashion as standard PET. Different models are trained for each PVP tuple in $\mathcal{P}$, and each model produces raw scores $s_{\mathbf{p}_m^i}$ for all samples in the unlabeled data. The final score for a sample is then a weighted combination of the scores of individual models.

$$s(l|\mathbf{x}_u^j) = \sum_i w_i * s_{\mathbf{p}_m^i}(l|\mathbf{x}_u^j) \tag{8.5}$$

where the weights w_i are calculated as the accuracy of model $\mathcal{M}_i$ on the train set D_m before training. The final classification model is then trained using KL-divergence between the predictions of the model and the scores s as target logits.

Name	Pattern
$P_0(x)$	`[MASK]`: a
$P_1(x)$	`[MASK]` - a
$P_2(x)$	"`[MASK]`" statement: a
$P_3(x)$	a (`[MASK]`)
$P_4(x)$	(`[MASK]`) a
$P_5(x)$	[Type: `[MASK]`] a

Table 8.3: Patterns for conclusion detection.

Label	Verbalizers
0	Text
1	Conclusion

Table 8.4: Verbalizers for PVPs for conclusion detection.

8.3.4 MT-PET for Scientific Exaggeration

We use MT-PET to learn from data labeled for both of our formulations of the problem (**T1**, **T2**). In this, the first step is to define PVPs for exaggeration detection (**T1**) and claim strength prediction (**T2**).

To do this, we develop an initial set of PVPs and use PETAL [212] to automatically find verbalizers which adequately represent the labels for each task. We then update the patterns manually and re-run PETAL, iterating as such until we find a satisfactory combination of verbalizers and patterns which adequately reflect the task. Additionally, we ensure that the patterns between **T1** and **T2** are roughly equivalent. This yields 2 patterns for each task, provided in Table 8.1, and verbalizers given in Table 8.2. The verbalizers found by PETAL capture multiple aspects of the task labels, selecting words such as "mistaken," "wrong," and "artificial" for exaggeration, "preliminary" and "conditional" for downplaying, and multiple levels of strength for strength detection such as "estimated" (correlational), "cautious" (conditional causal), and "proven" (direct causal).

For unlabeled data, we start with unlabeled pairs of full text press releases and abstracts. As we are concerned with detecting exaggeration in the primary conclusions, we first train a classifier based on single task PET for conclusion detection using a set of seed data. The patterns and verbalizers we use for conclusion detection are given in Table 8.3 and Table 8.4. After training the conclusion detection model, we apply it to the press releases and abstracts, choosing the sentence from each with the maximum score $s_{\mathbf{p}}(1|\mathbf{x})$.

8.4 Data Collection

One of the main contributions of this work is a curated benchmark dataset for scientific exaggeration detection. Labeled datasets exist for the related task of claim strength detection in scientific abstracts and press releases [266, 265], but these data are from press releases and abstracts which are unrelated (i.e. the given press releases are not written about the given abstracts), making them unsuitable for benchmarking exaggeration detection. Given this, we curate a dataset of paired sentences from abstracts and associated press releases, labeled by experts for exaggeration based on their claim strength. We then collect a large set of unlabeled press release/abstract pairs useful for semi-supervised learning.

8.4.1 Gold Data

The gold test data used in this work are from [224] and [38], who annotate scientific papers, their abstracts, and associated press releases along several dimensions to characterize how press releases exaggerate papers. The original data consists of 823 pairs of abstracts and press releases. The 462 pairs from [224] have been used in previous work to test claim strength prediction [143], but the data, which contain press release and abstract conclusion sentences that are mostly paraphrases of the originals, are used as is.

We focus on the annotations provided for claim strength. The annotations consist of six labels which we map to the four labels defined in [143]. The labels and their meaning are given in Table 8.5. This gives a claim strength label l_ρ for the press release and l_γ for the abstract. The final exaggeration label is then defined as follows:

$$l_e = \begin{cases} 0 & l_\rho < l_\gamma \\ 1 & l_\rho = l_\gamma \\ 2 & l_\rho > l_\gamma \end{cases}$$

As the original abstracts in the study are not provided, we automatically collect them using the Semantic Scholar API.[22] We perform a manual inspection of abstracts to ensure the correct ones are collected, discarding missing and incorrect abstracts. Gold conclusion sentences are obtained by sentence tokenizing abstracts using SciSpaCy [169] and finding the best matching sentence to the provided paraphrase in the data using ROUGE score [145]. We then manually fix sentences which do not correspond to a single sentence from the abstract. Gold press release sentences are gathered in the same way from the provided press releases.

This results in a dataset of 663 press release/abstract pairs labeled for claim strength and exaggeration. The label distribution is given in Table 8.6. We randomly sample 100 of these instances as training data for few shot learning (**T1**), leaving 553 instances for testing. Additionally, we create a small training set of 1,138 sentences labeled for whether or not they are the main conclusion sentence of the press release or abstract. This data is used in the first step of MT-PET to identify conclusion sentences in the unlabeled pairs.

[22] `https://api.semanticscholar.org/`

[224]	Description	[143]	Description
0	No relationship mentioned	-	-
1	Statement of no relationship	0	Statement of no relationship
2	Statements of correlation	1	Statement of correlation
3	Ambiguous statement of relationship		
4	Conditional statement of causation	2	Conditional statement of causation
5	Statement of “can”		
6	Statements of causation	3	Statement of causation

Table 8.5: Claim strength labels and their meaning from the original data in [224] and [38] and the mappings to the labels from [143]. We use the labels from [143] in this study, including for deriving the exaggeration labels.

Label	Count
Downplays	113
Same	406
Exaggerates	144

Table 8.6: Number of labels per class for benchmark exaggeration detection data.

For **T2** we use the data from [266, 265]. [265] create a dataset of 3,061 conclusion sentences labeled for claim strength from structured PubMed abstracts of health observational studies with conclusion sections of 3 sentences or less. [266] then annotate statements from press releases from EurekAlert. The selected data are from the title and first two sentences of the press releases, as [224] note that most press releases contain their main conclusion statements in these sentences, following an inverted pyramid structure common in journalism [187]. Both studies use the labeling scheme from [143] (see Table 8.5). The final data contains 2,076 labeled conclusion statements. From these two datasets, we select a random stratified sample of 4,500 instances for training in our full-data experiments, and subsample 200 for few-shot learning (100 from abstracts and 100 from press releases).

8.4.2 Unlabeled Data

We collect unlabeled data from ScienceDaily,[23] a science reporting website which aggregates and re-releases press releases from a variety of sources. To do this, we crawl press releases from ScienceDaily via the Internet Archive Wayback Machine[24] between January 1st 2016 and January 1st 2020 using Scrapy.[25] We discard press releases without paper DOIs and then pair each press release with a paper abstract by querying for each DOI using the Semantic Scholar API. This results in an unlabeled set of 7,741 press release/abstract pairs. Additionally, we use

[23] `https://www.sciencedaily.com/`

[24] `https://archive.org/web/`

[25] `https://scrapy.org/`

Method	P	R	F1
Supervised	28.06	33.10	29.05
PET	41.90	39.87	39.12
MT-PET	**47.80**	**47.99**	**47.35**

Table 8.7: Results for exaggeration detection with paired conclusion sentences from abstracts and press releases (**T1**). MT-PET uses 200 sentences for strength classification, 100 each from press releases and abstracts.

only the title, lead sentence, and first three sentences of each press release.

8.5 Experiments

Our experiments are focused on the following primary research questions:

- **RQ1**: Does MT-PET improve over PET for scientific exaggeration detection?
- **RQ2**: Which formulation of the problem leads to the best performance?
- **RQ3**: Does few-shot learning performance approach the performance of models trained with many instances?
- **RQ4**: What are the challenges of scientific exaggeration prediction?

We experiment with the following model variants:

- **Supervised**: A fully supervised setting where only labeled data is used.
- **PET**: Standard single-task PET.
- **MT-PET**: We run MT-PET with data from one task formulation as the main task and the other formulation as the auxiliary task.

We perform two evaluations in this setup: one with **T1** as the main task and one with **T2**. For **T1**, we use the 100 expert annotated instances with paired press release and abstract sentences labeled for exaggeration (200 sentences total). For **T2**, we use 100 sentences from the press data from [266] and 100 sentences from the abstract data in [265] labeled for claim strength. We use RoBERTa base [148] from the HuggingFace Transformers library [251] as the main model, and set α_m to be 1, and $\alpha_a = \text{min}(2, \frac{|D_m|}{|D_a|})$. All methods are evaluated using macro-F1 score, and results are reported as the average performance over 5 random seeds.

8.5.1 Performance Evaluation

We first examine the performance with **T1** as the base task (see Table 8.7). In a purely supervised setting, the model struggles to learn and mostly predicts the majority class. Basic PET yields a substantial improvement of 10 F1 points, with MT-PET further improving upon this by another 8 F1 points. Accordingly, we conclude that training with auxiliary task data provides much benefit for scientific exaggeration detection in the **T1** formulation.

Method	|**T2**|,|**T1**|	P	R	F1	Press F1	Abstract F1
Supervised	200,0	49.28	51.07	49.03	54.78	59.41
PET	200,0	55.76	58.58	56.57	63.56	62.76
MT-PET	200,100	**56.68**	**60.13**	**57.44**	**64.72**	**63.27**
Supervised	4500,0	58.20	59.99	58.66	63.26	**67.26**
PET	4500,0	59.53	61.84	60.45	**64.20**	64.92
MT-PET	4500,100	**60.09**	**62.68**	**61.11**	63.93	64.69
PET+in domain MLM	200,100	*57.18*	*60.12*	*58.06*	*64.29*	*62.69*
PET+in domain MLM	4500,100	*59.87*	*62.33*	*60.85*	*64.10*	*64.73*

Table 8.8: Results on exaggeration detection via strength classification (**T2**) with varying numbers of instances. MT-PET uses 100 instances from paired press and abstract sentences (200 sentences total).

We next examine performance with **T2** (strength classification) as the main task in both few-shot and full data settings (see Table 8.8). In terms of base performance, the model can predict exaggeration better than **T1** in a purely supervised setting. For PET and MT-PET, we see a similar trend; with 200 instances for **T2**, PET improves by 7 F1 points over supervised learning, and MT-PET improves on this by a further 0.9 F1 points. Additionally, MT-PET improves performance on the individual tasks of predicting the claim strength of conclusions in press releases and scientific abstracts with 200 examples. While less dramatic, we still see gains in performance using PET and MT-PET when 4,500 instances from **T2** are used, despite the fact that there are still only 100 instances from **T1**. We also test if the improvement in performance is simply due to training on more in-domain data ("PET+in domain MLM" in Table 8.8). We observe gains for exaggeration detection using masked language modeling on data from **T1**, but MT-PET still performs better at classifying the strength of claims in press releases and abstracts when 200 training instances from **T2** are used.

RQ1 Our results indicate that MT-PET does in fact improve over PET for both training setups. With **T1** as the main task and **T2** as the auxiliary task, we see that performance is substantially improved, demonstrating that learning claim strength prediction helps produce soft-labeled training data for *exaggeration detection*. Additionally, we find that the reverse holds with **T2** as main task and **T1** as auxiliary task. As performance can also be improved via masked language modeling on data from **T1**, this indicates that some of the performance improvement could be due to including data closer to the test domain. However, our error analysis in subsubsection 8.5.2 shows that these methods improve model performance on different types of data.

RQ2 We find that **T2** is better suited for scientific exaggeration detection in this setting, however, with a couple of caveats. First, the final exaggeration label is based on expert annotations for claim strength, so clearly claim strength prediction will be useful in this setup. Additionally, the task may be more forgiving here, as only the direction needs to be correct and not necessarily

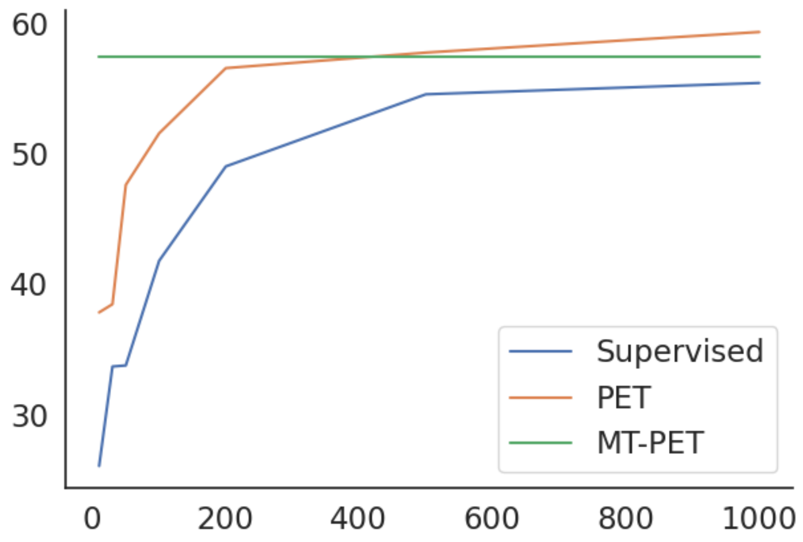


Figure 8.3: Learning curve for supervised learning and PET compared to performance of MT-PET using 200 instances from **T2** and 100 from **T1**.

the final strength label (i.e. predicting ‘0’ for the abstract and any of ‘1,’ ‘2,’ or ‘3’ for the press release label will result in an exaggeration label of ‘exaggerates’).

RQ3 We next examine the learning dynamics of our few-shot models with different amounts of training data (see Figure 8.3), comparing them to MT-PET to understand how well it performs compared to settings with more data. MT-PET with only 200 samples is highly competitive with purely supervised learning on 4,500 samples (57.44 vs. 58.66). Additionally, MT-PET performs at or above supervised performance up to 1000 input samples, and at or above PET up to 500 samples, again using only 200 samples from **T2** and 100 from **T1**.

8.5.2 Error Analysis

RQ4 Finally, we try to understand the difficulty of scientific exaggeration detection by observing where models succeed and fail (see Figure 8.4). The most difficult category of examples to predict involve direct causal claims, particularly exaggeration and downplaying when one document is a direct causal claim and the other an indirect causal claim (‘CON->CAU’, ‘CAU->CON’). Also, it is challenging to predict when both the press release and abstract conclusions are directly causal.

The models have the easiest time predicting when both statements involve correlational claims, and exaggerations involving correlational claims from abstracts. We also observe that MT-PET helps the most for the most difficult category: causal claims (see Figure G.1 in §G.1). The model is particularly better at differentiating when a causal claim in an abstract is

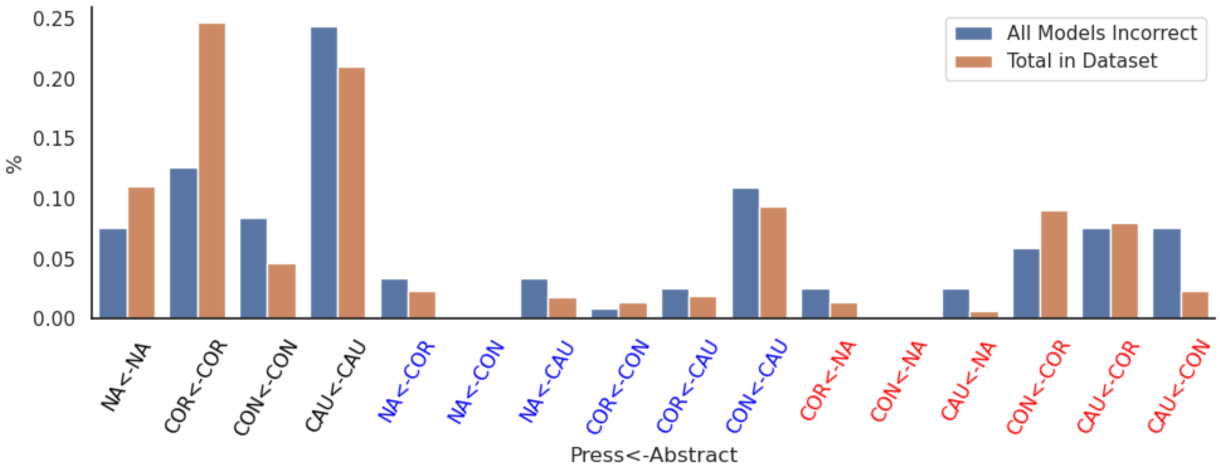


Figure 8.4: Proportion of examples by label which all models predict incorrectly.

downplayed by a press release. It is also better at identifying correlational claims than PET, where many claims involve association statements such as ‘linked to,’ ‘could predict,’ ‘more likely,‘ and ‘suggestive of.’

The model trained with MLM on data from **T1** also benefits causal statement prediction, but mostly for when both statements are causal, whereas MT-PET sees more improvement for pairs where one causal statement is exaggerated or downplayed by another (see Figure G.2 in §G.1). This suggests that training with the patterns from **T1** helps the model to differentiate direct causal claims from weaker claims, while MLM training mostly helps the model to understand better how direct causal claims are written. We hypothesize that combining the two methods would lead to mutual gains.

8.6 Related Work

8.6.1 Scientific Misinformation Detection

Misinformation detection focuses on a variety of problems, including fact verification [230, 13], check-worthiness detection [254, 166], stance [14, 21, 98] and clickbait detection [186]. While most work has focused on social media and general domain text, recent work has begun to explore different problems in detecting misinformation in scientific text such as SciFact [239] and CiteWorth [256], as well as related tasks such as summarization [63, 55].

Most work on scientific exaggeration detection has focused on flagging when the primary finding of a scientific paper has been exaggerated by a press release or news article [224, 38, 266, 265, 143]. [224] and [38] manually label pairs of press releases and scientific papers on a wide variety of metrics, finding that one third of press releases contain exaggerated claims, and 40% contain exaggerated advice. [143] is the first study into automatically predicting claim strength, using the data from [224] as a small labeled dataset. [265] and [266] extend this by building larger datasets for claim strength prediction, performing an analysis of a large set of unlabeled data to estimate the prevalence of claim exaggeration in press releases. Our work improves upon this by providing a more realistic task formulation of the problem, consisting of paired press releases and abstracts, as well as curating both labeled and unlabeled data to evaluate methods in this setting.

8.6.2 Learning from Task Descriptions

Using natural language to perform zero and few-shot learning has been demonstrated on a number of tasks, including question answering [191], text classification [190], relation extraction [36] and stance detection [99, 100]. Methods of learning from task descriptions have been gaining more popularity since the creation of GPT-3 [39]. [193] attempt to perform this with smaller language models by converting tasks into natural language and predicting tokens in the vocabulary. [213] propose PET, a method for few shot learning which converts tasks into cloze-style QA problems which can be solved by a pretrained language model in order to provide soft-labels for unlabeled data. We build on PET, showing that complementary cloze-style QA tasks can be trained on simultaneously to improve few-shot performance on scientific exaggeration detection.

8.7 Conclusion

In this work, we present a formalization of and investigation into the problem of scientific exaggeration detection. As data for this task is limited, we develop a gold test set for the problem and propose MT-PET, a semi-supervised approach based on PET, to solve it with limited training data. We find that MT-PET helps in the more difficult cases of identifying and differentiating direct causal claims from weaker claims, and that the most performant approach involves classifying and comparing the individual claim strength of statements from the source and target documents. The code and data for our experiments can be found online[26]. Future work should focus on building more resources e.g. datasets for exploring scientific exaggeration detection, including data from multiple domains beyond health science. Finally, it would be interesting to explore how MT-PET works on combinations of more general NLP tasks, such as question answering and natural language inference or part-of-speech tagging and named entity recognition.

[26] https://github.com/copenlu/scientific-exaggeration-detection

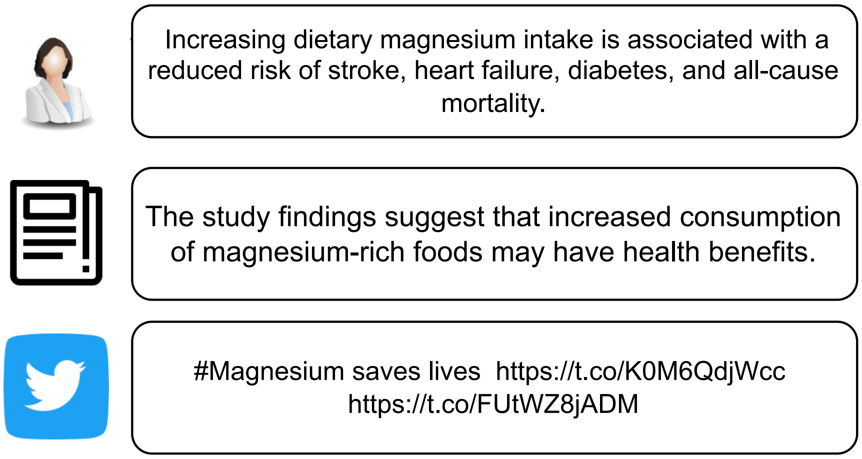
Acknowledgements

The research documented in this paper has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 801199.

Broader Impact Statement

Being able to automatically detect whether a press release exaggerates the findings of a scientific article could help journalists write press releases, which are more faithful to the scientific articles they are describing. We further believe it could benefit the research community working on fact checking and related tasks, as developing methods to detect subtle differences in a statement's veracity is currently understudied.

On the other hand, as our paper shows, this is currently still a very challenging task, and thus, the resulting models should only be applied in practice with caution. Moreover, it should be noted that the predictive performance results reported in this paper are for press releases written by science journalists – one could expect worse results for press releases which more strongly simplify scientific articles.



Increasing dietary magnesium intake is associated with a reduced risk of stroke, heart failure, diabetes, and all-cause mortality.

The study findings suggest that increased consumption of magnesium-rich foods may have health benefits.

#Magnesium saves lives https://t.co/K0M6QdjWcc https://t.co/FUtWZ8jADM

Figure 9.1: We are interested in measuring the information similarity of statements about scientific findings between different sources, including scientific papers, news, and tweets, shown here with real examples. The finding in this figure comes from [73] and the news quote is from [195].

9 Modeling Information Change in Science Communication with Semantically Matched Paraphrases

9.1 Introduction

Science communication disseminates scholarly information to audiences outside the research community, such as the public and policymakers [167]. This process usually involves translating highly technical language to non-technical, less-formal language that is engaging and easily understandable for lay people [210]. The public relies on the media to learn about new scientific findings, and media portrayals of science affect people's trust in science while at the same time influencing their future actions [95, 77, 135]. However, not all scientific communication accurately conveys the original information, as shown in Figure 9.1. Identifying cases where scientific information has changed is a critical but challenging task due to the complex translating and paraphrasing done by effective communicators. Our work introduces a new task of measuring scientific information change, and through developing new data and models aims to address the gap in studying faithful scientific communication.

Though efforts exist to track and flag when popular media misrepresent science,[27] the sheer volume of new studies, reporting, and online engagement make purely manual efforts both intractable and unattractive. Existing studies in NLP to help automate the study of science communication have examined exaggeration [257], certainty [179], and fact checking [34, 260], among others. However, these studies skip over the key first step needed to compare scientific texts for information change: automatically identifying content from both sources which describe

[27] See e.g. https://www.healthnewsreview.org/ and https://sciencefeedback.co/

the **same** scientific finding. In other words, to answer relevant questions about and analyze changes in scientific information at scale, one must first be able to point to which original information is being communicated in a new way.

To enable automated analysis of science communication, this work offers the following **contributions** (marked by **C**). First, we present the SCIENTIFIC PARAPHRASE AND INFORMATION CHANGE DATASET dataset (SPICED), a manually annotated dataset of paired scientific findings from news articles, tweets, and scientific papers (**C1**, §9.3). SPICED has the following merits: (1) existing datasets focus purely on semantic similarity, while SPICED focuses on differences in the *information* communicated in scientific findings; (2) scientific text datasets tend to focus solely on titles or paper abstracts, while SPICED includes sentences extracted from the full-text of papers and news articles; (3) SPICED is largely multi-domain, covering the 4 broad scientific fields that get the most media attention (namely: medicine, biology, computer science, and psychology) and includes data from the whole science communication pipeline, from research articles to science news and social media discussions.

In addition to extensively benchmarking the performance of current models on SPICED (**C2**, §9.4), we demonstrate that the dataset enables multiple downstream applications. In particular, we demonstrate how models trained on SPICED improve zero-shot performance on the task of sentence-level evidence retrieval for verifying real-world claims about scientific topics (**C3**, §9.5), and perform an applied analysis on unlabelled tweets and news articles where we show (1) media tend to exaggerate findings in the limitations sections of papers; (2) press releases and SciTech tend to have less informational change than general news outlets; and (3) organizations' Twitter accounts tend to discuss science more faithfully than verified users on Twitter and users with more followers (**C4**, §9.6).

9.2 Related Work

The analysis of scientific communication directly relates to fact checking, scientific language analysis, and semantic textual similarity. We briefly highlight our connections to these.

Fact Checking Automatic fact checking is concerned with verifying whether or not a given claim is true, and has been studied extensively in multiple domains [230, 13] including science [239, 34, 260]. Fact checking focuses on a specific type of information change, namely veracity. Additionally, the task generally assumes access to pre-existing knowledge resources, such as Wikipedia or PubMed, from which evidence can be retrieved that either supports or refutes a given claim. Our task is concerned with a more general type of information change beyond categorical falsehood and is a required task to complete prior to performing any kind of fact check.

Scientific Language Analysis Automating tasks beneficial for understanding changes in scientific information between the published literature and media is a growing area of research [257, 179, 34, 54, 17, 227, 236, 16, 86]. The three tasks most related to our work are understanding writing strategies for science communication [17], detecting changes in certainty [179], and detecting changes in causal claim strength i.e. exaggeration [257]. However, studying these requires access to paired scientific findings. To be able to do so at scale will require the ability to pair such findings automatically.

Semantic Similarity The topic of semantic similarity is well-studied in NLP. Several datasets exist with explicit similarity labels, many of which come from SemEval STS shared tasks (e.g. [44]) and paraphrasing datasets [81]. It is possible to build unlabelled datasets of semantic similarity automatically, which is the main method that has been used for scientific texts [49, 149]. However, such datasets fail to capture more subtle aspects of similarity, particularly when the focus is solely on the scientific findings conveyed by a sentence (see §H.1). And as we will show, approaches based on these datasets are insufficient for the task we are concerned with in this work, motivating the need for a new resource.

9.3 SPICED

We introduce SPICED, a new large-scale dataset of *scientific findings* paired with how they are communicated in news and social media. Communicating scientific findings is known to have a broad impact on public attitudes [249] and to influence behavior, e.g., the way vaccines are framed in the media has an effect on vaccine uptake [135]. Building upon prior work in NLP [256, 179, 224, 38], we define a scientific finding as **a statement that describes a particular research output of a scientific study, which could be a result, conclusion, product, etc.** This general definition holds across fields; for example, many findings from medicine and psychology report on effects on some dependent variable via manipulation of an independent variable, while in computer science many findings are related to new systems, algorithms, or methods. Following, we describe how the pairs of scientific findings were selected and annotated.

9.3.1 Data Collection

An initial dataset of unlabelled pairs of scientific communications was collected through Altmetric (`https://www.altmetric.com/`) a platform tracking mentions of scientific articles online. This initial pool contains 17,668 scientific papers, 41,388 paired news articles, and 733,755 tweets—note that a single paper may be communicated about multiple times. The scientific findings were extracted in different ways for each source. Similar to [188], we fine-tune a RoBERTa [148] model to classify sentences into methods, background, objective, results and conclusions using

200K paper abstracts from PubMed that had been self-labeled with these categories [42]. This sentence classifier attained 0.92 F1 score on a held-out 10% sample (details in §H.9) and then the classifier was applied to each sentence of the news stories and paper fulltexts. Given the domain difference between scientific abstracts and news, we additionally manually annotated a sample of 100 extracted conclusions; we find that the precision of the classifier is 0.88, suggesting that it is able to accurately identify scientific findings in news as well. We extract each sentence classified as “result” or “conclusion” and create pairs with each finding sentence from news articles written about it. This yields 45.7M potential pairs of ⟨news, paper⟩ findings. For tweets, we take full tweets as is, yielding 35.6M potential pairs of ⟨tweet, paper⟩ findings.

9.3.2 Data sampling

Pairing every finding from a news story with every finding from its matched paper results in an untenable amount of data to annotate. Additionally, it has been shown that proper data selection can reduce the need to annotate every possible sample [154, 108, 109]. Therefore, to obtain a sample of paired findings covering a range of similarities, we first filter our pool of unlabelled matched findings based on the semantics using Sentence-BERT (SBERT, [197]), a Siamese BERT network trained for semantic text similarity, trained on over 1B sentence pairs (see §H.7 for further details). We use this model to score pairs of findings from news articles and papers based on their embeddings’ cosine similarity and conduct a pilot study to determine which data to annotate.

For the pilot, we sample 400 pairs evenly for every 0.05 increment bucket in the range $[0, 1]$ of similarity scores (20 per bucket). Each sample is annotated by two of the authors of this study with a binary label of “matching” vs “not matching”, yielding a Krippendorff’s alpha of 0.73.[28] From this sample, we observed that there were no matches below 0.3 and only 2 ambiguous matches below 0.4. At the same time, the vast majority of samples from the entire dataset have a similarity score of less than 0.4. Additionally, above 0.9 we saw that each pair was essentially equivalent. Given the distribution of matched findings across the similarity scale, in order to balance the number of annotations we can acquire, the yield of positive samples, and the sample difficulty, we sampled data as follows based on their cosine similarity:

- Below 0.4 = automatically unmatched.
- Above 0.9 with a Jaccard index above 0.5 = automatically matched.
- Sample an equal number of pairs from each 0.05 increment bin between 0.4 and 0.9 for human expert annotation.

We sample 600 ⟨news, paper⟩ finding pairs from the four fields which receive the most media attention (medicine, biology, computer science, and psychology) using this method. This yields

[28] Note that many discussions about what constitutes matching vs. not matching were had in pilot work, leading to high agreement.

2,400 pairs to be annotated. For extensive details on the pilot annotation and visualizations, see §H.2.

We follow a similar procedure to sample pairs from papers and Twitter for annotation. However, rather than use the SBERT similarity scores, we instead first obtain annotations for news pairs using the scheme to be described later in §9.3.3 in order to train an initial model on our task (CiteBERT, [256]). We then use the trained model to obtain scores in the range [0,1] for each pair and sample an equal number of pairs from bins in 0.05 increments, for a total of 1,200 pairs (300 from each field of interest).

9.3.3 Finding Matching Annotation

We perform our final annotation based on the sampling scheme above using the Prolific platform (`https://www.prolific.co/`) as it allows prescreening annotators by educational background. We require each annotator to have at least a bachelor's degree in a relevant field to work on the task. Annotators are asked to label "whether the two sentences are discussing the same scientific finding" for 50 finding pairs with a 5-point Likert schema where each value indicates that "The information in the findings is..." (1): Completely different (2): Mostly different (3): Somewhat similar (4): Mostly the same, or (5): Completely the same. See §H.3 for details of how this rating scale was decided. We call this the INFORMATION MATCHING SCORE (IMS) of a pair of findings. Annotation was performed using POTATO [178]. Full annotation instructions and details are listed in §H.4. Notably, annotators were instructed to mark how similar the information in the *findings* was, as opposed to how similar the sentences are. Further, they were instructed to ignore extraneous information like "The scientists show..." and "our experiments demonstrate...".

Post processing To improve the reliability of the annotations, we use MACE [110] to estimate the competence score of each annotator and removed the labels from the annotators with the lowest competence scores. We further manually examine pairs with the most diverse labels (standard deviation of ratings $>$1.2) and manually replace the outliers with our expert annotations. The overall Krippendoff's α is 0.52, 0.57, 0.53, and 0.52 for CS, Medicine, Biology, and Psychology respectively, indicating that the final labels are reliable. While many annotators considered the task challenging, our quality control strategies allow us to collect reliable annotations.[29] For all the annotated pairs, we average the ratings as the final similarity score. In addition to the 3,600 manually annotated pairs, we include an extra 2,400 automatically annotated pairs as determined in §9.3.2 (unmatched pairs get an IMS of 1, matched pairs get an IMS of 5), for a total of 6,000 pairs. Given that there can be multiple pairs from a single news-paper pair, to avoid overlaps between training and test sets, we split the dataset 80%/10%/10% based on the

[29] For example, one participant commented "It was pretty hard to consider both the statements and their context then comparing them for similarities, but i enjoyed it"

Paper finding	News Finding	Similarity Score	IMS
However, the consistency of the erythritol results in both the central adiposity and usual glycemia comparisons lends strength to the findings, and the cluster of metabolites has biological plausibility.	Young adults who exhibited central adiposity gain over the course of 35 weeks had plasma erythritol levels 15-times higher at baseline than those with stable adiposity over the same period.	0.88	1
Our results showed that most of the official adult-onset men began their antisocial activities during early childhood.	Beckley, who is in the department of psychology and neuroscience at Duke, said the adult-onset group had a history of anti-social behavior back to childhood, but reported committing relatively fewer crimes.	0.38	4.4

Table 9.1: Annotated information matching score (IMS) and the similarity score estimated by SBERT [197] for selected finding pairs from SPICED. These examples demonstrate that simple similarity scores may not reflect whether the two sentences are covering the same scientific finding.

paper DOI and balance across subjects. Further dataset details in §H.5

Selected Examples To highlight the difficulty of SPICED, we show a pair of samples from our final dataset in Table 9.1. The IMS is compared to the cosine similarity between embeddings produced by SBERT. For the first case, SBERT presumably picks up on similarities in the discussed topics, such as erythritol and its relationship to adiposity, but the paper finding is concerned with the consistency of results and its biological implications while the news finding explicitly mentions a relationship between erythritol and adiposity. The second case expresses the opposite effect; the news finding contains a lot of extraneous information for context, but one of the core findings it expresses is the same as the paper finding, giving it a high rating in SPICED.

Comparison with existing datasets To further characterize the difficulty of SPICED compared to existing datasets, we show the average normalized edit distance between matching pairs in SPICED, STSB [44], and SNLI [37] (see §H.6 for the calculation). STSB is a semantic text similarity dataset consisting of pairs of sentences scored with their semantic similarity, sourced from multiple SemEval shared tasks. SNLI is a natural language inference corpus, and consists of pairs of sentences labeled for if they entail each other, contradict each other, or are neutral. We calculated the mean normalized edit distance across all pairs of *matching* sentences in each dataset's training data; For SPICED and STSB, pairs are considered matching if their IMS or similarity score is greater than 3, respectively. For SNLI, pairs are considered matching if the label is "entailment".

We find that there is a much greater lexical difference between the matching pairs in SPICED

STSB	SNLI	SPICED	News	Tweets
0.401	0.631	**0.726**	*0.712*	*0.749*

Table 9.2: The average normalized edit distance between matching pairs for various datasets shows that SPICED includes more pairs that are lexically dissimilar. For SPICED and STSB, pairs are considered matching if their similarity score is greater than 3. For SNLI, pairs are considered matching if the label is “entailment”.

(0.726) than existing general domain paired text datasets (0.401 for STSB and 0.631 for SNLI). This gap between STSB and SPICED also emphasizes the difference between traditional semantic textual similarity tasks and the information change task we describe here. Within SPICED, Twitter pairs had a higher distance (0.749) than news pairs (0.712), suggesting stronger domain differences. For qualitative examples showing the difference between SPICED and STSB, see §H.1.

Relationship of SPICED to Fact Checking The task introduced by SPICED captures information change more broadly than veracity as in automatic fact checking, as the task is concerned with the degree to which two sentences describe the same scientific information—indeed, two similar sentences may describe the same information equally poorly. Our task is similar to the sentence selection stage in the fact checking pipeline, and we later demonstrate that models trained on SPICED data are useful for this task for science in §9.5. However, our task and annotation are agnostic to whether a pair of sentences entail one another. This is especially useful if one wants to compare how a particular finding is presented across different media. Fact-checking datasets are also explicitly constructed to contain claims which are about a single piece of information—SPICED is not restricted in this way, focusing on a more general type of information change beyond categorical falsehood. Finally, we note two more unique features of SPICED: 1) SPICED contains naturally occurring sentences, while fact checking datasets like FEVER and SciFact often contain manually written claims. 2) The combination of domains in SPICED is unique; sentences are paired between (news, science) and (tweets, science), and these pairings dont exist currently.

9.4 Scientific Information Change Models

We now use SPICED to evaluate models for estimating the IMS of finding pairs in two settings: zero-shot transfer and supervised fine-tuning.

9.4.1 Experimental setup

We use the following four models to estimate zero-shot transfer performance. **Paraphrase**: RoBERTa [148] pre-trained for paraphrase detection on an adversarial paraphrasing task [173].

We convert the output probability of a pair being a paraphrase to the range [1,5] for comparison with our labels. **Natural Language Inference (NLI)**: RoBERTa pre-trained on a wide range of NLI datasets [170]. The final score is the model's measured probability of entailment mapped to the range [1,5]. **MiniLM**: SBERT with MiniLM as the base network [244]; we obtain sentence embeddings for pairs of findings and measure the cosine similarity between these two embeddings, clip the lowest score to 0, and convert this score to the range [1,5]. Note that this model was trained on over 1B sentence pairs, including from scientific text, using a contrastive learning approach where the embeddings of sentences known to be similar are trained to be closer than the embeddings of negatively sampled sentences. SBERT models represent a very strong baseline on this task, and have been used in the context of other matching tasks for fact checking including detecting previously fact-checked claims [216]. **MPNet**: The same setting and training data as MiniLM but with MPNet as the base network [220].

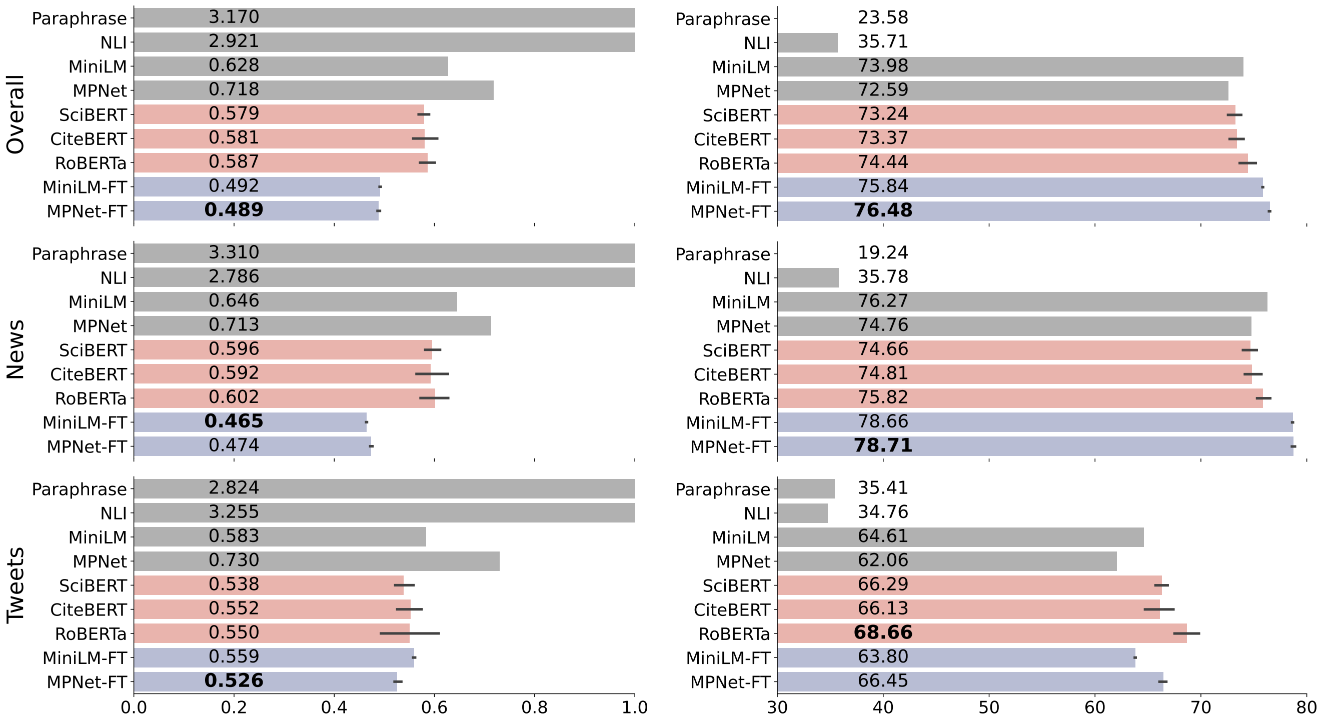
We fine-tune the following six models on SPICED to estimate IMS as a comparison with zero-shot transfer.

- **MiniLM-FT**: The same MiniLM model from the zero-shot transfer setup but further fine-tuned on SPICED. The training objective is to minimize the distance between the IMS and the cosine similarity of the output embeddings of the pair of findings.
- **MPNet-FT**: The same setup as MiniLM-FT but using MPNet as the base network.
- **RoBERTa**: The RoBERTa [148] base model; We perform a regression task where the model is trained to minimize the mean-squared error between the prediction and IMS.
- **SciBERT**: A transformer model trained using masked language modeling on a large corpus of scientific text [26]. The fine-tuning setup is the same as for the RoBERTa model.
- **CiteBERT**: A SciBERT model further fine-tuned on the task of citation detection, and was shown to have improved performance on downstream tasks using scientific text [256]. The training setup is the same as for the RoBERTa model.

Please see §H.7 for further details on the models and pretraining methods. For the fine-tuned models, we train on the entire training set of SPICED, including both news findings and tweets. For the test set we only use manually annotated pairs. Performance is measured in terms of mean-squared error (MSE) and Pearson correlation (r) (definitions of all metrics in §H.6). All results are reported as the average and standard deviation for each model across 5 random seeds.

9.4.2 Results

Paraphrase detection and natural language inference models perform very poorly for zero-shot transfer on this task (Figure 9.2, grey bars), with NLI having slightly better transfer, supporting our hypothesis that transferring from existing tasks to this domain is challenging. Fine-tuned models with Masked Language Model (MLM) pretraining can learn the task decently well (Figure 9.2,



(a) Mean Squared Error (b) Pearson correlation (r)

Figure 9.2: (a) Mean Squared Error (MSE, ↓ better) and (b) Pearson correlation (r, ↑ better) on the test set of SPICED. Grey = zero-shot transfer models, red = MLM models fine-tuned on SPICED, blue = SBERT models fine-tuned on SPICED. Results are averaged across 5 random seeds. Best results are given in bold.

red bars), but surprisingly RoBERTa performs just as well as SciBERT and CiteBERT which were specifically pretrained on scientific texts. We posit that this could be due to the fact that RoBERTa was pretrained on a wider range of texts that are reflective of the domains in SPICED, including news texts, while SciBERT and CiteBERT were trained solely on scientific papers.

SBERT models trained on large amounts of pretraining sentences perform well in the zero-shot transfer setup, with the MiniLM based model outperforming MPNet. The best setup was using SBERT fine-tuned on SPICED (Figure 9.2, blue bars), which yields up to 3.9 points gained overall in Pearson correlation and a reduction of 0.3 in terms of MSE (MPNet to MPNet-FT). We also note that there is a large gap between performance on this data and general semantic similarity datasets such as STSB, which see correlation scores in the 90s. As such, there is potentially much room to grow in terms of raw performance on this dataset.

Models performed worse for pairs with tweets versus those from news (Appendix Table H.4). This performance difference is in line with our expectations, as there is a large domain shift between tweets and scientific texts and our base models were not exposed to tweets during pre-training. All models, including the zero-shot transfer SBERT models, perform much worse on that split of the data. Additionally, we only see minor gains in performance in terms of MSE for MiniLM when fine-tuned on tweets. We see larger gains for MPNet. Interestingly, the

	CoVERT		COVID-Fact	
Method	MAP	MRR	MAP	MRR
BM25	$12.45_{0.00}$	$20.78_{0.00}$	$35.18_{0.00}$	$52.98_{0.00}$
MiniLM	$26.84_{0.00}$	$37.98_{0.00}$	$50.11_{0.00}$	$64.78_{0.00}$
+ FT	$\mathbf{28.23_{0.08}}$	$\mathbf{40.81_{0.16}}$	$52.66_{0.10}$	$66.91_{0.09}$
MPNet	$25.21_{0.00}$	$35.54_{0.00}$	$52.39_{0.00}$	$66.21_{0.00}$
+ FT	$26.84_{0.19}$	$37.65_{0.32}$	$\mathbf{53.61_{0.33}}$	$\mathbf{67.46_{0.28}}$

Table 9.3: Mean average precision (MAP) and mean reciprocal rank (MRR) for retrieval on the CoVERT and COVID-Fact datasets. All models are zero-shot i.e. without fine-tuning on the retrieval dataset.

best performance (Pearson r) for Tweets is RoBERTa, though the overall MSE is still best for MPNet-FT. We show extended benchmarking in §H.10 and the top-5 errors for RoBERTa and MPNet-FT in §H.11.

9.5 Application: Zero-Shot Evidence Retrieval for Scientific Fact Checking

Accurately measuring the similarity of scientific findings written in different domains enables a wide range of downstream analyses and tasks. As a first task, we consider evidence retrieval for scientific fact checking of real-world scientific claims. In general, automatic fact checking consists of retrieving relevant evidence for a given claim and predicting if that evidence supports or refutes the claim. We test the ability of models trained on SPICED to perform the evidence retrieval task in a zero-shot setting. In this, we use the models as is, with no further fine-tuning on any evidence retrieval data. We consider two fact checking datasets: CoVERT [163] is a dataset of scientific claims sourced from Twitter, mostly in the domain of biomedicine. We use the 300 claims and the 717 unique evidence sentences in the corpus in our experiment. COVID-Fact [206] is a semi-automatically curated dataset of claims related to COVID-19 sourced from Reddit. The corpus contains 4,086 claims with 3,219 unique evidence sentences.

Setup We compare different models' ability to rank the evidence sentences such that the ground truth evidence for a given claim is ranked highest. We use four models in a zero-shot setting for comparison (MiniLM, MiniLM-FT, MPNet, and MPNet-FT; '-FT' indicates fine-tuning on SPICED), and show results with the unsupervised BM25 [204], a widely used bag-of-words retrieval model. We report retrieval results in terms of mean average precision (MAP) and mean reciprocal rank (MRR), and average the results for models fine-tuned on SPICED across 5 random seeds.

Results We find that fine-tuning on SPICED provides consistent gains in retrieval performance on both datasets for both SBERT models (Table 9.3). This performance increase is encouraging, as there are two notable differences between SPICED and the two datasets in our experiment. The first is that the tasks are different: SPICED provides a general scientific information similarity task which proves to be useful for evidence sentence ranking. The second is that the domains are different: SPICED contains ⟨news, paper⟩ and ⟨tweet, paper⟩ pairs, while CoVERT and COVID-Fact have claims from Twitter and Reddit, respectively, paired with evidence in news. Our results show that training on SPICED improves the IR performance of the SBERT models, despite the domain and topic differences from our setting.

9.6 Application: Modeling Information Change in Science Communication

Whether the media faithfully communicate scientific information has long been a core question to the science community [167]. Our dataset and models allow us to conduct a large-scale analysis to study information change in science communication. Here, we focus on three research questions:

- **RQ1:** Do findings reported by different types of outlets express different degrees of information change from their respective papers?
- **RQ2:** Do different types of social media users systematically vary in information change when discussing scientific findings?
- **RQ3:** Which parts of a paper are more likely to be miscommunicated by the media?

RQ1-2 focus on the holistic information change captured in IMS, while RQ3 focuses on what types of information might be changing.

9.6.1 RQ1: Comparing Media Outlets

Different types of media target different audiences and tend to report the same issue differently [202, 160]. While good science journalism requires outlets to prioritize quality, in real practices, journalists may adopt different writing strategies for different types of audiences [205]. Thus, we investigate if findings reported by different types of outlets express different levels of information change, focusing on three types of outlets: General News (e.g., NYTimes), Press Releases (e.g., Science Daily), and Science & Technology (e.g., Popular Mechanics). We use our best-performing MPNet-FT model to estimate the IMS of over 1B pairs and keep those with IMS > 3, which finally leads to 1.1M paired findings from 26,784 news stories and 12,147 papers. We then build a linear mixed effect regression model [79] to predict IMS for matching pairs from news stories and research articles. We include a fixed effect for the type of news outlet, using General News as the reference category. To account for reporting differences across fields and variations specific to highly-publicized papers, we also include a fixed effect for the scientific

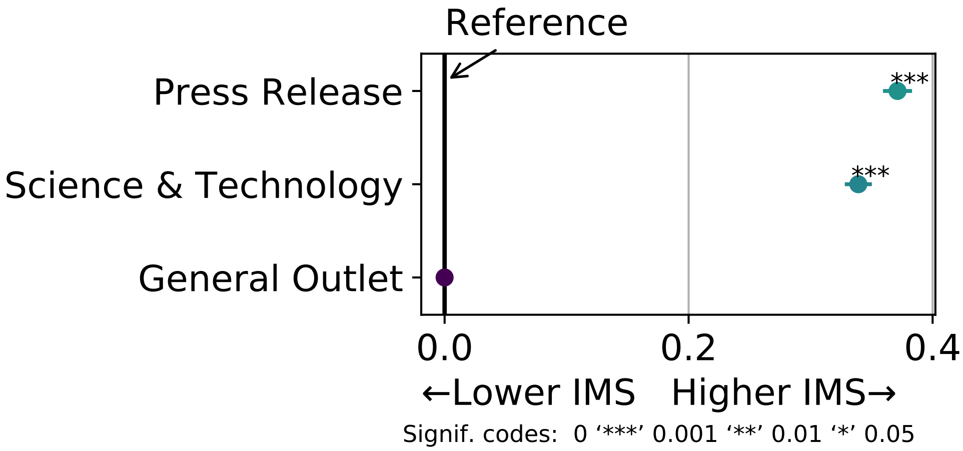


Figure 9.3: Scientific findings covered by Press Release and SciTech generally have less informational changes compared with findings presented in General Outlets

subject and a random effect for each paper with 30+ pairs (all other papers are pooled in a single random effect).

Results. Compared with General News, Science & Technology news outlets and Press Releases report findings that more closely match those from the original paper (Figure 9.3 shows the regression coefficients). This difference likely is due to some form of audience design where the journalist is writing for a more science-savvy readership in the latter two, whereas General News journalists must more heavily paraphrase the results for lay people.

9.6.2 RQ2: Comparing Social Media Accounts

Social media play an important role in disseminating scientific findings [268], so what factors affect the presentation of scientific information on social media becomes an important question. Here, we focus on the types of Twitter users who tweet about scientific findings. Based on 182K matched tweets and paper findings, we again build a linear mixed effect regression model to predict IMS. We include fixed effects of (1) if the account is run by an organization, as inferred using M3 [247], (2) if the account is verified (3) the number of followers and following, both log-transformed, and (4) the account age in years. We use the same field fixed effects and paper random effects as in RQ1.

Results The type of user strongly influences how faithful the tweets are to the original findings (Figure 9.4). Accounts from organizations tend to be more faithful to the original paper findings, which could be due to intentional actions of image management to build trust [208]. Surprisingly, verified accounts were far more likely to change information away from its original meaning; similarly, accounts with more followers had the same trend. Given their prominent roles in

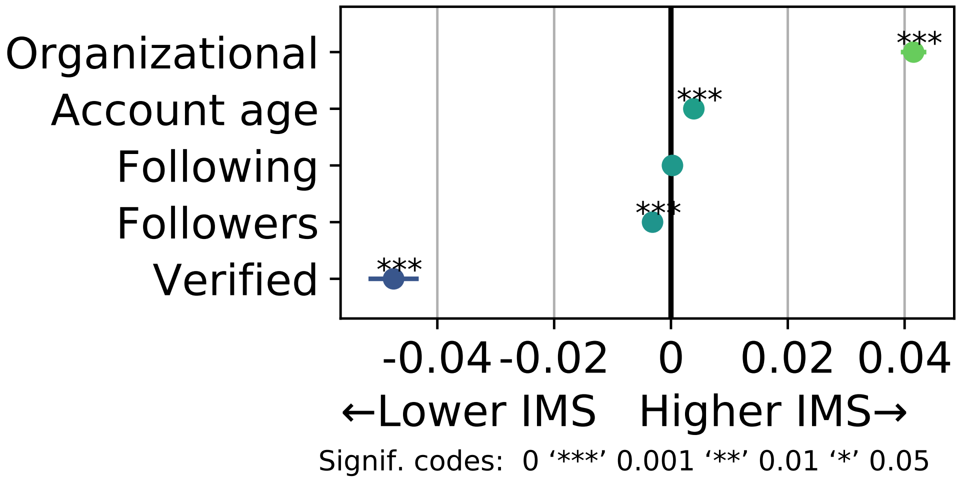


Figure 9.4: Organizational Twitter accounts keep more original information from the paper finding while verified users and those with more followers change more information when tweeting about a scientific finding.

Twitter communication [20, 103], multiple mechanisms may explain this gap such as adding more commentary or trying to translate original scientific findings to lay language to make the findings easier to understand. §H.12 shows the details of regression results.

9.6.3 RQ3: What Information Changes

Most studies on scientific misinformation focus on paper titles and abstracts [224], which cannot fully reflect the information presented in the full papers. Analyzing the information change of findings paired from all sections of papers could help to better understand the mechanisms behind scientific misinformation and develop strategies to reduce them. We use the same 1.1M finding pair dataset as RQ1 and analyze what information might have changed using two models trained for changes in scientific communication: identifying exaggerations [257] and certainty [179]. See §H.8 for more details on the exaggeration detection task.

Results Journalists tend to downplay the certainty and strength of findings from abstracts (Figure 9.5), mirroring the results of [179]. However, this pattern does not persist for findings in other parts of papers, especially the limitations. Existing studies suggest that journalists might fail to report the limitations of scientific findings [77], and our results here suggest that findings presented in limitations are more likely to be exaggerated and overstated. However, it is also possible that scientists may adopt different discourse strategies for different parts of a paper [47]. Nonetheless, our result obviates the necessity of analyzing the full text of a paper when studying science communication.

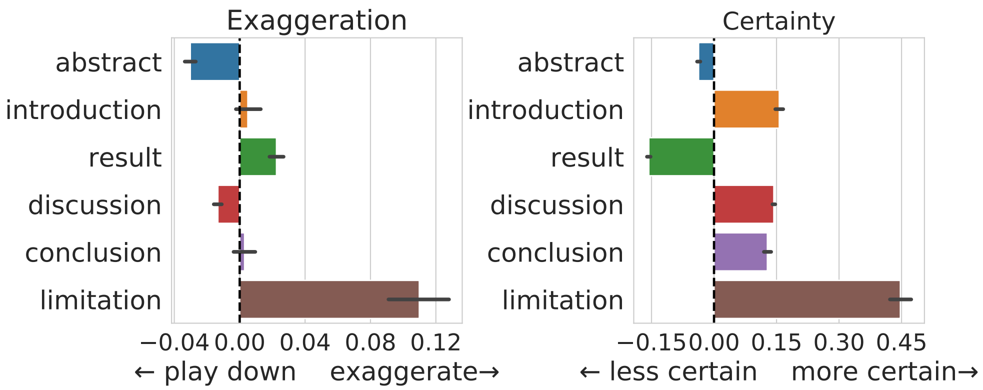


Figure 9.5: Journalists tend to downplay the certainty and strength of findings in abstracts, but overstate findings discussed in limitations sections.

9.7 Conclusion

Faithful communication of scientific results is critical for disseminating new information and establishing public trust in science. Given the challenge of—and occasional failures in—communicating science, new resources and models are needed to evaluate how science is reported. Here, we introduce SPICED, a new science communication paraphrases dataset labeled with information similarity. Extensive experiments demonstrate that models can predict the degree to which two reports of a scientific finding have the same information but that this is a challenging task even for current SOTA pre-trained language models. In downstream applications, we show SPICED improves model performance for evidence retrieval for scientific fact checking; and, using the trained model to perform a large-scale analysis of information change in science communication, we show systematic behaviors in how different people and news outlets faithfully convey scientific results. Data, code, and pretrained models are available at `http://www.copenlu.com/publication/2022_emnlp_wright/`.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 801199 and a Rackham Graduate Student Research Grant at the University of Michigan.

Limitations

We note three limitations of our study. Our data and analysis in social media is limited to only one platform, Twitter, and includes only tweets directly linked to the original paper, as indicated through Altmetric. While Twitter is among the largest social media platforms and is the most common in the Altmetric data, our data potentially omits other kinds of scientific communication about papers that do not directly link to a paper or tweets that link to a paper that cannot be easily identified to a DOI (e.g., linking to a PDF hosted on a personal website). Other types of tweets may be omitted from our dataset such as those written in a thread, or in a tweetorial, about a paper [84], which may include additional tweets that describe a paper's findings. While our models would likely still be able to effectively analyze such tweets, these additional forms of scientific communication could add new variety. We leave identifying and collecting such tweets to future work.

Second, our study focuses on only four large scientific fields. While these fields do cover a broad selection of papers, we were unable to annotate additional fields due to annotation budget and limitations from the Prolific platform. On Prolific, not all potential domains had sufficient numbers of qualified annotators (we required at least a Bachelor's degree in the domain) and the number of unique surveys to run scaled linearly with the number of domains, creating a significant human overhead. However, we will open source our annotation interface and pipeline and we encourage further efforts to build a larger dataset across more scientific domains.

Finally, while our models achieve moderately high performance at inferring the information matching (Figure 9.2), performance is not perfect, which potentially limits our ability in downstream models and tasks. While we show the data is still useful in training for related tasks (§9.5) and a trained model can be used to identify systematic behavior by types of users and outlets (§9.6), more accurate models would likely be needed to identify any trends for finer-grained settings, such as looking at the behavior of a specific outlet. For this reason,w e have kept our analyses at a higher level (e.g., outlet categories).

Ethics and Impacts

Miscommunication of scientific information can have negative impacts on many aspects of our society. Our study contributes to a large research program on the science of science communications [167]. Our dataset and model could be used to keep track of information change in science communication, enable large-scale analysis to understand the current science communication ecosystem, and finally help to facilitate better and more effective science communications.

Crowdsourcing ethics Annotating paired findings requires deep attention and may lead to annotator burnout. We carefully designed our annotation pipeline to provide a good annotation experience for the annotators. We designed a user-friendly Web-based annotation interface that allows annotators to do annotations using keyboard shortcuts. All the annotators are encouraged to leave comments and answer several questions about their annotation experience. More than 95% of the annotators are satisfied with their annotation experience and many people suggest that our study helps them to better understand the science communication process[30] and our annotation interface makes their task easier.[31]

[30] For example, one participant said "Nice learning experience, Helps to understand the news can be far more different then the research paper cited"

[31] For example, one participant said "i liked the option of using my keyboard, it made the experience more comfortable and efficient."

References

[1] ABULAISH, M., KUMARI, N., FAZIL, M., AND SINGH, B. A Graph-Theoretic Embedding-Based Approach for Rumor Detection in Twitter. In *IEEE/WIC/ACM International Conference on Web Intelligence* (2019), pp. 466–470.

[2] AHARONI, R., AND GOLDBERG, Y. Unsupervised Domain Clusters in Pretrained Language Models. In *ACL* (2020).

[3] ALLEIN, L., AUGENSTEIN, I., AND MOENS, M.-F. Time-Aware Evidence Ranking for Fact-Checking. *arXiv preprint arXiv:2009.06402* (2020).

[4] ALLEN-ZHU, Z., AND LI, Y. Towards Understanding Ensemble, Knowledge Distillation and Self-Distillation in Deep Learning. *CoRR abs/2012.09816* (2020).

[5] ALY, R., GUO, Z., SCHLICHTKRULL, M. S., THORNE, J., VLACHOS, A., CHRISTODOULOPOULOS, C., COCARASCU, O., AND MITTAL, A. FEVEROUS: fact extraction and verification over unstructured and structured information. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual* (2021), J. Vanschoren and S. Yeung, Eds.

[6] AMMAR, W., GROENEVELD, D., BHAGAVATULA, C., BELTAGY, I., CRAWFORD, M., DOWNEY, D., DUNKELBERGER, J., ELGOHARY, A., FELDMAN, S., HA, V., ET AL. Construction of the Literature Graph in Semantic Scholar. *NAACL HLT 2018* (2018), 84–91.

[7] ASHUKHA, A., LYZHOV, A., MOLCHANOV, D., AND VETROV, D. P. Pitfalls of In-Domain Uncertainty Estimation and Ensembling in Deep Learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020* (2020), OpenReview.net.

[8] ATANASOVA, P., BARRON-CEDENO, A., ELSAYED, T., SUWAILEH, R., ZAGHOUANI, W., KYUCHUKOV, S., MARTINO, G. D. S., AND NAKOV, P. Overview of the CLEF-2018 CheckThat! Lab on Automatic Identification and Verification of Political Claims. Task 1: Check-Worthiness. *arXiv preprint arXiv:1808.05542* (2018).

[9] ATANASOVA, P., SIMONSEN, J. G., LIOMA, C., AND AUGENSTEIN, I. A Diagnostic Study of Explainability Techniques for Text Classification. In *Proceedings of EMNLP* (2020), Association for Computational Linguistics.

[10] ATANASOVA, P., SIMONSEN, J. G., LIOMA, C., AND AUGENSTEIN, I. Generating fact checking explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020* (2020), D. Jurafsky,

J. Chai, N. Schluter, and J. R. Tetreault, Eds., Association for Computational Linguistics, pp. 7352–7364.

[11] ATANASOVA, P., WRIGHT, D., AND AUGENSTEIN, I. Generating Label Cohesive and Well-Formed Adversarial Claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Online, Nov. 2020), Association for Computational Linguistics, pp. 3168–3177.

[12] AUGENSTEIN, I., DAS, M., RIEDEL, S., VIKRAMAN, L., AND MCCALLUM, A. SemEval 2017 Task 10: ScienceIE-Extracting Keyphrases and Relations from Scientific Publications. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* (2017), pp. 546–555.

[13] AUGENSTEIN, I., LIOMA, C., WANG, D., CHAVES LIMA, L., HANSEN, C., HANSEN, C., AND SIMONSEN, J. G. 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)* (Hong Kong, China, 2019), Association for Computational Linguistics, pp. 4685–4697.

[14] AUGENSTEIN, I., ROCKTÄSCHEL, T., VLACHOS, A., AND BONTCHEVA, K. Stance Detection with Bidirectional Conditional Encoding. In *EMNLP* (2016), J. Su, X. Carreras, and K. Duh, Eds., The Association for Computational Linguistics, pp. 876–885.

[15] AUGENSTEIN, I., AND SØGAARD, A. Multi-Task Learning of Keyphrase Boundary Classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (Vancouver, Canada, July 2017), Association for Computational Linguistics, pp. 341–346.

[16] AUGUST, T., CARD, D., HSIEH, G., SMITH, N. A., AND REINECKE, K. Explain like I am a scientist: The linguistic barriers of entry to r/science. In *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020* (2020), R. Bernhaupt, F. F. Mueller, D. Verweij, J. Andres, J. McGrenere, A. Cockburn, I. Avellino, A. Goguey, P. Bjøn, S. Zhao, B. P. Samson, and R. Kocielnik, Eds., ACM, pp. 1–12.

[17] AUGUST, T., KIM, L., REINECKE, K., AND SMITH, N. A. Writing strategies for science communication: Data and computational analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Online, 2020), Association for Computational Linguistics, pp. 5327–5344.

[18] BA, J., AND CARUANA, R. Do Deep Nets Really Need to be Deep? In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing*

Systems 2014, December 8-13 2014, Montreal, Quebec, Canada (2014), Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 2654–2662.

[19] Badal, V. D., Wright, D., Katsis, Y., Kim, H.-C., Swafford, A. D., Knight, R., and Hsu, C.-N. Challenges in the construction of knowledge bases for human microbiome-disease associations. *Microbiome 7*, 1 (2019), 1–15.

[20] Bakshy, E., Hofman, J. M., Mason, W. A., and Watts, D. J. Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the Forth International Conference on Web Search and Web Data Mining, WSDM 2011, Hong Kong, China, February 9-12, 2011* (2011), I. King, W. Nejdl, and H. Li, Eds., ACM, pp. 65–74.

[21] Baly, R., Mohtarami, M., Glass, J. R., Màrquez, L., Moschitti, A., and Nakov, P. Integrating Stance Detection and Fact Checking in a Unified Corpus. In *NAACL-HLT (2)* (2018), M. A. Walker, H. Ji, and A. Stent, Eds., Association for Computational Linguistics, pp. 21–27.

[22] Barrón-Cedeño, A., Elsayed, T., Nakov, P., Da San Martino, G., Hasanain, M., Suwaileh, R., and Haouari, F. CheckThat! at CLEF 2020: Enabling the Automatic Identification and Verification of Claims in Social Media. In *European Conference on Information Retrieval* (2020), Springer, pp. 499–507.

[23] Barrón-Cedeño, A., Elsayed, T., Suwaileh, R., Màrquez, L., Atanasova, P., Zaghouani, W., Kyuchukov, S., Da San Martino, G., and Nakov, P. Overview of the CLEF-2018 CheckThat! Lab on Automatic Identification and Verification of Political Claims. Task 2: Factuality.

[24] Beam, A. L., Kompa, B., Schmaltz, A., Fried, I., Weber, G. M., Palmer, N. P., Shi, X., Cai, T., and Kohane, I. S. Clinical Concept Embeddings Learned from Massive Sources of Multimodal Medical Data. In *Pacific Symposium on Biocomputing 2020, Fairmont Orchid, Hawaii, USA, January 3-7, 2020* (2020), pp. 295–306.

[25] Bekker, J., and Davis, J. Learning from positive and unlabeled data: A survey. *arXiv preprint arXiv:1811.04820* (2018).

[26] Beltagy, I., Lo, K., and Cohan, A. SciBERT: A Pretrained Language Model for Scientific Text. 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)* (2019), pp. 3606–3611.

[27] Beltagy, I., Peters, M. E., and Cohan, A. Longformer: The Long-Document Transformer. *CoRR abs/2004.05150* (2020).

[28] BIRD, S. NLTK: The Natural Language Toolkit. In *ACL 2006, 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Sydney, Australia, 17-21 July 2006* (2006), N. Calzolari, C. Cardie, and P. Isabelle, Eds., The Association for Computer Linguistics.

[29] BIRD, S., DALE, R., DORR, B. J., GIBSON, B., JOSEPH, M. T., KAN, M.-Y., LEE, D., POWLEY, B., RADEV, D. R., AND TAN, Y. F. The ACL Anthology Reference Corpus: A Reference Dataset for Bibliographic Research in Computational Linguistics.

[30] BLITZER, J., DREDZE, M., AND PEREIRA, F. Biographies, Bollywood, Boom-Boxes and Blenders: Domain Adaptation for Sentiment Classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (2007), pp. 440–447.

[31] BLITZER, J., MCDONALD, R., AND PEREIRA, F. Domain Adaptation with Structural Correspondence Learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (Sydney, Australia, July 2006), Association for Computational Linguistics, pp. 120–128.

[32] BODE, L., AND VRAGA, E. K. See something, say something: Correction of global health misinformation on social media. *Health communication 33*, 9 (2018), 1131–1140.

[33] BODENREIDER, O. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Res. 32*, Database-Issue (2004), 267–270.

[34] BOISSONNET, A., SAEIDI, M., PLACHOURAS, V., AND VLACHOS, A. Explainable assessment of healthcare articles with QA. In *Proceedings of the 21st Workshop on Biomedical Language Processing, BioNLP@ACL 2022, Dublin, Ireland, May 26, 2022* (2022), D. Demner-Fushman, K. B. Cohen, S. Ananiadou, and J. Tsujii, Eds., Association for Computational Linguistics, pp. 1–9.

[35] BOJANOWSKI, P., GRAVE, E., JOULIN, A., AND MIKOLOV, T. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics 5* (2017), 135–146.

[36] BOURAOUI, Z., CAMACHO-COLLADOS, J., AND SCHOCKAERT, S. Inducing Relational Knowledge from BERT. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020* (2020), AAAI Press, pp. 7456–7463.

[37] BOWMAN, S. R., ANGELI, G., POTTS, C., AND MANNING, C. D. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical*

Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015 (2015), L. Màrquez, C. Callison-Burch, J. Su, D. Pighin, and Y. Marton, Eds., The Association for Computational Linguistics, pp. 632–642.

[38] Bratton, L., Adams, R. C., Challenger, A., Boivin, J., Bott, L., Chambers, C. D., and Sumner, P. The Association Between Exaggeration in Health-Related Science News and Academic Press Releases: A Replication Study. *Wellcome open research 4* (2019).

[39] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual* (2020), H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds.

[40] Button, K. S., Ioannidis, J., Mokrysz, C., Nosek, B. A., Flint, J., Robinson, E. S., and Munafò, M. R. Power failure: why small sample size undermines the reliability of neuroscience. *Nature reviews neuroscience 14*, 5 (2013), 365–376.

[41] Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., and Jatowt, A. YAKE! Keyword extraction from single documents using multiple local features. *Inf. Sci. 509* (2020), 257–289.

[42] Canese, K., and Weis, S. Pubmed: the bibliographic database. *The NCBI handbook 2*, 1 (2013).

[43] Carpenter, B. Multilevel Bayesian Models of Categorical Data Annotation. *Unpublished manuscript 17*, 122 (2008), 45–50.

[44] Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* (Vancouver, Canada, 2017), Association for Computational Linguistics, pp. 1–14.

[45] Chandrasekaran, M. K., Feigenblat, G., Hovy, E. H., Ravichander, A., Shmueli-Scheuer, M., and de Waard, A. Overview and insights from the shared tasks at scholarly document processing 2020: Cl-scisumm, laysumm and longsumm. In *Proceedings of the First Workshop on Scholarly Document Processing, SDP@EMNLP 2020, Online, November*

19, 2020 (2020), M. K. Chandrasekaran, A. de Waard, G. Feigenblat, D. Freitag, T. Ghosal, E. H. Hovy, P. Knoth, D. Konopnicki, P. Mayr, R. M. Patton, and M. Shmueli-Scheuer, Eds., Association for Computational Linguistics, pp. 214–224.

[46] Choi, E., Palomaki, J., Lamm, M., Kwiatkowski, T., Das, D., and Collins, M. Decontextualization: Making Sentences Stand-Alone. *Trans. Assoc. Comput. Linguistics 9* (2021), 447–461.

[47] Clark, S. K. *Writing strategies for science*. Teacher Created Materials, 2013.

[48] Cohan, A., Ammar, W., van Zuylen, M., and Cady, F. Structural Scaffolds for Citation Intent Classification in Scientific Publications. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (2019), pp. 3586–3596.

[49] Cohan, A., Feldman, S., Beltagy, I., Downey, D., and Weld, D. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (Online, 2020), Association for Computational Linguistics, pp. 2270–2282.

[50] Collins, E., Augenstein, I., and Riedel, S. A supervised approach to extractive summarisation of scientific papers. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August 3-4, 2017* (2017), R. Levy and L. Specia, Eds., Association for Computational Linguistics, pp. 195–205.

[51] Condit, C. Science reporting to the public: Does the message get twisted? *CMAJ 170*, 9 (2004), 1415–1416.

[52] Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. In *EMNLP 2017* (2017), pp. 670–680.

[53] Dagan, I., Glickman, O., and Magnini, B. The PASCAL Recognising Textual Entailment Challenge. In *Machine Learning Challenges, Evaluating Predictive Uncertainty, Visual Object Classification and Recognizing Textual Entailment, First PASCAL Machine Learning Challenges Workshop, MLCW 2005, Southampton, UK, April 11-13, 2005, Revised Selected Papers* (2005), J. Q. Candela, I. Dagan, B. Magnini, and F. d'Alché-Buc, Eds., vol. 3944 of *Lecture Notes in Computer Science*, Springer, pp. 177–190.

[54] Dai, E., Sun, Y., and Wang, S. Ginger cannot cure cancer: Battling fake health news with a comprehensive data repository. In *Proceedings of the Fourteenth International AAAI Conference on Web and Social Media, ICWSM 2020, Held Virtually, Original Venue: Atlanta,*

Georgia, USA, June 8-11, 2020 (2020), M. D. Choudhury, R. Chunara, A. Culotta, and B. F. Welles, Eds., AAAI Press, pp. 853–862.

[55] DANGOVSKI, R., SHEN, M., BYRD, D., JING, L., TSVETKOVA, D., NAKOVA, P., AND SOLJACIC, M. We Can Explain Your Research in Layman's Terms: Towards Automating Science Journalism at Scale. In *AAAI 2021* (2021), AAAI Press.

[56] DAUMÉ, III, H. Frustratingly Easy Domain Adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (Prague, Czech Republic, June 2007), Association for Computational Linguistics, pp. 256–263.

[57] DAWID, A. P., AND SKENE, A. M. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics) 28*, 1 (1979), 20–28.

[58] DE COMITÉ, F., DENIS, F., GILLERON, R., AND LETOUZEY, F. Positive and Unlabeled Examples Help Learning. In *International Conference on Algorithmic Learning Theory* (1999), Springer, pp. 219–230.

[59] DEL VICARIO, M., BESSI, A., ZOLLO, F., PETRONI, F., SCALA, A., CALDARELLI, G., STANLEY, H. E., AND QUATTROCIOCCHI, W. The Spreading of Misinformation Online. *Proceedings of the National Academy of Sciences 113*, 3 (2016), 554–559.

[60] DEMSZKY, D., GUU, K., AND LIANG, P. Transforming Question Answering Datasets Into Natural Language Inference Datasets. *CoRR abs/1809.02922* (2018).

[61] DENIS, F. PAC Learning From Positive Statistical Queries. In *International Conference on Algorithmic Learning Theory* (1998), Springer, pp. 112–126.

[62] DEVLIN, J., CHANG, M.-W., LEE, K., AND TOUTANOVA, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT 2019* (2019), pp. 4171–4186.

[63] DEYOUNG, J., BELTAGY, I., VAN ZUYLEN, M., KUEHL, B., AND WANG, L. L. MS2: Multi-Document Summarization of Medical Studies. *CoRR abs/2104.06486* (2021).

[64] DIA, O. A., BARSHAN, E., AND BABANEZHAD, R. Semantics Preserving Adversarial Learning. *arXiv preprint arXiv:1903.03905* (2019).

[65] DOĞAN, R. I., LEAMAN, R., AND LU, Z. NCBI Disease Corpus: a Resource for Disease Name Recognition and Concept Normalization. *Journal of Biomedical Informatics 47* (2014), 1–10.

[66] DONAHUE, J., HOFFMAN, J., RODNER, E., SAENKO, K., AND DARRELL, T. Semi-supervised Domain Adaptation with Instance Constraints. In *CVPR* (2013), IEEE Computer Society, pp. 668–675.

[67] DU PLESSIS, M. C., NIU, G., AND SUGIYAMA, M. Analysis of Learning From Positive and Unlabeled Data. In *Advances in Neural Information Processing Systems* (2014), pp. 703–711.

[68] DUAN, N., TANG, D., CHEN, P., AND ZHOU, M. Question Generation for Question Answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017* (2017), M. Palmer, R. Hwa, and S. Riedel, Eds., Association for Computational Linguistics, pp. 866–874.

[69] DUMITRACHE, A., AROYO, L., AND WELTY, C. Crowdsourcing Semantic Label Propagation in Relation Classification. *CoRR abs/1809.00537* (2018).

[70] EBRAHIMI, J., RAO, A., LOWD, D., AND DOU, D. 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)* (2018), pp. 31–36.

[71] ELKAN, C., AND NOTO, K. Learning Classifiers From Only Positive and Unlabeled Data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Minin* (2008), pp. 213–220.

[72] ELSAYED, T., NAKOV, P., BARRÓN-CEDEÑO, A., HASANAIN, M., SUWAILEH, R., DA SAN MARTINO, G., AND ATANASOVA, P. Overview of the CLEF-2019 CheckThat! Lab: Automatic Identification and Verification of Claims. In *International Conference of the Cross-Language Evaluation Forum for European Languages* (2019), Springer, pp. 301–321.

[73] FANG, X., WANG, K., HAN, D., HE, X., WEI, J., ZHAO, L., IMAM, M. U., PING, Z., LI, Y., XU, Y., ET AL. Dietary magnesium intake and the risk of cardiovascular disease, type 2 diabetes, and all-cause mortality: a dose–response meta-analysis of prospective cohort studies. *BMC medicine 14*, 1 (2016), 1–13.

[74] FÄRBER, M., THIEMANN, A., AND JATOWT, A. A High-Quality Gold Standard for Citation-Based Tasks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)* (2018).

[75] FÄRBER, M., THIEMANN, A., AND JATOWT, A. To Cite, or Not to Cite? Detecting Citation Contexts in Text. In *European Conference on Information Retrieval* (2018), Springer, pp. 598–603.

[76] FINKEL, J. R., AND MANNING, C. D. Hierarchical Bayesian Domain Adaptation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (Boulder, Colorado, June 2009), Association for Computational Linguistics, pp. 602–610.

[77] FISCHHOFF, B. Communicating uncertainty: Fulfilling the duty to inform. *Issues in Science and Technology 28* (2012).

[78] FORNACIARI, T., UMA, A., PAUN, S., PLANK, B., HOVY, D., AND POESIO, M. Beyond Black & White: Leveraging Annotator Disagreement via Soft-Label Multi-Task Learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021* (2021), K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tür, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, Eds., Association for Computational Linguistics, pp. 2591–2597.

[79] GAŁECKI, A., AND BURZYKOWSKI, T. Linear mixed-effects model. In *Linear mixed-effects models using R*. Springer, 2013, pp. 245–273.

[80] GANIN, Y., AND LEMPITSKY, V. Unsupervised Domain Adaptation by Backpropagation. In *International Conference on Machine Learning* (2015), pp. 1180–1189.

[81] GANITKEVITCH, J., VAN DURME, B., AND CALLISON-BURCH, C. PPDB: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (Atlanta, Georgia, 2013), Association for Computational Linguistics, pp. 758–764.

[82] GAO, H., AND OATES, T. Universal Adversarial Perturbation for Text Classification. *arXiv preprint arXiv:1910.04618* (2019).

[83] GENCHEVA, P., NAKOV, P., MÀRQUEZ, L., BARRÓN-CEDEÑO, A., AND KOYCHEV, I. A Context-Aware Approach for Detecting Worth-Checking Claims in Political Debates. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017* (Varna, Bulgaria, Sept. 2017), INCOMA Ltd., pp. 267–276.

[84] GERO, K. I., LIU, V., HUANG, S., LEE, J., AND CHILTON, L. B. What makes tweetorials tick: How experts communicate complex topics on twitter. *Proceedings of the ACM on Human-Computer Interaction 5*, CSCW2 (2021), 1–26.

[85] GIMPEL, K., SCHNEIDER, N., O'CONNOR, B., DAS, D., MILLS, D., EISENSTEIN, J., HEILMAN, M., YOGATAMA, D., FLANIGAN, J., AND SMITH, N. A. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments. In *The 49th Annual Meeting of the*

Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA - Short Papers (2011), The Association for Computer Linguistics, pp. 42–47.

[86] GINEV, D., AND MILLER, B. R. Scientific statement classification over arXiv.org. In *Proceedings of the 12th Language Resources and Evaluation Conference* (Marseille, France, 2020), European Language Resources Association, pp. 1219–1226.

[87] GOODFELLOW, I. J., SHLENS, J., AND SZEGEDY, C. Explaining and Harnessing Adversarial Examples. *stat 1050* (2015), 20.

[88] GORDON, M. L., ZHOU, K., PATEL, K., HASHIMOTO, T., AND BERNSTEIN, M. S. The Disagreement Deconvolution: Bringing Machine Learning Performance Metrics In Line With Reality. In *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama, Japan, May 8-13, 2021* (2021), Y. Kitamura, A. Quigley, K. Isbister, T. Igarashi, P. Bjørn, and S. M. Drucker, Eds., ACM, pp. 388:1–388:14.

[89] GOYAL, N., KIVLICHAN, I., ROSEN, R., AND VASSERMAN, L. Is Your Toxicity My Toxicity? Exploring the Impact of Rater Identity on Toxicity Annotation. *CoRR abs/2205.00501* (2022).

[90] GRAVES, L., AND CHERUBINI, F. The Rise of Fact-Checking Sites in Europe. *Reuters Institute for the Study of Journalism* (2016).

[91] GUI, T., ZHANG, Q., HUANG, H., PENG, M., AND HUANG, X.-J. Part-of-Speech Tagging for Twitter with Adversarial Neural Networks. In *EMNLP 2017* (2017), pp. 2411–2420.

[92] GUO, J., SHAH, D., AND BARZILAY, R. Multi-Source Domain Adaptation with Mixture of Experts. In *EMNLP 2018* (2018), pp. 4694–4703.

[93] GUO, Z., SCHLICHTKRULL, M. S., AND VLACHOS, A. A survey on automated fact-checking. *Trans. Assoc. Comput. Linguistics 10* (2022), 178–206.

[94] GURURANGAN, S., MARASOVIC, A., SWAYAMDIPTA, S., LO, K., BELTAGY, I., DOWNEY, D., AND SMITH, N. A. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020* (2020), D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, Eds., Association for Computational Linguistics, pp. 8342–8360.

[95] GUSTAFSON, A., AND RICE, R. E. The effects of uncertainty frames in three science communication topics. *Science Communication 41*, 6 (2019), 679–706.

[96] HAN, X., AND EISENSTEIN, J. Unsupervised Domain Adaptation of Contextualized Embeddings: A Case Study in Early Modern English. 4229–4239.

[97] Hansen, C., Hansen, C., Alstrup, S., Grue Simonsen, J., and Lioma, C. Neural Check-Worthiness Ranking With Weak Supervision: Finding Sentences for Fact-Checking. In *Companion Proceedings of the 2019 World Wide Web Conference* (2019), pp. 994–1000.

[98] Hardalov, M., Arora, A., Nakov, P., and Augenstein, I. A Survey on Stance Detection for Mis- and Disinformation Identification, 2021.

[99] Hardalov, M., Arora, A., Nakov, P., and Augenstein, I. Cross-Domain Label-Adaptive Stance Detection. In *Proceedings of EMNLP* (2021), Association for Computational Linguistics.

[100] Hardalov, M., Arora, A., Nakov, P., and Augenstein, I. Few-Shot Cross-Lingual Stance Detection with Sentiment-Based Pre-Training. *CoRR* (2021).

[101] Hart, P. S., and Feldman, L. The impact of climate change–related imagery and text on public opinion and behavior change. *Science Communication 38*, 4 (2016), 415–441.

[102] Hassan, N., Zhang, G., Arslan, F., Caraballo, J., Jimenez, D., Gawsane, S., Hasan, S., Joseph, M., Kulkarni, A., Nayak, A. K., et al. ClaimBuster: the First-Ever End-to-End Fact-Checking System. *Proceedings of the VLDB Endowment 10*, 12 (2017), 1945–1948.

[103] Hentschel, M., Alonso, O., Counts, S., and Kandylas, V. Finding users we trust: Scaling up verified twitter users using their communication patterns. In *Eighth International AAAI Conference on Weblogs and Social Media* (2014).

[104] Hidey, C., Chakrabarty, T., Alhindi, T., Varia, S., Krstovski, K., Diab, M., and Muresan, S. DeSePtion: Dual sequence prediction and adversarial examples for improved fact-checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (Online, July 2020), Association for Computational Linguistics, pp. 8593–8606.

[105] Hinton, G. E., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *CoRR abs/1503.02531* (2015).

[106] Hochreiter, S., and Schmidhuber, J. Long short-term memory. *Neural computation 9*, 8 (1997), 1735–1780.

[107] Holm, A. N., Plank, B., Wright, D., and Augenstein, I. Longitudinal Citation Prediction using Temporal Graph Neural Networks. *arXiv preprint arXiv:2012.05742* (2020).

[108] Holub, A., Perona, P., and Burl, M. C. Entropy-based active learning for object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*

Workshops 2008, Anchorage, AK, USA, 23-28 June, 2008 (2008), IEEE Computer Society, pp. 1–8.

[109] HOULSBY, N., HUSZAR, F., GHAHRAMANI, Z., AND LENGYEL, M. Bayesian active learning for classification and preference learning. *CoRR abs/1112.5745* (2011).

[110] HOVY, D., BERG-KIRKPATRICK, T., VASWANI, A., AND HOVY, E. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (Atlanta, Georgia, 2013), Association for Computational Linguistics, pp. 1120–1130.

[111] HOVY, D., PLANK, B., AND SØGAARD, A. Experiments with crowdsourced re-annotation of a POS tagging data set. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers* (2014), The Association for Computer Linguistics, pp. 377–382.

[112] HOWARD, J., AND RUDER, S. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers* (2018), I. Gurevych and Y. Miyao, Eds., Association for Computational Linguistics, pp. 328–339.

[113] HOWELL, L., ET AL. Digital Wildfires in a Hyperconnected World. *WEF report 3* (2013), 15–94.

[114] IOANNIDIS, J. P. Why most published research findings are false. *PLoS medicine 2*, 8 (2005), e124.

[115] JARADAT, I., GENCHEVA, P., BARRÓN-CEDEÑO, A., MÀRQUEZ, L., AND NAKOV, P. Claimrank: Detecting Check-Worthy Claims in Arabic and English. 26–30.

[116] JIANG, Z., XU, F. F., ARAKI, J., AND NEUBIG, G. How can we know what language models know. *Trans. Assoc. Comput. Linguistics 8* (2020), 423–438.

[117] JIN, D., JIN, Z., ZHOU, J. T., AND SZOLOVITS, P. TextFool: Fool your Model with Natural Adversarial Text.

[118] JOLLY, S., ATANASOVA, P., AND AUGENSTEIN, I. Generating Fluent Fact Checking Explanations with Unsupervised Post-Editing. *Information 13* (2022).

[119] JÜRGENS, D., KUMAR, S., HOOVER, R., MCFARLAND, D., AND JURAFSKY, D. Measuring the Evolution of a Scientific Field through Citation Frames. *Transactions of the Association for Computational Linguistics 6* (2018), 391–406.

[120] KESKAR, N. S., MCCANN, B., VARSHNEY, L. R., XIONG, C., AND SOCHER, R. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858* (2019).

[121] KIM, J.-D., OHTA, T., TATEISI, Y., AND TSUJII, J. GENIA Corpus – a Semantically Annotated Corpus for Bio-Textmining. *Bioinformatics 19*, suppl_1 (2003), i180–i182.

[122] KIM, J.-D., OHTA, T., TSURUOKA, Y., TATEISI, Y., AND COLLIER, N. Introduction to the Bio-Entity Recognition Task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications* (2004), Citeseer, pp. 70–75.

[123] KIM, Y., AND ALLAN, J. FEVER breaker's run of team NbAuzDrLqg. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)* (Hong Kong, China, Nov. 2019), Association for Computational Linguistics, pp. 99–104.

[124] KIM, Y.-B., STRATOS, K., AND KIM, D. Domain Attention With an Ensemble of Experts. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (2017), pp. 643–653.

[125] KINDERMANS, P.-J., SCHÜTT, K., MÜLLER, K.-R., AND DÄHNE, S. Investigating the Influence of Noise and Distractors on the Interpretation of Neural Networks. *arXiv preprint arXiv:1611.07270* (2016).

[126] KIRYO, R., NIU, G., DU PLESSIS, M. C., AND SUGIYAMA, M. Positive-Unlabeled Learning With Non-Negative Risk Estimator. In *Advances in Neural Information Processing Systems* (2017), pp. 1675–1685.

[127] KONSTANTINOVSKIY, L., PRICE, O., BABAKAR, M., AND ZUBIAGA, A. Towards Automated Factchecking: Developing an Annotation Schema and Benchmark For Consistent Automated Claim Detection. *arXiv preprint arXiv:1809.08193* (2018).

[128] KOTONYA, N., AND TONI, F. Explainable Automated Fact-Checking for Public Health Claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Online, Nov. 2020), Association for Computational Linguistics, pp. 7740–7754.

[129] KOUW, W. M., AND LOOG, M. A Review of Domain Adaptation Without Target Labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).

[130] KRINGELUM, J., KJAERULFF, S. K., BRUNAK, S., LUND, O., OPREA, T. I., AND TABOUREAU, O. ChemProt-3.0: a Global Chemical Biology Diseases Mapping. *Database 2016* (2016).

[131] KRIPPENDORFF, K. Computing Krippendorff's alpha-reliability.

[132] KUA, E., REDER, M., AND GROSSEL, M. J. Science in the news: a study of reporting genomics. *Public Understanding of Science 13*, 3 (2004), 309–322.

[133] KUHN, T., BARBANO, P. E., NAGY, M. L., AND KRAUTHAMMER, M. Broadening the Scope of Nanopublications. In *The Semantic Web: Semantics and Big Data, 10th International Conference, ESWC 2013, Montpellier, France, May 26-30, 2013. Proceedings* (2013), P. Cimiano, Ó. Corcho, V. Presutti, L. Hollink, and S. Rudolph, Eds., vol. 7882 of *Lecture Notes in Computer Science*, Springer, pp. 487–501.

[134] KULIS, B., SAENKO, K., AND DARRELL, T. What You Saw is Not What You Get: Domain Adaptation Using Asymmetric Kernel Transforms. In *CVPR* (2011), IEEE Computer Society, pp. 1785–1792.

[135] KURU, O., STECULA, D., LU, H., OPHIR, Y., CHAN, M.-P. S., WINNEG, K., HALL JAMIESON, K., AND ALBARRACÍN, D. The effects of scientific messages and narratives about vaccination. *PLoS One 16*, 3 (2021), e0248328.

[136] LEHMAN, E., DEYOUNG, J., BARZILAY, R., AND WALLACE, B. C. Inferring which medical treatments work from reports of clinical trials. 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)* (Minneapolis, Minnesota, June 2019), Association for Computational Linguistics, pp. 3705–3717.

[137] LETOUZEY, F., DENIS, F., AND GILLERON, R. Learning From Positive and Unlabeled Examples. In *International Conference on Algorithmic Learning Theory* (2000), Springer, pp. 71–85.

[138] LEWIS, M., LIU, Y., GOYAL, N., GHAZVININEJAD, M., MOHAMED, A., LEVY, O., STOYANOV, V., AND ZETTLEMOYER, L. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020* (2020), D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, Eds., Association for Computational Linguistics, pp. 7871–7880.

[139] LI, H., CHEN, Z., LIU, B., WEI, X., AND SHAO, J. Spotting Fake Reviews Via Collective Positive-Unlabeled Learning. In *2014 IEEE International Conference on Data Mining* (2014), IEEE, pp. 899–904.

[140] LI, J., SUN, Y., JOHNSON, R. J., SCIAKY, D., WEI, C.-H., LEAMAN, R., DAVIS, A. P., MATTINGLY, C. J., WIEGERS, T. C., AND LU, Z. BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database 2016* (2016).

[141] LI, X., BURNS, G. A., AND PENG, N. A Paragraph-level Multi-task Learning Model for Scientific Fact-Verification. In *Proceedings of the Workshop on Scientific Document Understanding co-located with 35th AAAI Conference on Artificial Inteligence, SDU@AAAI 2021, Virtual Event, February 9, 2021* (2021), A. P. B. Veyseh, F. Dernoncourt, T. H. Nguyen, W. Chang, and L. A. Celi, Eds., vol. 2831 of *CEUR Workshop Proceedings*, CEUR-WS.org.

[142] LI, Y., BALDWIN, T., AND COHN, T. What's in a Domain? Learning Domain-Robust Text Representations Using Adversarial Training. 474–479.

[143] LI, Y., ZHANG, J., AND YU, B. An NLP Analysis of Exaggerated Claims in Science News. In *Proceedings of the 2017 Workshop: Natural Language Processing meets Journalism, NLPmJ@EMNLP, Copenhagen, Denmark, September 7, 2017* (2017), O. Popescu and C. Strapparava, Eds., Association for Computational Linguistics, pp. 106–111.

[144] LIN, C., BETHARD, S., DLIGACH, D., SADEQUE, F., SAVOVA, G., AND MILLER, T. A. Does BERT Need Domain Adaptation for Clinical Negation Detection? *Journal of the American Medical Informatics Association 27*, 4 (2020), 584–591.

[145] LIN, C.-Y. Rouge: A Package for Automatic Evaluation of Summaries. In *Text summarization branches out* (2004), pp. 74–81.

[146] LIPTON, Z. C., WANG, Y.-X., AND SMOLA, A. J. Detecting and Correcting for Label Shift with Black Box Predictors. In *ICML* (2018), J. G. Dy and A. Krause, Eds., vol. 80 of *Proceedings of Machine Learning Research*, PMLR, pp. 3128–3136.

[147] LIU, P., YUAN, W., FU, J., JIANG, Z., HAYASHI, H., AND NEUBIG, G. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *CoRR abs/2107.13586* (2021).

[148] LIU, Y., OTT, M., GOYAL, N., DU, J., JOSHI, M., CHEN, D., LEVY, O., LEWIS, M., ZETTLEMOYER, L., AND STOYANOV, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR abs/1907.11692* (2019).

[149] LO, K., WANG, L. L., NEUMANN, M., KINNEY, R., AND WELD, D. S. S2ORC: The Semantic Scholar Open Research Corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (2020), pp. 4969–4983.

[150] LUAN, Y., HE, L., OSTENDORF, M., AND HAJISHIRZI, H. Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (Brussels, Belgium, Oct.-Nov. 2018), Association for Computational Linguistics, pp. 3219–3232.

[151] MA, J., GAO, W., JOTY, S. R., AND WONG, K. Sentence-level evidence embedding for claim verification with hierarchical attention networks. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers* (2019), A. Korhonen, D. R. Traum, and L. Màrquez, Eds., Association for Computational Linguistics, pp. 2561–2571.

[152] MA, J., GAO, W., AND WONG, K.-F. Detect Rumors on Twitter by Promoting Information Campaigns With Generative Adversarial Learning. In *The World Wide Web Conference* (2019), pp. 3049–3055.

[153] MA, X., XU, P., WANG, Z., NALLAPATI, R., AND XIANG, B. Domain Adaptation with BERT-based Domain Classification and Data Selection. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)* (2019), pp. 76–83.

[154] MACKAY, D. J. C. Information-based objective functions for active data selection. *Neural Comput. 4*, 4 (1992), 590–604.

[155] MAHABADI, R. K., ZETTLEMOYER, L., HENDERSON, J., SAEIDI, M., MATHIAS, L., STOYANOV, V., AND YAZDANI, M. PERFECT: prompt-free and efficient few-shot learning with language models. *CoRR abs/2204.01172* (2022).

[156] MALON, C. Team Papelo: Transformer Networks at FEVER. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)* (2018), pp. 109–113.

[157] MARCUS, M. P., SANTORINI, B., AND MARCINKIEWICZ, M. A. Building a Large Annotated Corpus of English: The Penn Treebank. *Comput. Linguistics 19*, 2 (1993), 313–330.

[158] MCCARTHY, P. M., AND JARVIS, S. Mtld, vocd-d, and hd-d: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior research methods 42*, 2 (2010), 381–392.

[159] MEJZINI, R., FLYNN, L. L., PITOUT, I. L., FLETCHER, S., WILTON, S. D., AND AKKARI, P. A. ALS Genetics, Mechanisms, and Therapeutics: Where Are We Now? *Frontiers in Neuroscience 13* (2019).

[160] MENCHER, M., AND SHILTON, W. P. *News reporting and writing*. Brown & Benchmark Publishers Madison, WI, 1997.

[161] MICHEL, P., LI, X., NEUBIG, G., AND PINO, J. M. On Evaluation of Adversarial Perturbations for Sequence-to-Sequence Models. In *Proceedings of NAACL-HLT* (2019), pp. 3103–3114.

[162] MOHAN, S., AND LI, D. MedMentions: A Large Biomedical Corpus Annotated with UMLS Concepts. In *1st Conference on Automated Knowledge Base Construction, AKBC 2019, Amherst, MA, USA, May 20-22, 2019* (2019).

[163] MOHR, I., WÜHRL, A., AND KLINGER, R. Covert: A corpus of fact-checked biomedical COVID-19 tweets. *CoRR abs/2204.12164* (2022).

[164] MOORE, A. Bad science in the headlines: Who takes responsibility when science is distorted in the mass media? *EMBO reports 7*, 12 (2006), 1193–1196.

[165] NADKARNI, R., WADDEN, D., BELTAGY, I., SMITH, N. A., HAJISHIRZI, H., AND HOPE, T. Scientific language models for biomedical knowledge base completion: An empirical study. In *3rd Conference on Automated Knowledge Base Construction, AKBC 2021, Virtual, October 4-8, 2021* (2021), D. Chen, J. Berant, A. McCallum, and S. Singh, Eds.

[166] NAKOV, P., MARTINO, G. D. S., ELSAYED, T., BARRÓN-CEDEÑO, A., MÍGUEZ, R., SHAAR, S., ALAM, F., HAOUARI, F., HASANAIN, M., BABULKOV, N., NIKOLOV, A., SHAHI, G. K., STRUSS, J. M., AND MANDL, T. The CLEF-2021 CheckThat! Lab on Detecting Check-Worthy Claims, Previously Fact-Checked Claims, and Fake News. In *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part II* (2021), D. Hiemstra, M. Moens, J. Mothe, R. Perego, M. Potthast, and F. Sebastiani, Eds., vol. 12657 of *Lecture Notes in Computer Science*, Springer, pp. 639–649.

[167] NATIONAL ACADEMIES OF SCIENCES, ENGINEERING, AND MEDICINE. Communicating science effectively: A research agenda.

[168] NELKIN, D. Selling Science: How the Press Covers Science and Technology.

[169] NEUMANN, M., KING, D., BELTAGY, I., AND AMMAR, W. ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task, BioNLP@ACL 2019, Florence, Italy, August 1, 2019* (2019), D. Demner-Fushman, K. B. Cohen, S. Ananiadou, and J. Tsujii, Eds., Association for Computational Linguistics, pp. 319–327.

[170] NIE, Y., WILLIAMS, A., DINAN, E., BANSAL, M., WESTON, J., AND KIELA, D. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (Online, 2020), Association for Computational Linguistics, pp. 4885–4901.

[171] NIELSEN, F. On a Generalization of the Jensen-Shannon Divergence and the Jensen-Shannon Centroid. *Entropy 22*, 2 (2020), 221.

[172] NIEWINSKI, P., PSZONA, M., AND JANICKA, M. GEM: Generative enhanced model for adversarial attacks. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)* (Hong Kong, China, Nov. 2019), Association for Computational Linguistics, pp. 20–26.

[173] NIGHOJKAR, A., AND LICATO, J. Improving paraphrase detection with the adversarial paraphrasing task. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (Online, 2021), Association for Computational Linguistics, pp. 7106–7116.

[174] NYE, B., LI, J. J., PATEL, R., YANG, Y., MARSHALL, I. J., NENKOVA, A., AND WALLACE, B. C. A Corpus With Multi-Level Annotations of Patients, Interventions and Outcomes to Support Language Processing for Medical Literature. In *Proceedings of the conference. Association for Computational Linguistics. Meeting* (2018), vol. 2018, NIH Public Access, p. 197.

[175] OSTROWSKI, W., ARORA, A., ATANASOVA, P., AND AUGENSTEIN, I. Multi-Hop Fact Checking of Political Claims. *arXiv preprint arXiv:2009.06401* (2020).

[176] PAN, L., CHEN, W., XIONG, W., KAN, M., AND WANG, W. Y. Zero-shot Fact Verification by Claim Generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021* (2021), C. Zong, F. Xia, W. Li, and R. Navigli, Eds., Association for Computational Linguistics, pp. 476–483.

[177] PAUN, S., CARPENTER, B., CHAMBERLAIN, J., HOVY, D., KRUSCHWITZ, U., AND POESIO, M. Comparing Bayesian Models of Annotation. *Trans. Assoc. Comput. Linguistics 6* (2018), 571–585.

[178] PEI, J., ANANTHASUBRAMANIAM, A., WANG, X., ZHOU, N., DEDELOUDIS, A., SARGENT, J., AND JURGENS, D. Potato: The portable text annotation tool. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (2022).

[179] PEI, J., AND JURGENS, D. Measuring sentence-level and aspect-level (un)certainty in science communications. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021* (2021), M. Moens, X. Huang, L. Specia, and S. W. Yih, Eds., Association for Computational Linguistics, pp. 9959–10011.

[180] PELLECHIA, M. G. Trends in science coverage: A content analysis of three us newspapers. *Public Understanding of Science 6*, 1 (1997), 49.

[181] PENG, M., XING, X., ZHANG, Q., FU, J., AND HUANG, X.-J. Distantly Supervised Named Entity Recognition using Positive-Unlabeled Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (2019), pp. 2409–2419.

[182] PETERS, M. E., NEUMANN, M., IYYER, M., GARDNER, M., CLARK, C., LEE, K., AND ZETTLEMOYER, L. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)* (2018), M. A. Walker, H. Ji, and A. Stent, Eds., Association for Computational Linguistics, pp. 2227–2237.

[183] PETERSON, J. C., BATTLEDAY, R. M., GRIFFITHS, T. L., AND RUSSAKOVSKY, O. Human Uncertainty Makes Classification More Robust. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019* (2019), IEEE, pp. 9616–9625.

[184] PLANK, B., HOVY, D., AND SØGAARD, A. Linguistically debatable or just plain wrong? In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers* (2014), The Association for Computer Linguistics, pp. 507–511.

[185] POLIAK, A., NARADOWSKY, J., HALDAR, A., RUDINGER, R., AND DURME, B. V. Hypothesis Only Baselines in Natural Language Inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics, *SEM@NAACL-HLT 2018, New Orleans, Louisiana, USA, June 5-6, 2018* (2018), M. Nissim, J. Berant, and A. Lenci, Eds., Association for Computational Linguistics, pp. 180–191.

[186] POTTHAST, M., GOLLUB, T., KOMLOSSY, K., SCHUSTER, S., WIEGMANN, M., GARCES FERNANDEZ, E. P., HAGEN, M., AND STEIN, B. Crowdsourcing a large corpus of clickbait on Twitter. In *Proceedings of the 27th International Conference on Computational Linguistics* (Santa Fe, New Mexico, USA, Aug. 2018), Association for Computational Linguistics, pp. 1498–1507.

[187] PÖTTKER, H. News and its Communicative Quality: The Inverted PyramidWhen and Why Did it Appear? *Journalism Studies 4*, 4 (2003), 501–511.

[188] PRABHAKARAN, V., HAMILTON, W. L., MCFARLAND, D., AND JURAFSKY, D. Predicting the rise and fall of scientific topics from trends in their rhetorical framing. In *Proceedings of*

the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Berlin, Germany, 2016), Association for Computational Linguistics, pp. 1170–1180.

[189] PRADEEP, R., MA, X., NOGUEIRA, R., AND LIN, J. Scientific Claim Verification with VERT5ERINI. *CoRR abs/2010.11930* (2020).

[190] PURI, R., AND CATANZARO, B. Zero-shot Text Classification With Generative Language Models. *CoRR abs/1912.10165* (2019).

[191] RADFORD, A., NARASIMHAN, K., SALIMANS, T., AND SUTSKEVER, I. Improving Language Understanding by Generative Pre-Training. *Technical report*.

[192] RADFORD, A., WU, J., CHILD, R., LUAN, D., AMODEI, D., AND SUTSKEVER, I. Language models are unsupervised multitask learners. *OpenAI Blog 1*, 8 (2019), 9.

[193] RAFFEL, C., SHAZEER, N., ROBERTS, A., LEE, K., NARANG, S., MATENA, M., ZHOU, Y., LI, W., AND LIU, P. J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res. 21* (2020), 140:1–140:67.

[194] RAJPURKAR, P., ZHANG, J., LOPYREV, K., AND LIANG, P. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016* (2016), J. Su, X. Carreras, and K. Duh, Eds., The Association for Computational Linguistics, pp. 2383–2392.

[195] RAPPAPORT, L. Dietary magnesium tied to lower risk of heart disease and diabetes. *Reuters*.

[196] REDI, M., FETAHU, B., MORGAN, J., AND TARABORELLI, D. Citation Needed: A Taxonomy and Algorithmic Assessment of Wikipedia's Verifiability. In *The World Wide Web Conference* (2019), pp. 1567–1578.

[197] REIMERS, N., AND GUREVYCH, I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (Hong Kong, China, 2019), Association for Computational Linguistics, pp. 3982–3992.

[198] REN, S., DENG, Y., HE, K., AND CHE, W. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (2019), pp. 1085–1097.

[199] REN, Y., JI, D., AND ZHANG, H. Positive Unlabeled Learning for Deceptive Reviews Detection. In *EMNLP 2014* (2014), pp. 488–498.

[200] RIABI, A., SCIALOM, T., KERARON, R., SAGOT, B., SEDDAH, D., AND STAIANO, J. Synthetic Data Augmentation for Zero-Shot Cross-Lingual Question Answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021* (2021), M. Moens, X. Huang, L. Specia, and S. W. Yih, Eds., Association for Computational Linguistics, pp. 7016–7030.

[201] RIBEIRO, M. T., SINGH, S., AND GUESTRIN, C. 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)* (2018), pp. 856–865.

[202] RICHARDSON, L. *Writing strategies: Reaching diverse audiences*, vol. 21. Sage Publications, 1990.

[203] RIETZLER, A., STABINGER, S., OPITZ, P., AND ENGL, S. Adapt or Get Left Behind: Domain Adaptation Through Bert Language Model Finetuning for Aspect-Target Sentiment Classification. 4933–4941.

[204] ROBERTSON, S. E., WALKER, S., JONES, S., HANCOCK-BEAULIEU, M., AND GATFORD, M. Okapi at trec-3. In *TREC* (1994).

[205] ROLAND, M.-C. Quality and integrity in scientific writing: prerequisites for quality in science communication. *Journal of Science Communication 8*, 2 (2009), A04.

[206] SAAKYAN, A., CHAKRABARTY, T., AND MURESAN, S. COVID-Fact: Fact Extraction and Verification of Real-World Claims on COVID-19 Pandemic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021* (2021), C. Zong, F. Xia, W. Li, and R. Navigli, Eds., Association for Computational Linguistics, pp. 2116–2129.

[207] SAFAVI, T., ZHU, J., AND KOUTRA, D. NegatER: Unsupervised Discovery of Negatives in Commonsense Knowledge Bases. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021* (2021), M. Moens, X. Huang, L. Specia, and S. W. Yih, Eds., Association for Computational Linguistics, pp. 5633–5646.

[208] SAFFER, A. J., SOMMERFELDT, E. J., AND TAYLOR, M. The effects of organizational twitter interactivity on organization–public relationships. *Public relations review 39*, 3 (2013), 213–215.

[209] SAI, A. B., MOHANKUMAR, A. K., AND KHAPRA, M. M. A Survey of Evaluation Metrics Used for NLG Systems. *CoRR abs/2008.12009* (2020).

[210] SALITA, J. T. Writing for lay audiences: A challenge for scientists. *Medical Writing 24* (2015), 183–189.

[211] SANH, V., DEBUT, L., CHAUMOND, J., AND WOLF, T. DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter. *arXiv preprint arXiv:1910.01108* (2019).

[212] SCHICK, T., SCHMID, H., AND SCHÜTZE, H. Automatically Identifying Words That Can Serve as Labels for Few-Shot Text Classification. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020* (2020), D. Scott, N. Bel, and C. Zong, Eds., International Committee on Computational Linguistics, pp. 5569–5578.

[213] SCHICK, T., AND SCHÜTZE, H. Exploiting Cloze Questions for Few-Shot Text Classification and Natural Language Inference. *Computing Research Repository arXiv:2001.07676* (2020).

[214] SCHICK, T., AND SCHÜTZE, H. It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners. *Computing Research Repository arXiv:2009.07118* (2020).

[215] SCHLICHTKRULL, M. S., KARPUKHIN, V., OGUZ, B., LEWIS, M., YIH, W., AND RIEDEL, S. Joint verification and reranking for open fact checking over tables. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021* (2021), C. Zong, F. Xia, W. Li, and R. Navigli, Eds., Association for Computational Linguistics, pp. 6787–6799.

[216] SHAAR, S., BABULKOV, N., DA SAN MARTINO, G., AND NAKOV, P. That is a known lie: Detecting previously fact-checked claims. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (Online, July 2020), Association for Computational Linguistics, pp. 3607–3618.

[217] SHIN, T., RAZEGHI, Y., IV, R. L. L., WALLACE, E., AND SINGH, S. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020* (2020), B. Webber, T. Cohn, Y. He, and Y. Liu, Eds., Association for Computational Linguistics, pp. 4222–4235.

[218] SIMMONS, J. P., NELSON, L. D., AND SIMONSOHN, U. False-positive psychology: undisclosed flexibility in data collection and analysis allows presenting anything as significant.

[219] SNOW, R., O'CONNOR, B., JURAFSKY, D., AND NG, A. Y. Cheap and Fast - But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks. In *2008 Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, Proceedings of the Conference, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL* (2008), ACL, pp. 254–263.

[220] SONG, K., TAN, X., QIN, T., LU, J., AND LIU, T. Mpnet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual* (2020), H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds.

[221] STERCKX, L., DEMEESTER, T., DEVELDER, C., AND CARAGEA, C. Supervised Keyphrase Extraction as Positive Unlabeled Learning. In *EMNLP 2016* (2016), pp. 1–6.

[222] STOLAROFF, J. K., SAMARAS, C., ONEILL, E. R., LUBERS, A., MITCHELL, A. S., AND CEPERLEY, D. Energy use and life cycle greenhouse gas emissions of drones for commercial package delivery. *Nature communications 9*, 1 (2018), 1–13.

[223] SUGIYAMA, K., KUMAR, T., KAN, M.-Y., AND TRIPATHI, R. C. Identifying Citing Sentences in Research Papers Using Supervised Learning. In *2010 International Conference on Information Retrieval & Knowledge Management (CAMP)* (2010), IEEE, pp. 67–72.

[224] SUMNER, P., VIVIAN-GRIFFITHS, S., BOIVIN, J., WILLIAMS, A., VENETIS, C. A., DAVIES, A., OGDEN, J., WHELAN, L., HUGHES, B., DALTON, B., ET AL. The Association Between Exaggeration in Health Related Science News and Academic Press Releases: Retrospective Observational Study. *BMJ 349* (2014).

[225] SUN, B., FENG, J., AND SAENKO, K. Return of Frustratingly Easy Domain Adaptation. In *AAAI* (2016), D. Schuurmans and M. P. Wellman, Eds., AAAI Press, pp. 2058–2065.

[226] SZEGEDY, C., ZAREMBA, W., SUTSKEVER, I., BRUNA, J., ERHAN, D., GOODFELLOW, I., AND FERGUS, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199* (2013).

[227] TAN, C., AND LEE, L. A corpus of sentence-level revisions in academic writing: A step towards understanding statement strength in communication. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (Baltimore, Maryland, 2014), Association for Computational Linguistics, pp. 403–408.

[228] TAYLOR, J. W., LONG, M., ASHLEY, E., DENNING, A., GOUT, B., HANSEN, K., HUWS, T., JENNINGS, L., QUINN, S., SARKIES, P., ET AL. When medical news comes from press

releasesa case study of pancreatic cancer and processed meat. *PloS one 10*, 6 (2015), e0127848.

[229] THORNE, J., AND VLACHOS, A. Automated Fact Checking: Task Formulations, Methods and Future Directions. In *Proceedings of the 27th International Conference on Computational Linguistics* (Santa Fe, New Mexico, USA, Aug. 2018), Association for Computational Linguistics, pp. 3346–3359.

[230] THORNE, J., VLACHOS, A., CHRISTODOULOPOULOS, C., AND MITTAL, A. 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, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)* (2018), M. A. Walker, H. Ji, and A. Stent, Eds., Association for Computational Linguistics, pp. 809–819.

[231] THORNE, J., VLACHOS, A., CHRISTODOULOPOULOS, C., AND MITTAL, A. Evaluating adversarial attacks against multiple fact verification systems. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (2019), pp. 2937–2946.

[232] THORNE, J., VLACHOS, A., COCARASCU, O., CHRISTODOULOPOULOS, C., AND MITTAL, A. The FEVER2.0 shared task. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)* (Hong Kong, China, Nov. 2019), Association for Computational Linguistics, pp. 1–6.

[233] UMA, A., ALMANEA, D., AND POESIO, M. Scaling and Disagreements: Bias, Noise, and Ambiguity. *Frontiers Artif. Intell. 5* (2022), 818451.

[234] UMA, A., FORNACIARI, T., HOVY, D., PAUN, S., PLANK, B., AND POESIO, M. A Case for Soft Loss Functions. In *Proceedings of the Eighth AAAI Conference on Human Computation and Crowdsourcing, HCOMP 2020, Hilversum, The Netherlands (virtual), October 25-29, 2020* (2020), L. Aroyo and E. Simperl, Eds., AAAI Press, pp. 173–177.

[235] UMA, A., FORNACIARI, T., HOVY, D., PAUN, S., PLANK, B., AND POESIO, M. Learning from Disagreement: A Survey. *J. Artif. Intell. Res. 72* (2021), 1385–1470.

[236] VADAPALLI, R., SYED, B., PRABHU, N., SRINIVASAN, B. V., AND VARMA, V. When science journalism meets artificial intelligence : An interactive demonstration. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (Brussels, Belgium, 2018), Association for Computational Linguistics, pp. 163–168.

[237] VASWANI, A., SHAZEER, N., PARMAR, N., USZKOREIT, J., JONES, L., GOMEZ, A. N., KAISER, Ł., AND POLOSUKHIN, I. Attention is All You Need. In *Advances in Neural Information Processing Systems* (2017), pp. 5998–6008.

[238] VLACHOS, A., AND RIEDEL, S. Fact Checking: Task Definition and Dataset Construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science* (2014), pp. 18–22.

[239] WADDEN, D., LIN, S., LO, K., WANG, L. L., VAN ZUYLEN, M., COHAN, A., AND HAJISHIRZI, H. Fact or Fiction: Verifying Scientific Claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020* (2020), B. Webber, T. Cohn, Y. He, and Y. Liu, Eds., Association for Computational Linguistics, pp. 7534–7550.

[240] WADDEN, D., LO, K., WANG, L. L., COHAN, A., BELTAGY, I., AND HAJISHIRZI, H. Longchecker: Improving scientific claim verification by modeling full-abstract context. *CoRR abs/2112.01640* (2021).

[241] WALLACE, E., FENG, S., KANDPAL, N., GARDNER, M., AND SINGH, S. Universal Adversarial Triggers for Attacking and Analyzing NLP. 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)* (2019), pp. 2153–2162.

[242] WALSH-CHILDERS, K., BRADDOCK, J., RABAZA, C., AND SCHWITZER, G. One step forward, one step back: changes in news coverage of medical interventions. *Health communication 33*, 2 (2018), 174–187.

[243] WANG, C., AND FAN, J. Medical Relation Extraction with Manifold Models. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers* (2014), The Association for Computer Linguistics, pp. 828–838.

[244] WANG, W., WEI, F., DONG, L., BAO, H., YANG, N., AND ZHOU, M. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual* (2020), H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds.

[245] WANG, W., WEI, F., DONG, L., BAO, H., YANG, N., AND ZHOU, M. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers, 2020.

[246] WANG, W. Y. "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, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers* (2017), R. Barzilay and M. Kan, Eds., Association for Computational Linguistics, pp. 422–426.

[247] WANG, Z., HALE, S. A., ADELANI, D. I., GRABOWICZ, P. A., HARTMANN, T., FLÖCK, F., AND JURGENS, D. Demographic inference and representative population estimates from multilingual social media data. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019* (2019), L. Liu, R. W. White, A. Mantrach, F. Silvestri, J. J. McAuley, R. Baeza-Yates, and L. Zia, Eds., ACM, pp. 2056–2067.

[248] WEI, C., PENG, Y., LEAMAN, R., DAVIS, A. P., MATTINGLY, C. J., LI, J., WIEGERS, T. C., AND LU, Z. Assessing the state of the art in biomedical relation extraction: overview of the biocreative V chemical-disease relation (CDR) task. *Database J. Biol. Databases Curation 2016* (2016).

[249] WEIGOLD, M. F. Communicating science: A review of the literature. *Science communication 23*, 2 (2001), 164–193.

[250] WILLIAMS, A., NANGIA, N., AND BOWMAN, S. R. 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, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)* (2018), M. A. Walker, H. Ji, and A. Stent, Eds., Association for Computational Linguistics, pp. 1112–1122.

[251] WOLF, T., DEBUT, L., SANH, V., CHAUMOND, J., DELANGUE, C., MOI, A., CISTAC, P., RAULT, T., LOUF, R., FUNTOWICZ, M., DAVISON, J., SHLEIFER, S., VON PLATEN, P., MA, C., JERNITE, Y., PLU, J., XU, C., SCAO, T. L., GUGGER, S., DRAME, M., LHOEST, Q., AND RUSH, A. M. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020* (2020), Q. Liu and D. Schlangen, Eds., Association for Computational Linguistics, pp. 38–45.

[252] WOLOSHIN, S., AND SCHWARTZ, L. M. Press Releases: Translating Research Into News. *Jama 287*, 21 (2002), 2856–2858.

[253] WOLOSHIN, S., SCHWARTZ, L. M., CASELLA, S. L., KENNEDY, A. T., AND LARSON, R. J. Press Releases by Academic Medical Centers: Not So Academic? *Annals of Internal Medicine 150*, 9 (2009), 613–618.

[254] WRIGHT, D., AND AUGENSTEIN, I. Claim Check-Worthiness Detection as Positive Unlabelled Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020* (Online, Nov. 2020), Association for Computational Linguistics, pp. 476–488.

[255] WRIGHT, D., AND AUGENSTEIN, I. Transformer Based Multi-Source Domain Adaptation. In *Proceedings of EMNLP* (2020), Association for Computational Linguistics.

[256] WRIGHT, D., AND AUGENSTEIN, I. CiteWorth: Cite-Worthiness Detection for Improved Scientific Document Understanding. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021* (2021), C. Zong, F. Xia, W. Li, and R. Navigli, Eds., vol. ACL/IJCNLP 2021 of *Findings of ACL*, Association for Computational Linguistics, pp. 1796–1807.

[257] WRIGHT, D., AND AUGENSTEIN, I. Semi-supervised exaggeration detection of health science press releases. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing* (2021), pp. 10824–10836.

[258] WRIGHT, D., GENTILE, A. L., FAUX, N., AND BECK, K. L. Bioact: Biomedical knowledge base construction using active learning. *bioRxiv* (2022).

[259] WRIGHT, D., KATSIS, Y., MEHTA, R., AND HSU, C.-N. NormCo: Deep Disease Normalization for Biomedical Knowledge Base Construction. In *AKBC* (2019).

[260] WRIGHT, D., WADDEN, D., LO, K., KUEHL, B., COHAN, A., AUGENSTEIN, I., AND WANG, L. L. Generating scientific claims for zero-shot scientific fact checking. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022* (2022), S. Muresan, P. Nakov, and A. Villavicencio, Eds., Association for Computational Linguistics, pp. 2448–2460.

[261] YAO, T., PAN, Y., NGO, C.-W., LI, H., AND MEI, T. Semi-supervised Domain Adaptation with Subspace Learning for Visual Recognition. In *CVPR* (2015), IEEE Computer Society, pp. 2142–2150.

[262] YASUNAGA, M., KASAI, J., ZHANG, R., FABBRI, A. R., LI, I., FRIEDMAN, D., AND RADEV, D. R. Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019* (2019), AAAI Press, pp. 7386–7393.

[263] Yavchitz, A., Boutron, I., Bafeta, A., Marroun, I., Charles, P., Mantz, J., and Ravaud, P. Misrepresentation of randomized controlled trials in press releases and news coverage: a cohort study.

[264] Yin, W., and Roth, D. TwoWingOS: A Two-Wing Optimization Strategy for Evidential Claim Verification. In *EMNLP* (2018), E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, Eds., Association for Computational Linguistics, pp. 105–114.

[265] Yu, B., Li, Y., and Wang, J. Detecting Causal Language Use in Science Findings. In *EMNLP* (2019), pp. 4656–4666.

[266] Yu, B., Wang, J., Guo, L., and Li, Y. Measuring Correlation-to-Causation Exaggeration in Press Releases. In *Proceedings of the 28th International Conference on Computational Linguistics* (2020), pp. 4860–4872.

[267] Yuille, A. L., and Rangarajan, A. The Concave-Convex Procedure (CCCP). In *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]* (2001), T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds., MIT Press, pp. 1033–1040.

[268] Zakhlebin, I., and Horvát, E.-A. Diffusion of scientific articles across online platforms. In *Proceedings of the International AAAI Conference on Web and Social Media* (2020), vol. 14, pp. 762–773.

[269] Zhang, W. E., Sheng, Q. Z., Alhazmi, A., and Li, C. Adversarial Attacks on Deep Learning Models in Natural Language Processing: A Survey, 2019.

[270] Zhou, J., and Bhat, S. Paraphrase Generation: A Survey of the State of the Art. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021* (2021), M. Moens, X. Huang, L. Specia, and S. W. Yih, Eds., Association for Computational Linguistics, pp. 5075–5086.

[271] Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., and He, Q. A comprehensive survey on transfer learning. *Proc. IEEE 109*, 1 (2021), 43–76.

[272] Ziser, Y., and Reichart, R. Neural Structural Correspondence Learning for Domain Adaptation. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)* (2017), pp. 400–410.

[273] ZISER, Y., AND REICHART, R. Pivot based language modeling for improved neural domain adaptation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)* (2018), M. A. Walker, H. Ji, and A. Stent, Eds., Association for Computational Linguistics, pp. 1241–1251.

[274] ZUBIAGA, A., AKER, A., BONTCHEVA, K., LIAKATA, M., AND PROCTER, R. Detection and Resolution of Rumours in Social Media: A Survey. *ACM Computing Surveys (CSUR) 51*, 2 (2018), 1–36.

[275] ZUBIAGA, A., KOCHKINA, E., LIAKATA, M., PROCTER, R., LUKASIK, M., BONTCHEVA, K., COHN, T., AND AUGENSTEIN, I. Discourse-Aware Rumour Stance Classification in Social Media Using Sequential Classifiers. *Information Processing & Management 54*, 2 (2018), 273–290.

[276] ZUBIAGA, A., LIAKATA, M., AND PROCTER, R. Exploiting Context for Rumour Detection in Social Media. In *International Conference on Social Informatics* (2017), Springer, pp. 109–123.

[277] ZUBIAGA, A., LIAKATA, M., PROCTER, R., HOI, G. W. S., AND TOLMIE, P. Analysing How People Orient to and Spread Rumours in Social Media by Looking at Conversational Threads. *PloS one 11*, 3 (2016).

A Appendices for Claim Check-Worthiness Detection as Positive Unlabelled Learning

A.1 Examples of PUC Improvements for Rumour Detection

Examples of improvements for rumour detection using *PUC* can be found in Table A.1.

Rumour text	nPUC	nBaseline
Germanwings co-pilot had serious depressive episode: Bild newspaper http://t.co/RgSTrehD21	13	5
Now hearing 148 passengers + crew on board the #A320 that has crashed in southern French Alps. #GermanWings flight. @BBCWorld	10	2
It appears that #Ferguson PD are trying to assassinate Mike Brown's character after literally assassinating Mike Brown.	13	5
#Ferguson cops beat innocent man then charged him for bleeding on them: http://t.co/u1ot9Eh5Cq via @MichaelDalynyc http://t.co/AGJW2Pid1r	9	2

Table A.1: Examples of rumours which the *PUC* model judges correctly vs the baseline model with no pretraining on citation needed detection. n* is the number of models among the 15 seeds which predicted the correct label (rumour).

A.2 Reproducibility

A.2.1 Computing Infrastructure

All experiments were run on a shared cluster. Requested jobs consisted of 16GB of RAM and 4 Intel Xeon Silver 4110 CPUs. We used a single NVIDIA Titan X GPU with 12GB of RAM.

Non-Rumour text	nPUC	nBaseline
A female hostage stands by the front entrance of the cafe as she turns the lights off in Sydney. #sydneysiege http://t.co/qNfCMv9yZt	11	5
Map shows where gun attack on satirical magazine #CharlieHebdo took place in central Paris http://t.co/5AZAKumpNd http://t.co/ECFYztMVk9	10	4
"Hands up! Don't shoot!" #ferguson https://t.co/svCE1S0Zek	12	7
Australian PM Abbott: Motivation of perpetrator in Sydney hostage situation is not yet known - @9NewsAUS http://t.co/SI01B997xf	10	6

Table A.2: Examples of non-rumours which the *PUC* model judges correctly vs the baseline model with no pretraining on citation needed detection. n* is the number of models among the 15 seeds which predicted the correct label (non-rumour).

Method	Wikipedia	PHEME	Political Speeches
BERT	34m30s	14m25s	8m11s
BERT + PU	40m7s	20m40s	15m38s
BERT + *PUC*	40m8s	21m20s	15m32s
BERT + Wiki	-	14m28s	8m50s
BERT + WikiPU	-	14m25s	8m41s
BERT + Wiki*PUC*	-	14m28s	8m38s
BERT + PU + WikiPU	-	20m41s	15m32s
BERT + *PUC* + WikiPUC	-	21m52s	15m40s

Table A.3: Average runtime of each tested system for each split of the data

Method	Wikipedia	PHEME	Political Speeches
BERT	88.9	81.6	31.3
BERT + PU	89.0	83.7	18.2
BERT + *PUC*	89.2	82.8	32.0
BERT + Wiki	-	80.8	32.3
BERT + WikiPU	-	82.0	35.7
BERT + Wiki*PUC*	-	80.4	34.3
BERT + PU + WikiPU	-	82.9	33.3
BERT + *PUC* + WikiPUC	-	84.1	34.0

Table A.4: Validation F1 performances for each tested model.

A.2.2 Average Runtimes

See Table A.3 for model runtimes.

A.2.3 Number of Parameters per Model

We used BERT with a classifier on top for each model which consists of 109,483,778 parameters.

A.2.4 Validation Performance

Validation performances for the tested models are given in Table A.4.

A.2.5 Evaluation Metrics

The primary evaluation metric used was F1 score. We used the sklearn implementation of `precision_recall_fscore_support`, which can be found here:

`https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html`. Briefly:

$$p = \frac{tp}{tp + fp}$$

$$r = \frac{tp}{tp + fn}$$

$$F1 = \frac{2 * p * r}{p + r}$$

where tp are true positives, fp are false positives, and fn are false negatives.

Hyperparameter	Value
Learning Rate	3e-5
Weight Decay	0.01
Batch Size	8
Dropout	0.1
Warmup Steps	200
Epochs	2

Table A.5: Validation F1 performances used for each tested model.

Additionally, we used the mean average precision calculation from the Clef19 Check That! challenge for political speech data, which can be found here:

https://github.com/apepa/clef2019-factchecking-task1/tree/master/scorer Briefly:

$$\text{AP} = \frac{1}{|P|}\sum_i \frac{tp(i)}{i}$$

$$\text{mAP} = \frac{1}{|Q|}\sum_{q\in Q} \text{AP}(q)$$

where P are the set of positive instances, $tp(i)$ is an indicator function which equals one when the ith ranked sample is a true positive, and Q is the set of queries. In this work Q consists of the ranking of statements from each split of the political speech data.

A.2.6 Links to Data

- Citation Needed Detection [196]:

  https://drive.google.com/drive/folders/1zG6orf0_h2jYBvGvso1pSy3ikbNiW0xJ

- PHEME [277]:

  https://figshare.com/articles/PHEME_dataset_for_Rumour_Detection_and_Veracity_Classification/6392078.

- Political Speeches: We use the same 7 splits as used in [97]. The first 5 can be found here: http://alt.qcri.org/clef2018-factcheck/data/uploads/clef18_fact_checking_lab_submissions_and_scores_and_combinations.zip. The files can be found under "task1_test_set/English/task1-en-file(3,4,5,6,7)". The last two files can be found here: https://github.com/apepa/claim-rank/tree/master/data/transcripts_all_sources. The files are “clinton_acceptance_speech_ann.tsv” and “trump_inauguration _ann.tsv”.

A.2.7 Hyperparameters

We found that good defaults worked well, and thus did not perform hyperparameter search. The hyperparameters we used are given in Table A.5.

B Appendices for Generating Label Cohesive and Well-Formed Adversarial Claims

Class	Trigger	F1	STS	PPL
FC Objective				
S→R	only	0.014	4.628	11.660 (36.191)
S→R	nothing	0.017	4.286	13.109 (56.882)
S→R	nobody	0.036	4.167	12.784 (37.390)
S→NEI	neither	0.047	3.901	11.509 (31.413)
S→NEI	none	0.071	4.016	13.136 (39.894)
S→NEI	Neither	0.155	3.641	11.957 (44.274)
R→S	some	0.687	4.694	11.902 (33.348)
R→S	Sometimes	0.724	4.785	10.813 (32.058)
R→S	Some	0.743	4.713	11.477 (37.243)
R→NEI	recommended	0.658	4.944	12.658 (36.658)
R→NEI	Recommend	0.686	4.789	10.854 (32.432)
R→NEI	Supported	0.710	4.739	11.972 (40.267)
NEI→R	Only	0.624	4.668	12.939 (57.666)
NEI→R	nothing	0.638	4.476	11.481 (48.781)
NEI→R	nobody	0.678	4.361	16.345 (111.60)
NEI→S	nothing	0.638	4.476	18.070 (181.85)
NEI→S	existed	0.800	4.950	15.552 (79.823)
NEI→S	area	0.808	4.834	13.857 (93.295)
FC+STS Objectives				
S→R	never	0.048	4.267	12.745 (50.272)
S→R	every	0.637	4.612	13.714 (51.244)
S→R	didn	0.719	4.986	12.416 (41.080)
S→NEI	always	0.299	4.774	11.906 (35.686)
S→NEI	every	0.637	4.612	12.222 (38.440)
S→NEI	investors	0.696	4.920	12.920 (42.567)
R→S	over	0.761	4.741	12.139 (33.611)
R→S	about	0.765	4.826	12.052 (37.677)
R→S	her	0.774	4.513	12.624 (41.350)
R→NEI	top	0.757	4.762	12.787 (39.418)
R→NEI	also	0.770	5.034	11.751 (35.670)
R→NEI	when	0.776	4.843	12.444 (37.658)
NEI→R	only	0.562	4.677	14.372 (83.059)
NEI→R	there	0.764	4.846	11.574 (42.949)
NEI→R	just	0.786	4.916	16.879 (135.73)
NEI→S	of	0.802	4.917	11.844 (55.871)
NEI→S	is	0.815	4.931	17.507 (178.55)
NEI→S	A	0.818	4.897	12.526 (67.880)

Table B.1: Top-3 triggers found with the Universal Adversarial Triggers methods. The triggers are generated given claims from a source class (column *Class*), so that the classifier is fooled to predict a different target class. The classes are SUPPORTS (S), REFUTES (R), NOT ENOUGH INFO (NEI).

B.1 Implementation Details

Models. The RoBERTa FC model (125M parameters) is fine-tuned with a batch size of 8, learning rate of 2e-5 and for a total of 4 epochs, where the epoch with the best performance is saved. We used the implementation provided by HuggingFace library. We performed a grid hyper-parameter search for the learning rate between the values 1e-5, 2e-5, and 3e-5. The average time for training a model with one set of hyperparameters is 155 minutes (± 3). The average accuracy over the different hyperparameter runs is 0.862(± 0.005) F1 score on the validation set.

For the models that measure the perplexity and the semantical similarity we use the pretrained models provided by HuggingFace– RoBERTa large model (125M parameters) fine tuned on the STS-b task and RoBERTa base model (355M parameters) pretrained on a LM objective.

We used the HuggingFace implementation of the small GPT-2 model, which consists of 124M parameters and is fine-tuned with a batch size of 4, learning rate of 3e-5, and for a total of 20 epochs. We perform early stopping on the loss of the model on a set of validation data. The average validation loss is 0.910. The average runtime for training one of the models is 31 hours and 28 minutes.

We note that, the intermediate models used in this work and described in this section, are trained on large relatively general-purpose datasets. While, they can make some mistakes, they work well enough and using them, we don't have to rely on additional human annotations for the intermediate task.

Adversarial Triggers. The adversarial triggers are generated based on instances from the validation set. We run the algorithm for three epochs to allow for the adversarial triggers to converge. At each epoch the initial trigger is updated with the best performing trigger for the epoch (according to the loss of the FC or FC+STS objective). At the last step, we select only the top 10 triggers and remove any that have a negative loss. We choose the top 10 triggers as those are the most potent ones, adding more than top ten of the triggers preserves the same tendencies in the results, but smooths them as further down the list of adversarial attacks, the triggers do not decrease the performance of the model substantially. This is also supported by related literature [241], where only the top few triggers are selected.

The adversarial triggers method is run for 28.75 (± 1.47) minutes for with the FC objective and 168.6(± 28.44) minutes for the FC+STS objective. We perform the trigger generation with a batch size of four. We additionally normalize the loss for each objective to be in the range [0,1] and also re-weight the losses with a wieht of 0.6 for the FC loss and a weight of 0.4 for the SST loss as when generated with an equal weight, the SST loss tends to preserve the same initial token in all epochs.

Datasets The datasets used for training the FC model consist of 161,249 SUPPORTS, 60,227 REFUTES, and 69,885 NEI claims for the training split; 6,207 SUPPORTS, 6,235 REFUTES, and 6,554 NEI claims for the dev set; 6,291 SUPPORTS, 5,992 REFUTES, and 6522 NEI claims. The evidence for each claim is the gold evidence provided from the FEVER dataset, which is available for REFUTES and SUPPORTS claims. When there is more than one annotation of different evidence sentences for an instance, we include them as separate instances in the datasets. For NEI claims, we use the system of [156] to retrieve evidence sentences.

B.2 Top Adversarial Triggers

Table B.1 presents the top adversarial triggers for each direction found with the Universal Adversarial Triggers method. It offers an additional way of estimating the effectiveness of the STS objective by comparing the number of negation words generated by the basic model (8) and the STS objective (2) in the top-3 triggers for each direction.

B.3 Supplemental Material

B.3.1 Computing Infrastructure

All experiments were run on a shared cluster. Requested jobs consisted of 16GB of RAM and 4 Intel Xeon Silver 4110 CPUs. We used two NVIDIA Titan RTX GPUs with 12GB of RAM for training GPT-2 and one NVIDIA Titan X GPU with 8GB of RAM for training the FC models and finding the universal adversarial triggers.

B.3.2 Evaluation Metrics

The primary evaluation metric used was macro-F1 score. We used the sklearn implementation of `precision_recall_fscore_support`, which can be found here: `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html`. Briefly:

$$p = \frac{tp}{tp + fp}$$

$$r = \frac{tp}{tp + fn}$$

$$F1 = \frac{2 * p * r}{p + r}$$

where tp are true positives, fp are false positives, and fn are false negatives.

B.3.3 Manual Evaluation

After generating the claims, two independent annotators label the overall claim quality (score of 1-5) and the true label for the claim. The inter-annotator agreement for the quality label using Krippendorff's alpha is 0.54 for the quality score and 0.38 for the claim label. Given this, we take the average of the two annotator's scores for the final quality score and have a third expert annotator examine and select the best label for each contested claim label.

C Appendices for Transformer Based Multi-Source Domain Adaptation

C.1 BERT Domain Adversarial Training Results

Additional results on domain adversarial training with Bert can be found in Table C.1.

Method	Sentiment Analysis (Accuracy)					Rumour Detection (F1)					
	D	E	K	B	macroA	CH	F	GW	OS	S	μF1
Bert	90.3	91.6	91.7	90.4	91.0	66.4	46.2	68.3	67.3	62.3	63.3
Bert-Adv-12	89.8	91.4	91.2	90.1	90.6	66.6	47.8	62.5	65.3	62.8	62.5
Bert-Adv-4	89.9	91.1	91.7	90.4	90.8	65.6	43.6	71.0	68.1	60.8	62.8

Table C.1: Experiments for sentiment analysis in (D)VD, (E)lectronics, (K)itchen and housewares, and (B)ooks domains and rumour detection for different events ((C)harlie(H)ebdo, (F)erguson, (G)erman(W)ings, (O)ttawa(S)hooting, and (S)ydneySiege) using leave-one-out cross validation for BERT. Results are averaged across 3 random seeds. The results for sentiments analysis are in terms of accuracy and the results for rumour detection are in terms of F1.

C.2 Reproducibility

C.2.1 Computing Infrastructure

All experiments were run on a shared cluster. Requested jobs consisted of 16GB of RAM and 4 Intel Xeon Silver 4110 CPUs. We used a single NVIDIA Titan X GPU with 12GB of RAM.

C.2.2 Average Runtimes

The average runtime performance of each model is given in Table C.2. Note that different runs may have been placed on different nodes within a shared cluster, thus why large time differences occurred.

C.2.3 Number of Parameters per Model

The number of parameters in each model is given in Table C.3.

C.2.4 Validation Performance

The validation performance of each tested model is given in Table C.4.

Method	Sentiment Analysis	Rumour Detection
Basic	0h44m37s	0h23m52s
Adv-6	0h54m53s	0h59m31s
Adv-3	0h53m43s	0h57m29s
Independent-Avg	1h39m13s	1h19m27
Independent-Ft	1h58m55s	1h43m13
MoE-Avg	2h48m23s	4h03m46s
MoE-Att	2h49m44s	4h07m3s
MoE-Att-Adv-6	4h51m38s	4h58m33s
MoE-Att-Adv-3	4h50m13s	4h54m56s
MoE-DC	3h23m46s	4h09m51s

Table C.2: Average runtimes for each model on each dataset (runtimes are taken for the entire run of an experiment).

Method	Sentiment Analysis	Rumour Detection
Basic	66,955,010	66,955,010
Adv-6	66,958,082	66,958,850
Adv-3	66,958,082	66,958,850
Independent-Avg	267,820,040	334,775,050
Independent-Ft	267,820,040	334,775,050
MoE-Avg	267,820,040	334,775,050
MoE-Att	268,999,688	335,954,698
MoE-Att-Adv-6	269,002,760	335,958,538
MoE-Att-Adv-3	269,002,760	335,958,538
MoE-DC	267,821,576	334,777,354

Table C.3: Number of parameters in each model

C.2.5 Evaluation Metrics

The primary evaluation metrics used were accuracy and F1 score. For accuracy, we used our implementation provided with the code. The basic implementation is as follows.

$$\text{accuracy} = \frac{tp + tn}{tp + fp + tn + fn}$$

We used the sklearn implementation of `precision_recall_fscore_support` for F1 score, which can be found here: `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html`. Briefly:

$$p = \frac{tp}{tp + fp}$$

$$r = \frac{tp}{tp + fn}$$

$$F1 = \frac{2 * p * r}{p + r}$$

where tp are true positives, fp are false positives, and fn are false negatives.

Method	Sentiment Analysis (Acc)	Rumour Detection (F1)
Basic	91.7	82.4
Adv-6	91.5	83.3
Adv-3	91.2	83.4
Independent-Avg	92.7	82.8
Independent-Ft	92.6	82.5
MoE-Avg	92.2	83.5
MoE-Att	92.0	83.3
MoE-Att-Adv-6	91.2	83.3
MoE-Att-Adv-3	91.4	82.8
MoE-DC	89.8	84.6

Table C.4: Average validation performance for each of the models on both datasets.

C.2.6 Hyperparameters

We performed and initial hyperparameter search to obtain good hyperparameters that we used across models. The bounds for each hyperparameter was as follows:

- Learning rate: [0.00003, 0.00004, 0.00002, 0.00001, 0.00005, 0.0001, 0.001].
- Weight decay: [0.0, 0.1, 0.01, 0.005, 0.001, 0.0005, 0.0001].
- Epochs: [2, 3, 4, 5, 7, 10].
- Warmup steps: [0, 100, 200, 500, 1000, 5000, 10000].
- Gradient accumulation: [1,2]

We kept the batch size at 8 due to GPU memory constraints and used gradient accumulation instead. We performed a randomized hyperparameter search for 70 trials. Best hyperparameters are chosen based on validation set performance (accuracy for sentiment data, F1 for rumour detection data). The final hyperparameters selected are as follows:

- Learning rate: 3e-5.
- Weight decay: 0.01.
- Epochs: 5.
- Warmup steps: 200.
- Batch Size: 8
- Gradient accumulation: 1

Additionally, we set the objective weighting parameters to $\lambda = 0.5$ for the mixture of experts models and $\gamma = 0.003$ for the adversarial models, in line with previous work [92, 142].

C.2.7 Links to data

- Amazon Product Reviews [30]: https://www.cs.jhu.edu/~mdredze/datasets/sentiment/
- PHEME [277]: https://figshare.com/articles/PHEME_dataset_for_Rumour_Detection_and_Veracity_Classification/6392078.

D Appendices for Multi-View Knowledge Distillation from Crowd Annotations for Out of Domain Generalization

D.1 Evaluation Metrics

F1 We used the sklearn implementation of precision_recall_fscore_support for F1 score, which can be found here: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html. Briefly:

$$p = \frac{tp}{tp + fp}$$

$$r = \frac{tp}{tp + fn}$$

$$F1 = \frac{2 * p * r}{p + r}$$

where tp are true positives, fp are false positives, and fn are false negatives.

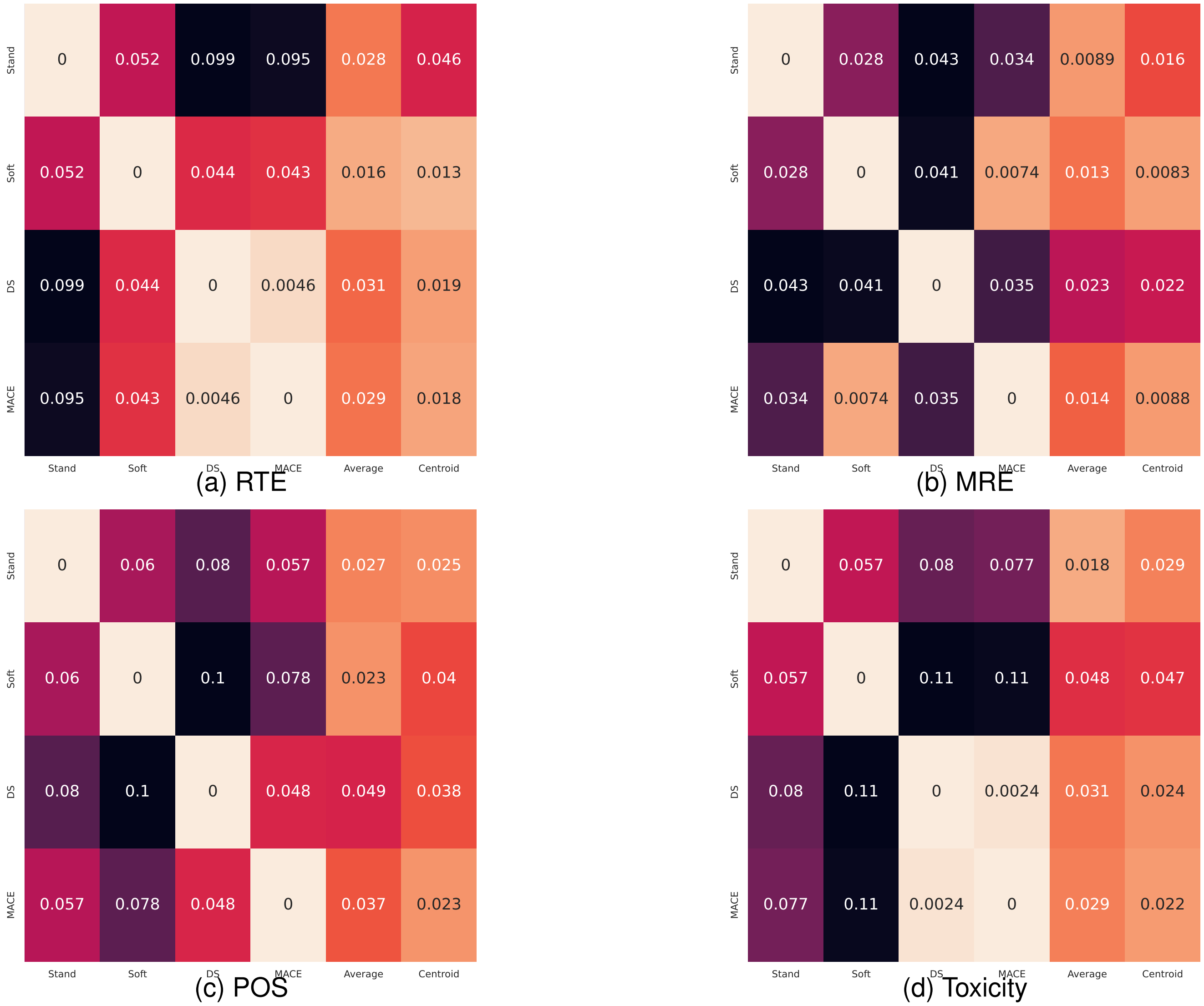
Calibrated Log-Likelihood The calibrated log-likelihood is defined in [7] as a method to fairly compare uncertainty estimation between models on the same test set. The key observation is that in order to obtain a fair comparison, one must first perform temperature scaling at the optimal temperature on the classifier output for each model under comparison. Additionally, this temperature must be optimized on an in-domain validation set. The procedure to calculate the calibrated log-likelihood is:

1. Split the **test set** in half, one half for validation and one half for test.
2. Optimize a temperature parameter T to minimize the average negative log-likelihood $-\frac{1}{n}\sum_i \log \tilde{p}(y_i = y_i^* | x_i)$, where $\tilde{p}_i =$ softmax$(\frac{l_i}{T})$ and l_i is the logits of the classifier, on the validation half of the test set.
3. Measure the temperature scaled log-likelihood on the test half of the test set.

Following the suggestion from [7], we run this procedure 5 times on different splits of the test set and take the average test-half log-likelihood as the result.

D.2 Visualization

Here we plot the JSD between individual methods and the averaging and JSC methods for each dataset in Figure D.1.



(a) RTE

(b) MRE

(c) POS

(d) Toxicity

Figure D.1: Heatmaps of the average Jensen-Shannon divergence between individual soft labeling methods and average and JS centroid aggregation for (a) RTE, (b) MRE, (c) POS, and (d) Toxicity datasets.

E Appendices for CiteWorth: Cite-Worthiness Detection for Improved Scientific Document Understanding

E.1 List of Permissible Section Titles

- introduction
- abstract
- method
- methods
- results
- discussion
- discussions
- conclusion
- conclusions
- results and discussion
- related work
- experimental results
- literature review
- experiments
- background
- methodology
- conclusions and future work
- related works
- limitations
- procedure
- material and methods
- discussion and conclusion
- implementation
- evaluation
- performance evaluation
- experiments and results
- overview
- experimental design
- discussion and conclusions
- results and discussions
- motivation
- proposed method

- analysis
- future work
- results and analysis
- implementation details

E.2 List of Regular Expressions

Citation format regexes:

- `\[([0-9]+\s*[,-;]*\s*)*[0-9]+\s*\]`
- `\(?[12][0-9]3[a-z]?\s*\)`

Hanging citation regex:

```
\s+\(?(\(\s*\)|like|reference|
including|include|with|for instance|for example|see also|at|following|of|from|to|in|by|
see|as|e\.?g\.?(,)?|viz(\.)?(,)?)\s*
(,)*(-)*[\)\]]?\s*[.?!]\s*$
```

E.3 Reproducibility

E.3.1 Computing Infrastructure

All experiments were run on a shared cluster. Requested jobs consisted of 16GB of RAM and 4 Intel Xeon Silver 4110 CPUs. We used a single NVIDIA Titan X GPU with 12GB of RAM.

E.3.2 Average Runtimes

The average runtime performance of each model is given in Table E.1. Note that different runs may have been placed on different nodes within a shared cluster.

Setting	Time
Logistic Regression	00h01m43s
Transformer	02h55m13s
BERT	05h30m30s
SciBERT (no weighting)	09h22m00s
SciBERT	09h32m37s
SciBERT + PU	16h01m27s
Longformer-Solo	75h27m22s
Longformer-Ctx	19h16m07s

Table E.1: Average runtimes for each model (runtimes are taken for the entire run of an experiment).

E.3.3 Number of Parameters per Model

The number of parameters in each model is given in Table E.2.

Method	# Parameters
Logistic Regression	198,323
Transformer	9,789,042
BERT	109,484,290
SciBERT	109,920,514
Longformer	149,251,586

Table E.2: Number of parameters in each model

E.3.4 Validation Performance

The validation performance of each tested model is given in Table E.3.

Method	F1
Logistic Regression	-
Transformer	57.02
BERT	60.75
SciBERT (no weighting)	57.52
SciBERT	62.04
SciBERT + PU	61.43
Longformer-Solo	61.67
Longformer-Ctx	67.11

Table E.3: Average validation performance for each of the models.

E.3.5 Evaluation Metrics

The primary evaluation metric used was F1 score. We used the sklearn implementation of `precision_recall_fscore_support` for F1 score, which can be found here: `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html`. Briefly:

$$p = \frac{tp}{tp + fp}$$

$$r = \frac{tp}{tp + fn}$$

$$F1 = \frac{2 * p * r}{p + r}$$

where tp are true positives, fp are false positives, and fn are false negatives.

E.3.6 Hyperparameters

Logistic Regression We used a C value of 0.1151 for logistic regression.

Basic Transformer The final hyperparameters for the basic Transformer model are: batch size: 64; number of epochs: 33; feed-forward dimension: 128; learning rate: 0.0001406; number of heads: 3; number of layers: 5; weight decay: 0.1; dropout probability: 0.4. We performed a Bayesian grid search over the following ranges of values, optimizing validation F1 performance: learning rate: $[0.000001, 0.001]$; batch size: $\{4, 8, 16, 32, 64, 128\}$; weight decay: $\{0.0, 0.0001, 0.001, 0.01, 0.1\}$; dropout probability: $\{0.0, 0.1, 0.2, 0.3, 0.4, 0.5\}$; number of epochs: $[2, 40]$; feed-forward dimension: $\{128, 256, 512, 1024, 2048\}$; number of heads: $\{1, 2, 3, 4, 5, 6, 10, 12\}$; number of layers: $[1, 12]$.

BERT The final hyperparameters for BERT are: batch size: 8; number of epochs: 3; learning rate: 0.000008075; triangular learning rate warmup steps: 300; weight decay: 0.1; dropout probability: 0.1. We performed a Bayesian grid search over the following ranges of values, optimizer validation F1 performance: learning rate: $[0.0000001, 0.0001]$; triangular learning rate warmup steps: $\{0, 100, 200, 300, 400, 500, 1000, 1500, 2000, 2500, 5000\}$; batch size: $\{4, 8\}$; weight decay: $\{0.0, 0.0001, 0.001, 0.01, 0.1\}$; number of epochs: $[2, 40]$.

SciBERT The final hyperparameters for SciBERT are: batch size: 4; number of epochs: 3; learning rate: 0.000001351; triangular learning rate warmup steps: 300; weight decay: 0.1; dropout probability: 0.1. We performed a Bayesian grid search over the following ranges of values, optimizer validation F1 performance: learning rate: $[0.0000001, 0.0001]$; triangular learning rate warmup steps: $\{0, 100, 200, 300, 400, 500, 1000, 1500, 2000, 2500, 5000\}$; batch size: $\{4, 8\}$; weight decay: $\{0.0, 0.0001, 0.001, 0.01, 0.1\}$; number of epochs: $[2, 40]$.

Longformer-Ctx The final hyperparameters for Longformer-Ctx are: batch size: 4; number of epochs: 3; learning rate: 0.00001112; triangular learning rate warmup steps: 300; weight decay: 0.0; dropout probability: 0.1. We performed a Bayesian grid search over the following ranges of values, optimizer validation F1 performance: learning rate: $[0.0000001, 0.0001]$; triangular learning rate warmup steps: $\{0, 100, 200, 300, 400, 500, 1000, 1500, 2000, 2500, 5000\}$; batch size: $\{4, 8\}$; weight decay: $\{0.0, 0.0001, 0.001, 0.01, 0.1\}$; number of epochs: $[2, 6]$.

E.3.7 Data

CITEWORTH is constructed from the S2ORC dataset, which can be found here: `https://github.com/allenai/s2orc`. In particular, CITEWORTH is built using the `20200705v1` release of the data. A link to the CITEWORTH data can be found here: `https://github.com/copenlu/cite-worth`.

Model	# Params
RoBERTa	125M
BART	140M
GPT-2	125M
Longformer-SciFact	438M

Table F.1: Model sizes.

F Appendices for Generating Scientific Claims for Zero-Shot Scientific Fact Checking

F.1 Reproducibility

F.1.1 Computing Infrastructure

All experiments were run on an Amazon Web Services p3.2xlarge instance using a Tesla V100 GPU with 16GB of RAM.

F.1.2 Number of Parameters per Model

The sizes of each of the models used in this work are given in Table F.1.

F.1.3 Hyperparameters

F.1.3.1 Fact Checking

SciFact data Learning rate: 1e-5, 5 epochs, gradient accumulation for 8 batches, 1 sample per training batch, 16-bit precision, 809 total claims.

FEVER threshold We tune the NEI threshold on the training set of SciFact, testing values in the range [1e-5, 2e-5, 3e-5, 4e-5, 5e-5, 1e-4, 2e-4, 3e-4, 4e-4, 5e-4, 1e-3, 2e-3, 3e-3, 4e-3, 5e-5, 0.01, 0.12, 0.2, 0.25, 0.4, 0.5, 0.75, 0.8, 0.8, 0.99, 0.999] and find that 5e-5 produces the best result.

CLAIMGEN-BART Learning rate: 2e-6, 5 epochs, gradient accumulation for 8 batches, 1 sample per training batch, 16-bit precision, 1,561 total training claims.

CLAIMGEN-ENTITY Learning rate: 4e-8, 5 epochs, gradient accumulation for 8 batches, 1 sample per training batch, 16-bit precision, 8,592 total training claims.

F.1.3.2 ClaimGen-BART

Learning rate: 2e-5, 3 epochs, linear warmup for 200 steps followed by linear decay, weight decay of 0.01, batch size of 8.

F.1.4 Description of Datasets

We use a variety of datasets in this study for different components of models, training, and testing. Here we provide a description of each and in which module the dataset is used.

SciFact The SciFact dataset and rewritten claims used to train ClaimGen-BART can be found at https://github.com/allenai/scifact. The dataset consists of 585 original citances with rewritten claims for each of them. Each citance consists of 1-2 rewritten claims. The SciFact rewritten claims are used to train ClaimGen-BART for direct claim generation. Additionally, SciFact contains biomedical claims paired with evidence abstracts and veracity labels in {*supports*, *refutes*, *not enough info*} and is split into train, dev, and test sets. We use the train set for supervised fact checking experiments, and the dev set for testing since the test set does not come with labels.

FEVER FEVER is a general domain fact checking dataset built from Wikipedia. Like SciFact, the dataset consists of claims with paired evidence documents with labels in {*supports*, *refutes*, *not enough info*}. FEVER is used as pretraining data for our fact checking models for zero-shot transfer to biomedical claims. The dataset can be found here https://fever.ai/resources.html.

MedMentions The MedMentions dataset is a dataset of 4,392 biomedical papers annotated with mentions of UMLS entities. It is used to train the named entity recognition and normalization models used by ScispaCy, which we used for named entity recognition in ClaimGen-entity and for normalization in KBIN. The dataset can be found at https://github.com/chanzuckerberg/MedMentions

UMLS The UMLS meta-thesaurus is a large biomedical knowledge base which unifies hundreds of different ontologies in biomedicine. UMLS is used as the source knowledge base for normalization and candidate selection for KBIN. Additionally, it is the knowledge base used to train `cui2vec`, which is used for candidate concept selection in KBIN. UMLS can be found here https://www.nlm.nih.gov/research/umls/index.html.

SQuAD The SQuAD dataset can be found at: https://rajpurkar.github.io/SQuAD-explorer/. SQuAD is used as training data for the question generation module of ClaimGen-entity.

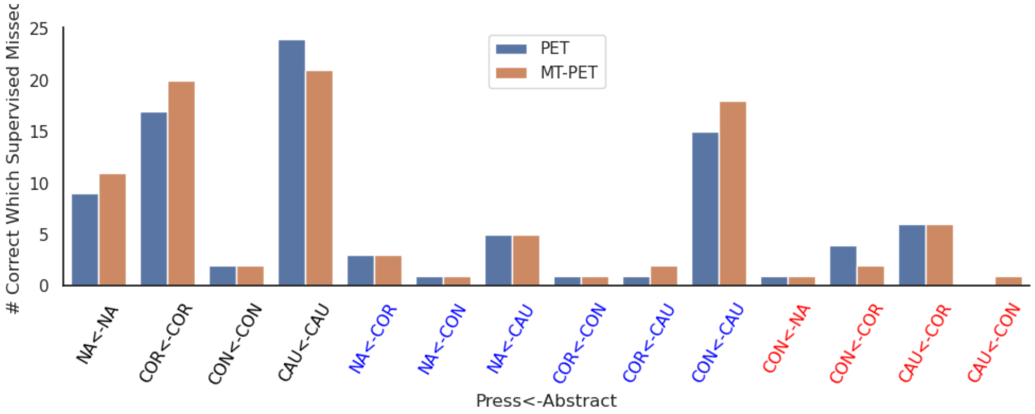


Figure G.1: Number of instances that each model predicted correctly which the supervised model predicted incorrectly.

SQuAD is a question answering dataset which contains data of the form (q, c, a), where q is the question, c is a context document, and a is an answer to the question which can be found in the context.

QA2D The QA2D dataset can be found here[32]. QA2D is used in the second part of the zero-shot CLAIMGEN-ENTITY model to generate declarative sentences from questions. It consists of data of the form (s, q, a) where q is a question, a is the answer to the question, and s is the declarative form of the question containing the answer.

MNLI MNLI is a crowd-sourced collection of 433k sentence pairs annotated for textual entailment. In other words, the data consists of pairs (p, h), where p is the premise and h is the hypothesis, and labels in {*entailment*, *contradiction*, *neutral*} which say if the hypothesis entails, contradicts, or is neutral towards the premise. MNLI is used to train a RoBERTa model for entailment, which is used by KBIN to select the best negation among a set of generated claims for a given source citance. The dataset can be found here https://cims.nyu.edu/ sbowman/multinli/

G Appendices for Semi-Supervised Exaggeration Detection of Health Science Press Releases

G.1 Error Analysis Plots

Extra plots from our error analysis are given in Figure G.1 and Figure G.2.

[32] https://worksheets.codalab.org/worksheets/ 0xd4ebc52cebb84130a07cbfe81597aaf0/

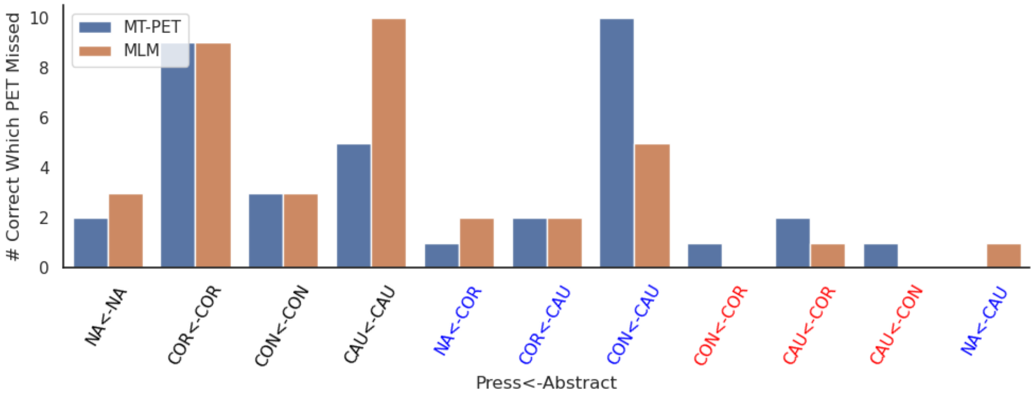


Figure G.2: Number of instances that each model predicted correctly which PET predicted incorrectly.

G.2 Reproducibility

G.2.1 Computing Infrastructure

All experiments were run on a shared cluster. Requested jobs consisted of 16GB of RAM and 4 Intel Xeon Silver 4110 CPUs. We used a single NVIDIA Titan X GPU with 24GB of RAM.

G.2.2 Average Runtimes

The average runtime performance of each model is given in Table G.1. Note that different runs may have been placed on different nodes within a shared cluster.

Setting	|**T1**|,|**T2**|	Time
Supervised	100,0	00h01m28s
PET	100,0	00h11m14s
MT-PET	100,200	00h13m05s
Supervised	0,200	00h01m20s
PET	0,200	00h16m22s
MT-PET	100,200	00h18m43s
Supervised	0,4500	00h03m23s
PET	0,4500	00h40m23s
MT-PET	100,4500	00h31m48s

Table G.1: Average runtimes for each model (runtimes are taken for the entire run of an experiment).

G.2.3 Number of Parameters per Model

We use RoBERTa, specifically the base model, for all experiments, which consists of 124,647,170 parameters.

G.2.4 Validation Performance

As we are testing a few shot setting, we do not use a validation set and only report the final test results.

G.2.5 Evaluation Metrics

The primary evaluation metric used was macro F1 score. We used the sklearn implementation of `precision_recall_fscore_support` for F1 score, which can be found here: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html. Briefly:

$$p = \frac{tp}{tp + fp}$$

$$r = \frac{tp}{tp + fn}$$

$$F1 = \frac{2 * p * r}{p + r}$$

where tp are true positives, fp are false positives, and fn are false negatives. Macro F1 is average F1 across all classes.

G.2.6 Hyperparameters

T1 Hyperparameters Supervised/PET training We used the following hyperparameters for experiments with **T1** as the main task: Epochs: 10; Batch Size: 4; Learning Rate: 0.00005598; Warmup Steps: 50; Weight decay: 0.001. We also weigh the cross-entropy loss based on the label distribution. These hyperparameters are found by performing a hyperparameter search using 4-fold cross validation on the 100 training examples. The bounds are as follows: Learning rate: $[0.000001, 0.0001$; Warmup steps: $\{0, 10, 20, 30, 40, 50, 100\}$; Batch size: $\{4, 8\}$; Weight decay: $\{0.0, 0.0001, 0.001, 0.01, 0.1\}$; Epochs: $[2, 10]$.

T2 Hyperparameters Supervised/PET training Epochs: 10; Batch Size: 4; Learning Rate: 0.00003; Warmup Steps: 50; Weight Decay: 0.001. We also weigh the cross-entropy loss based on the label distribution.

Hyperparameters for Distillation We used the following hyperparameters for distillation (training the final classifier after PET) for both **T1** and **T2** as the main task: Epochs: 3; Batch Size: 4; Learning Rate: 0.00001; Warmup Steps: 200; Weight decay: 0.01; Temperature: 2.0. We also weigh the cross-entropy loss based on the label distribution.

Sentence 1	Sentence 2
The polar bear is sliding on the snow.	A polar bear is sliding across the snow.
A plane is taking off	An air plane is taking off
A dog rides a skateboard	A dog is riding a skateboard
A man is playing the drums	A man plays the drum

Table H.1: Samples of sentence pairs in STSB which have a similarity score of 5

G.2.7 Data

We build our benchmark test dataset from the studies of [224] and [38]. The original data can be found at https://figshare.com/articles/dataset/InSciOut/903704 and https://osf.io/v8qhn/files/. A link to the test data will be provided upon acceptance of the paper (and included in the supplemental material). Claim strength data from [265] for abstracts can be found at https://github.com/junwang4/correlation-to-causation-exaggeration/blob/master/data/annotated_pubmed.csv. Claim strength data for press releases from [266] can be found at https://github.com/junwang4/correlation-to-causation-exaggeration/blob/master/data/annotated_eureka.csv

H Appendices for Modeling Information Change in Science Communication with Semantically Matched Paraphrases

H.1 Information Change vs. Semantic Similarity

We wish to highlight key differences between information change and semantic similarity, particularly with an eye to what makes the task introduced in SPICED difficult compared to semantic similarity scoring. To illustrate this, we present a sample of pairs in STSB that have the highest similarity score of '5' vs. samples in SPICED which have an IMS of 5 in Table H.1 and Table H.2.

In this, for a pair to be perfectly similar from a semantics perspective, the entire sentence must contain exactly equivalent meaning. This is not the case with our task. For the information change task, pairs are highly similar even if some aspects of the semantics of the sentence are changed e.g. in the first sample, there is a difference between the two sentences semantically: the second in the pair discusses "being intrigued" by the finding, which is shared between the pair. This also makes the task extremely difficult – a model must learn to compare only the salient scientific facts between the pair of sentences, as opposed to the entire meaning of each

Sentence 1	**Sentence 2**
Higher-income professionals had less tolerance for smartphone use in business meetings.	We are intrigued by the result that professionals with higher incomes are less accepting of mobile phone use in meetings.
If we allow people to retract recently posted comments, then we may be able to minimize regret from posting in the heat of the moment.	Allowing users to retract recently posted comments may help minimize regret .
Papers with shorter titles get more citations #science #metascience #sciencemetrics	Our analysis suggests that papers with shorter titles do receive greater numbers of citations.
Low levels of self-esteem and poor emotional processing skills were significantly correlated with gang involvement, as were low levels of parental monitoring, poor parental communication and housing instability.	Major findings also indicated that low levels of parental monitoring, poor parental communication and housing instability were significantly associated with gang involvement.

Table H.2: Samples of sentence pairs in SPICED which have an IMS of 5.

sentence.

H.2 Pilot Annotation Details

For the pilot, we use 20 pairs from 20 different cosine similarity score bins in increments of 0.05 starting from 0. In other words, we have 20 bins with ranges of scores as: $0.0 - 0.05$, $0.05 - 0.1 ... 0.9 - 0.95$, $0.95 - 1.0$. This results in 400 samples to annotate. The score distribution from 7,392,690 pairs from 3,525 source papers which we use for sampling is given in Figure H.1. Each sample is annotated by two of the authors of the study with a binary label of "matching" vs "not matching", yielding a Krippendorff's alpha of 0.73.

The number of positive samples per bin from the pilot study is given in Figure H.2. We see here that bins with a cosine similarity below 0.65 tend to have very few positive samples, and only above 0.8 do we start to see many positive samples in the bins. Almost all samples above 0.9 are matching, and the only unmatched pairs appear to be instances of SBERT failing, since the matched pairs are almost exactly copied text. Additionally, this histogram indicates that

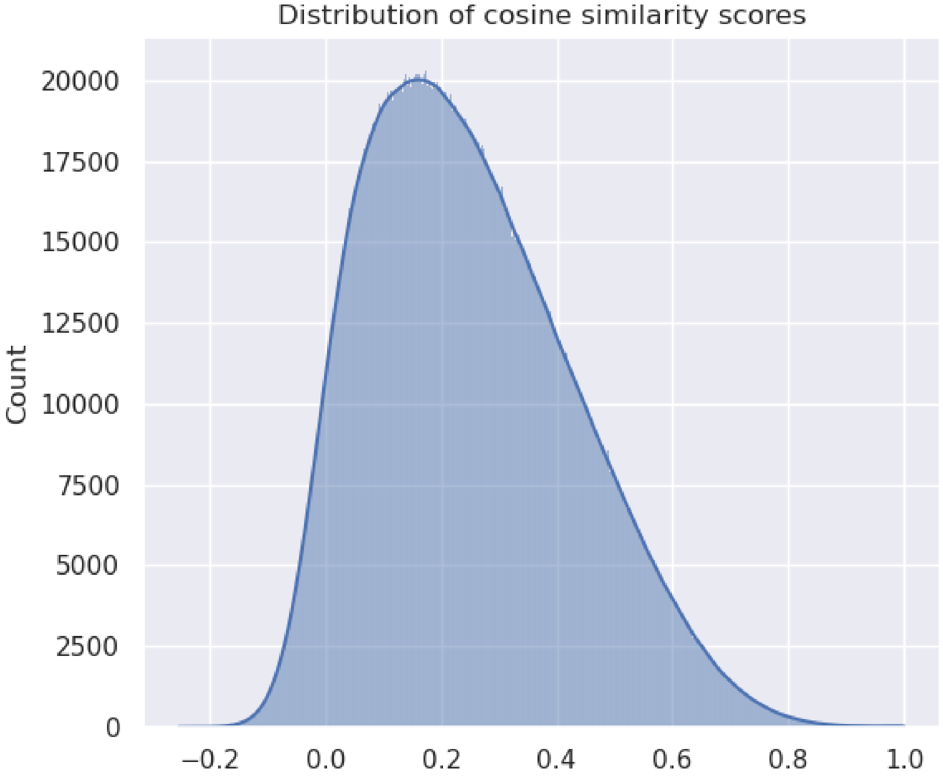


Figure H.1: Distribution of the cosine similarity between findings extracted from news articles about particular scientific papers. Cosine similarity is measured between the embeddings produced for both findings using SBERT [197].

the base rate of positive matching findings is low as the overall distribution of samples in the high cosine similarity region, where most of the matches exist, is small. At the same time, we note that some of the matches we find in the lower cosine similarity regions constitute quite interesting samples; for example, the following which has a cosine similarity of 0.41.

> **Paper finding:** For cases comparing a drone and a vehicle carrying a single package over similar distances, for example, a customer picking up a package from a retail store, the drone is clearly a lower-impact solution. [222]
>
> **News finding:** But if you forgot that essential ingredient for tonight's dinner, our findings suggest it's much better to have the grocery store send it to you by drone rather than to take your car to the store and back.[33]

Both sentences are talking about the same finding, that drone delivery is more efficient over short distances than using a car, but in entirely different ways. From this, it is clear that simply using semantic text similarity is insufficient for solving this task, and we should include some of

[33] https://www.enbridge.com/energy-matters/news-and-views/delivering-packages-with-drones-might-be-good-for-the-environment

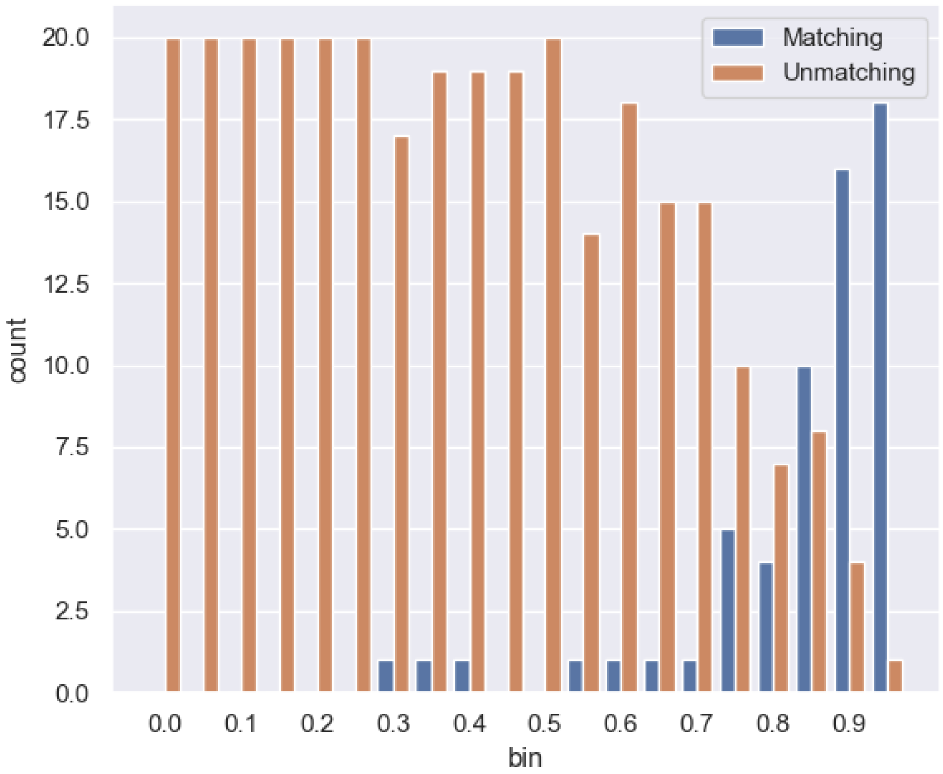


Figure H.2: Number of samples per bin rated as matching vs. not matching (samples limited to those where both annotators agreed on the label). Most matching samples come from higher similarity bins, while more difficult samples come from the middle bins.

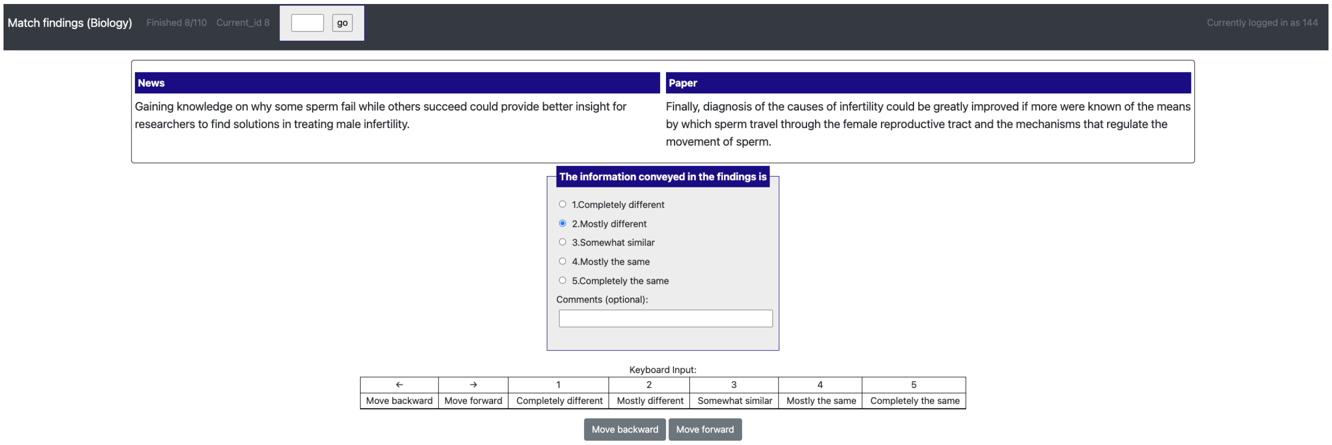
these lower similarity samples in our annotation. We, therefore, propose the following sampling scheme in order to balance the number of annotations we can acquire, the yield of positive samples, and the sample difficulty:

- Label all samples with a cosine similarity below 0.4 as unmatched.
- Label all samples above 0.9 with a Jaccard index above 0.5 as matching.
- Sample an equal number of pairs from each 0.05 increment bin between 0.4 and 0.9 for human expert annotation.

H.3 Experimented annotation

We experimented with two annotation schemas: a binary schema where the annotators are asked to label “whether the two sentences are discussing the same scientific finding” with Yes or No, and a Likert schema where the annotators are asked to label if “The information in the findings is...”

- 1: Completely different
- 2: Mostly different
- 3: Somewhat similar



News

Gaining knowledge on why some sperm fail while others succeed could provide better insight for researchers to find solutions in treating male infertility.

Paper

Finally, diagnosis of the causes of infertility could be greatly improved if more were known of the means by which sperm travel through the female reproductive tract and the mechanisms that regulate the movement of sperm.

The information conveyed in the findings is

○ 1.Completely different
● 2.Mostly different
○ 3.Somewhat similar
○ 4.Mostly the same
○ 5.Completely the same

Comments (optional):

Keyboard Input:

←	→	1	2	3	4	5
Move backward	Move forward	Completely different	Mostly different	Somewhat similar	Mostly the same	Completely the same

Figure H.3: The annotation page of our crowdsourcing task

- 4: Mostly the same
- 5: Completely the same

We ran several pilots using the two annotation schemas and the Likert a schema led to higher inter-annotator agreement (0.45 Krippendorff's alpha) compared with the binary schema (0.21 Krippendorff's alpha). Therefore we adopt the 5-point Likert schema for the annotation.

H.4 Full Annotation Instructions

Annotation was performed using Prolific workers who labeled using POTATO [178]. The annotation interface setup is available at `https://github.com/davidjurgens/potato/tree/master/example-projects/match_finding` which includes all the following instructions as well.

Task description: The task is to label to what degree two sentences have the same information. The information in the sentences is scientific findings. Here, a scientific finding is a statement that describes a research output of a scientific study, such as a result, conclusion, product, etc. You should rate how similar the findings are; you can ignore extra information like "The researchers showed...", "In vivo experiments demonstrated..." etc. For example, in the sentence "After controlling for weight and age, researchers found that overconsumption of sugar is linked with an increase in diabetes," the information in the finding is "overconsumption of sugar is linked with an increase in diabetes". Some sentences may have no findings or multiple findings, so use your best judgment about what are the core findings being said.

You will rate this on a 5-point scale, where each level means the following:

1. The information in the findings is completely different
    - Sentences in this category have findings which say completely different information
    - The sentences may be on totally different topics

    - Overconsumption of sugar causes diabetes
    - Regular exercise improves heart health
  - There may be some overlap in key words used between the two sentences, but the actual information is completely different
    - Chocolate contains a lot of sugar, and therefore can have an effect on weight.
    - Overconsumption of sugar leads to diabetes.

2. The information in the findings is mostly different
   - The findings may talk about the same topic, but the actual information is mostly different; for example, these sentences convey mostly different information even though they talk about the same topic:
     - Overconsumption of sugar causes diabetes
     - Sugar is good for your health
   - There could be a link between the two findings, but the information conveyed is still different
     - Overconsumption of sugar increases blood glucose levels
     - High blood glucose over time increases the risk of developing diabetes

3. The information in the findings is somewhat similar
   - The findings are discussing relevant research outputs but there are some differences in the information conveyed. Here the difference is that (i) talks about the relationship between overconsumption of sugar and diabetes and (ii) describes how genetics plays a role in overconsumption of sugar
     - Overconsumption of sugar causes diabetes
     - Overconsumption of sugar might be genetically determined

4. The information in the findings is mostly the same
   - In this case there may be some changes in e.g. the level of generality. Additionally, one sentence may go into more detail than the other and add additional context, but the information is largely the same
   - Here the two findings have the same information but at different levels of generality:
     - A link between sugar and diabetes was found
     - Overconsumption of sugar is associated with the onset of diabetes
   - Here both sentences have the same core finding, but one sentence goes into more detail

Metric	Papers	Overall	News	Tweets
Unique tokens	11047	12139	10203	5037
RTTR	32.01	36.59	33.48	38.46
MTLD	152.64	185.35	176.53	259.88
HDD	0.89	0.90	0.89	0.92

Table H.3: Various measures of lexical richness and diversity between findings in papers and other sources. RTTR is the root token-type ratio; MTLD is measure of textual lexical diversity [158]; HDD is the hypergeometric distribution diversity [158].

- Overconsumption of sugar causes diabetes
- Experiments demonstrated that overconsumption of sugar led to an increase in blood glucose levels, which over a long enough time period was linked to an increased prevalence of diabetes in the cohort.

- One finding could support the other
  - Overconsumption of sugar causes diabetes
  - Overconsumption of sugar can have negative effects on health

5. The information in the findings is completely the same

   - In this case there is complete overlap in the information in the findings conveyed by the two sentences
     - Overconsumption of sugar leads to diabetes.
     - The researchers found that overconsumption of sugar leads to diabetes
   - Note that there can be changes in e.g. the level of certainty or the strength of the information.
     - Overconsumption of sugar leads to diabetes.
     - It is likely that there is a link between overconsumption of sugar and the onset of diabetes.

H.5 Final dataset details

Figure H.4 shows the IMS distribution in SPICED. Figure H.5 shows the IMS distribution for annotated pairs in SPICED. Figure H.6 shows the IMS distribution for each split.

We measure various aspects of lexical richness between the different domains of the data in Table H.3.

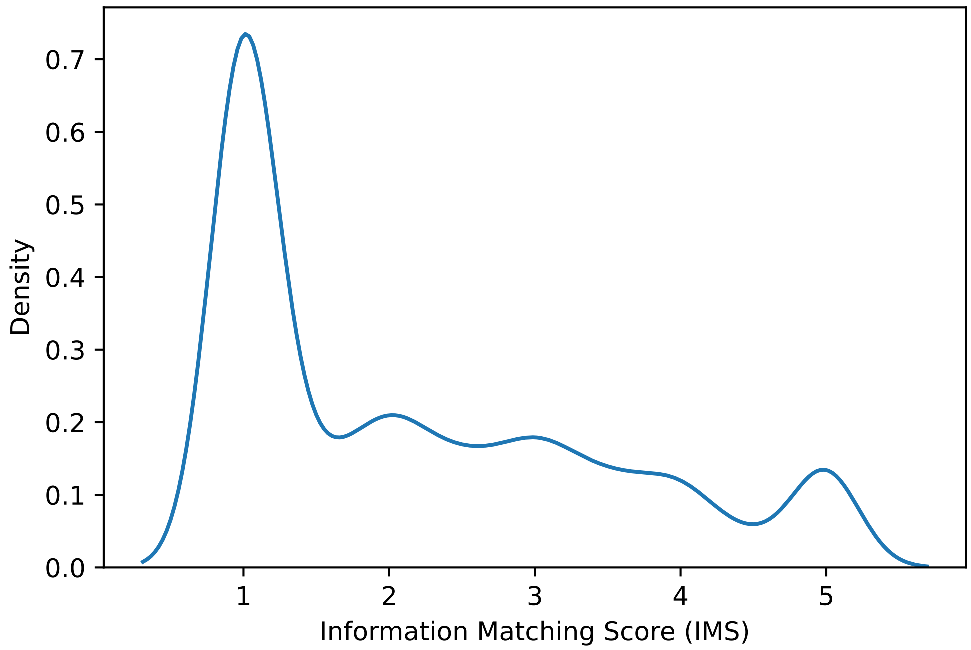


Figure H.4: Distribution of the final matching score in SPICED, which includes some pairs of scientific findings that are automatically labeled based on their extreme textual similarity (high or low), in addition to the annotated pairs.

H.6 Metrics

Average Normalized Edit Distance We calculate the normalized edit distance as follows:

$$d_N = \frac{1}{|D|} \sum_i \frac{d(s_1^{(i)}, s_2^{(i)})}{\max(|s_1^{(i)}|, |s_2^{(i)}|)}$$

where $|D|$ is the size of the dataset, $(s_1^{(i)}, s_2^{(i)})$ is a sentence pair, and d is the edit distance.

Jaccard Index The Jaccard index is calculated based on the overlap of the members of two sets (e.g. the words in two sentences X and Y):

$$J = \frac{|X \cap Y|}{|X \cup Y|}$$

Cosine Similarity The cosine similarity between two vectors **a** and **b** is calculated as:

$$S_C(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \|\mathbf{b}\|}$$

Which is their dot product divided by the product of their lengths.

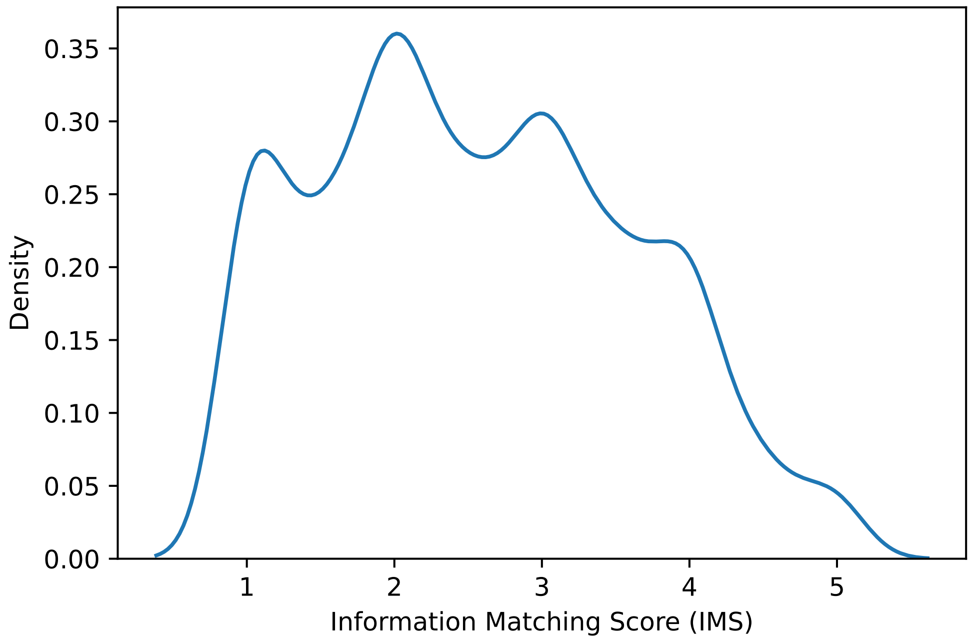


Figure H.5: Distribution of the final matching score for annotated pairs in SPICED

Mean Squared Error The mean squared error between two lists of numbers of length n is calculated as:

$$\text{MSE}(Y, \hat{Y}) = \frac{1}{n}\sum_{i}(Y_i - \hat{Y}_i)^2$$

Mean Average Precision The mean average precision in ranking takes the average Precision@k (P@k) for every relevant sample in a ranked list. First, P@k is calculated as follows:

$$\text{P@k}(\hat{Y}) = \frac{1}{k}\sum_{i}^{k} \mathbb{1}(\hat{Y}_i = 1)$$

where $\mathbb{1}$ is the indicator function. The average precision is then taken over all relevant items in the list, where there are r relevant items:

$$\text{AP}(\hat{Y}) = \frac{1}{r}\sum_{k} \text{P@k}(\hat{Y}[:k]) \text{ where } \hat{Y}_k = 1$$

The mean average precision for a set of n ranked lists D is then the mean of the average precision of each of these lists:

$$\text{MAP} = \frac{1}{n}\sum_{j} \text{AP}(D_n)$$

Mean Reciprocal Rank The mean reciprocal rank (MRR) calculates the mean rank for each relevant item in a list i.e. its position in that list. It is calculated as follows for D lists or relevant

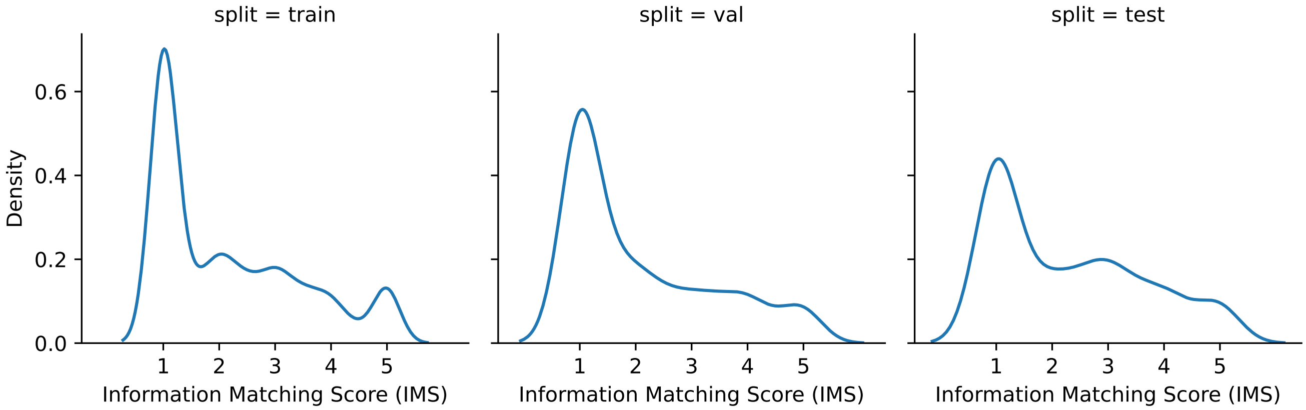


Figure H.6: Distribution of the final matching score for each split set in SPICED

items in $\hat{Y}$ ranked lists:

$$\text{MRR}(D) = \frac{1}{|D|}\sum_j \frac{1}{|D_j|}\sum_i \frac{1}{\text{rank}_i(\hat{Y}_j)}$$

where $\text{rank}_i(\hat{Y}_j)$ is the rank of item i in list $\hat{Y}_j$.

H.7 Full Model Details

All baseline experiments were run on a shared cluster. Requested jobs consisted of 16GB of RAM and 4 Intel Xeon Silver 4110 CPUs. We used a single NVIDIA Titan RTX GPU for experiments. Training takes approximately 3 minutes for all MLM-based models and 2 minutes for SBERT models.

RoBERTa RoBERTa is a large pretrained transformer language model, trained using the masked language modeling (MLM) objective on a large corpus of English text. We use the base model of RoBERTa for our experiments. Huggingface model name: roberta-base – 124,647,170 parameters

MiniLM We use a popular pretrained SBERT model based on MiniLM [245], which is trained by distilling multiple language models into one compressed model. SBERT uses siamese BERT encoders to obtain sentence embeddings for pairs of sentences and is trained to decrease the distance between these two embeddings. The pretraining for the sentence similarity task consists of a wide range of datasets covering multiple domains and $>$ 1 billion sentence pairs, including science [49, 149]. As much of the data is collected automatically, it uses a contrastive learning objective where known relevant pairs are treated as positive values and other samples in a batch are treated as negative values. The model is then trained to minimize the cross-entropy between the dot-product of embeddings and the label acquired from positive/negative samples.

Huggingface model name (sentence transformers): all-MiniLM-L6-v2 – 22,713,216 parameters

MPNet This is the same setup as in MiniLM but with using MPNet as the base network [220]. MPNet is trained using a permuted language modeling (PLM) objective with position information as input to achieve the best of both worlds between MLM and PLM. The base network is used in the SBERT setup where it is further fine-tuned on the same dataset and same task as with MiniLM

Huggingface model name (sentence transformers): all-mpnet-base-v2 – 109,486,464 parameters

Paraphrase Detection This is a paraphrase detection model based on RoBERTa used in [173]. The model is trained on the adversarial paraphrase dataset introduced in that paper.

Huggingface model name (sentence transformers): coderpotter/adversarial-paraphrasing-detector – 124,647,170 parameters

NLI This is a RoBERTa model trained on a wide array of NLI datasets, including SNLI [37], MNLI [250], FEVER (a fact-checking dataset) [230] and ANLI [170].

Huggingface model name (sentence transformers):
ynie/roberta-large-snli_mnli_fever_anli_R1_R2_R3-nli – 124,647,170 parameters

SciBERT SciBERT is the original BERT model trained using MLM on a large set of scientific papers from Semantic Scholar [149].

Huggingface model name (sentence transformers): allenai/scibert_scivocab_uncased – 109,920,514 parameters

CiteBERT CiteBERT is SciBERT further fine-tuned on the CiteWorth dataset for the task of citation detection, which predicts if a given sentence requires a citation or not [256].

Huggingface model name (sentence transformers): copenlu/citebert – 109,920,514 parameters

We use sane defaults when fine-tuning each of our models. In this, for the MLM based models we use [lr: 2e-5, n_epochs: 3, warmup_steps: 200, weight_decay: 0.01, batch_size: 8]. For SBERT models we use the same setting except we train for 5 epochs.

H.8 Exaggeration Detection

The problem of scientific exaggeration detection was studied in [257]. The basic task is: given a pair of scientific findings (e.g. a reference finding from a paper and its counterpart in a news

	All		News		Twitter	
Method	MSE	ρ	MSE	ρ	MSE	ρ
Paraphrase	$3.170_{0.000}$	$23.58_{0.00}$	$3.310_{0.000}$	$19.24_{0.00}$	$2.824_{0.000}$	$35.41_{0.00}$
NLI	$2.921_{0.000}$	$35.71_{0.00}$	$2.786_{0.000}$	$35.78_{0.00}$	$3.255_{0.000}$	$34.76_{0.00}$
MiniLM	$0.628_{0.000}$	$73.98_{0.00}$	$0.646_{0.000}$	$76.27_{0.00}$	$0.583_{0.000}$	$64.61_{0.00}$
MPNet	$0.718_{0.000}$	$72.59_{0.00}$	$0.713_{0.000}$	$74.76_{0.00}$	$0.730_{0.000}$	$62.06_{0.00}$
SciBERT	$0.579_{0.011}$	$73.24_{0.73}$	$0.596_{0.018}$	$74.66_{0.75}$	$0.538_{0.021}$	$66.29_{0.67}$
CiteBERT	$0.581_{0.027}$	$73.37_{0.78}$	$0.592_{0.034}$	$74.81_{0.91}$	$0.552_{0.030}$	$66.13_{1.43}$
RoBERTa	$0.587_{0.017}$	$74.44_{0.81}$	$0.602_{0.033}$	$75.82_{0.71}$	$0.550_{0.067}$	$\mathbf{68.66_{1.29}}$
MiniLM-FT	$0.492_{0.001}$	$75.84_{0.03}$	$\mathbf{0.465_{0.001}}$	$78.66_{0.05}$	$0.559_{0.002}$	$63.80_{0.05}$
MPNet-FT	$\mathbf{0.489_{0.003}}$	$\mathbf{76.48_{0.07}}$	$0.474_{0.003}$	$\mathbf{78.71_{0.17}}$	$\mathbf{0.526_{0.008}}$	$66.45_{0.37}$

Table H.4: OVERALL –MSE and Pearson correlation (ρ) on predicting the similarity of the scientific findings for different models. Results are averaged over 5 random seeds; standard deviation is given in the subscript.

article), determine if one finding is exaggerating the other finding. More formally, the task focuses on differences in the causal claim strength of the two findings, where the claim strength can take on one of four values:

- 0: No statement of relationship
- 1: Correlational statement (e.g. “X is associated with Y”)
- Conditional causal statement (e.g. “X might cause Y under circumstance Z”)
- Causal statement (e.g. “X causes Y”)

[257] curate data and build models for performing the exaggeration detection task in two different settings: as predicting the individual claim strengths and comparing, and as an inference task where a model is fed both findings and asked to predict if the reference finding is being exaggerated, downplayed, or faithfully represented by its counterpart. We use the best-performing model from their paper, which is a multi-task few-shot learning model based on pattern exploiting training (PET) called MT-PET. In particular, we use the model for strength classification which has seen 4,500 individual findings labeled for claim strength and 200 pairs labeled for exaggeration.

H.9 Scientific Text Parser

We fine-tuned a RoBERTa model over 200K self-labeled abstracts from PubMed. The model is trained to predict five labels including: BACKGROUND, CONCLUSIONS, METHODS, OBJECTIVE and RESULTS. We did a 8:1:1 split for the data and fine-tune the RoBERTa model for 1 epoch. 0.92 F1 is attained on the test set.

H.10 Extended Benchmarking

Tables with extended benchmarking results can be found in Table H.4 to Table H.8.

	All		News		Twitter	
Method	MSE	ρ	MSE	ρ	MSE	ρ
Paraphrase	$2.773_{0.000}$	$27.16_{0.00}$	$2.846_{0.000}$	$30.22_{0.00}$	$2.577_{0.000}$	$28.18_{0.00}$
NLI	$2.529_{0.000}$	$40.23_{0.00}$	$2.225_{0.000}$	$47.55_{0.00}$	$3.339_{0.000}$	$6.23_{0.00}$
MiniLM	$0.618_{0.000}$	$76.45_{0.00}$	$0.658_{0.000}$	$80.31_{0.00}$	$\mathbf{0.509_{0.000}}$	$\mathbf{63.78_{0.00}}$
MPNet	$0.804_{0.000}$	$73.14_{0.00}$	$0.815_{0.000}$	$76.91_{0.00}$	$0.777_{0.000}$	$56.11_{0.00}$
SciBERT	$0.554_{0.020}$	$71.67_{0.94}$	$0.507_{0.026}$	$76.69_{0.73}$	$0.681_{0.058}$	$43.56_{5.32}$
CiteBERT	$0.542_{0.031}$	$72.55_{0.92}$	$0.496_{0.034}$	$77.31_{1.11}$	$0.663_{0.029}$	$46.01_{2.61}$
RoBERTa	$0.511_{0.036}$	$75.40_{1.19}$	$0.475_{0.035}$	$79.33_{0.78}$	$0.608_{0.056}$	$53.72_{4.56}$
MiniLM-FT	$\mathbf{0.377_{0.002}}$	$\mathbf{79.46_{0.15}}$	$\mathbf{0.327_{0.003}}$	$\mathbf{84.08_{0.14}}$	$0.512_{0.001}$	$60.00_{0.23}$
MPNet-FT	$0.412_{0.005}$	$77.98_{0.23}$	$0.361_{0.004}$	$82.30_{0.22}$	$0.548_{0.013}$	$57.79_{0.72}$

Table H.5: BIOLOGY – MSE and Pearson correlation (ρ) on predicting the similarity of the scientific findings for different models. Results are averaged over 5 random seeds; standard deviation is given in the subscript.

	All		News		Twitter	
Method	MSE	ρ	MSE	ρ	MSE	ρ
Paraphrase	$3.282_{0.000}$	$15.95_{0.00}$	$3.525_{0.000}$	$31.32_{0.00}$	$2.629_{0.000}$	$29.56_{0.00}$
NLI	$2.820_{0.000}$	$37.03_{0.00}$	$2.841_{0.000}$	$34.60_{0.00}$	$2.763_{0.000}$	$49.39_{0.00}$
MiniLM	$0.706_{0.000}$	$76.46_{0.00}$	$0.739_{0.000}$	$78.34_{0.00}$	$0.615_{0.000}$	$62.92_{0.00}$
MPNet	$0.738_{0.000}$	$79.41_{0.00}$	$0.726_{0.000}$	$81.42_{0.00}$	$0.771_{0.000}$	$64.96_{0.00}$
SciBERT	$0.429_{0.039}$	$81.44_{1.44}$	$0.440_{0.027}$	$83.37_{1.31}$	$0.400_{0.085}$	$\mathbf{70.35_{2.90}}$
CiteBERT	$0.431_{0.044}$	$81.80_{1.19}$	$0.433_{0.042}$	$83.92_{1.32}$	$0.425_{0.067}$	$69.49_{1.21}$
RoBERTa	$0.437_{0.040}$	$82.20_{0.60}$	$0.425_{0.046}$	$84.77_{1.12}$	$0.470_{0.185}$	$69.73_{5.02}$
MiniLM-FT	$0.436_{0.004}$	$79.31_{0.15}$	$0.445_{0.003}$	$81.80_{0.11}$	$0.412_{0.007}$	$64.08_{0.47}$
MPNet-FT	$\mathbf{0.371_{0.005}}$	$\mathbf{82.58_{0.17}}$	$\mathbf{0.369_{0.005}}$	$\mathbf{85.20_{0.22}}$	$\mathbf{0.377_{0.008}}$	$65.03_{0.38}$

Table H.6: MEDICINE – MSE and Pearson correlation (ρ) on predicting the similarity of the scientific findings for different models. Results are averaged over 5 random seeds; standard deviation is given in the subscript.

	All		News		Twitter	
Method	MSE	ρ	MSE	ρ	MSE	ρ
Paraphrase	$3.208_{0.000}$	$32.23_{0.00}$	$3.568_{0.000}$	$27.56_{0.00}$	$2.618_{0.000}$	$46.52_{0.00}$
NLI	$3.066_{0.000}$	$39.57_{0.00}$	$3.125_{0.000}$	$27.39_{0.00}$	$2.970_{0.000}$	$50.61_{0.00}$
MiniLM	$0.539_{0.000}$	$75.16_{0.00}$	$0.525_{0.000}$	$77.98_{0.00}$	$0.561_{0.000}$	$66.81_{0.00}$
MPNet	$0.634_{0.000}$	$72.22_{0.00}$	$0.650_{0.000}$	$72.26_{0.00}$	$0.608_{0.000}$	$69.44_{0.00}$
SciBERT	$0.531_{0.022}$	$74.57_{1.36}$	$0.571_{0.020}$	$74.68_{1.56}$	$\mathbf{0.467_{0.030}}$	$\mathbf{74.14_{1.41}}$
CiteBERT	$0.555_{0.015}$	$73.23_{0.39}$	$0.585_{0.036}$	$73.68_{0.52}$	$0.505_{0.031}$	$72.50_{1.29}$
RoBERTa	$0.655_{0.040}$	$71.28_{1.24}$	$0.720_{0.085}$	$71.38_{1.88}$	$0.550_{0.057}$	$71.35_{1.58}$
MiniLM-FT	$\mathbf{0.500_{0.004}}$	$\mathbf{75.52_{0.11}}$	$\mathbf{0.467_{0.005}}$	$\mathbf{78.48_{0.12}}$	$0.555_{0.004}$	$66.50_{0.16}$
MPNet-FT	$0.520_{0.009}$	$75.21_{0.25}$	$0.550_{0.006}$	$75.48_{0.18}$	$0.471_{0.014}$	$72.25_{0.67}$

Table H.7: PSYCHOLOGY – MSE and Pearson correlation (ρ) on predicting the similarity of the scientific findings for different models. Results are averaged over 5 random seeds; standard deviation is given in the subscript.

	All		News		Twitter	
Method	MSE	ρ	MSE	ρ	MSE	ρ
Paraphrase	$3.373_{0.000}$	$24.35_{0.00}$	$3.346_{0.000}$	$26.48_{0.00}$	$3.463_{0.000}$	$37.48_{0.00}$
NLI	$3.177_{0.000}$	$29.97_{0.00}$	$2.945_{0.000}$	$36.51_{0.00}$	$3.926_{0.000}$	$-8.74_{0.00}$
MiniLM	$0.656_{0.000}$	$71.40_{0.00}$	$0.656_{0.000}$	$73.09_{0.00}$	$0.656_{0.000}$	$66.64_{0.00}$
MPNet	$0.705_{0.000}$	$70.03_{0.00}$	$0.670_{0.000}$	$72.43_{0.00}$	$0.815_{0.000}$	$60.18_{0.00}$
SciBERT	$0.738_{0.020}$	$67.66_{0.71}$	$0.777_{0.031}$	$67.46_{1.03}$	$0.609_{0.029}$	$69.97_{1.76}$
CiteBERT	$0.733_{0.045}$	$68.05_{1.38}$	$0.770_{0.051}$	$67.83_{1.34}$	$0.612_{0.040}$	$69.79_{2.33}$
RoBERTa	$0.690_{0.021}$	$71.53_{1.15}$	$0.731_{0.031}$	$71.24_{0.80}$	$\mathbf{0.560_{0.075}}$	$\mathbf{75.49_{3.09}}$
MiniLM-FT	$0.611_{0.003}$	$72.32_{0.05}$	$0.577_{0.001}$	$74.13_{0.06}$	$0.721_{0.008}$	$66.44_{0.25}$
MPNet-FT	$\mathbf{0.603_{0.004}}$	$\mathbf{73.00_{0.20}}$	$\mathbf{0.575_{0.006}}$	$\mathbf{74.46_{0.36}}$	$0.692_{0.011}$	$67.18_{0.59}$

Table H.8: COMPUTER SCIENCE – MSE and Pearson correlation (ρ) on predicting the similarity of the scientific findings for different models. Results are averaged over 5 random seeds; standard deviation is given in the subscript.

H.11 Error Examples

Examples of errors which our best models made on ⟨tweet, paper⟩ pairs can be found in Table H.9 and Table H.10.

H.12 Regression details

Table H.12 shows the regression table for RQ1. Table H.13 shows the regression table for RQ2.

Tweet	Paper Finding	Prediction	Ground Truth
Mixed reality variations improve learning, over screen-only options. CMU researchers.	The overall improvement from pre to post was 11.3 % in the mixed-reality conditions and 2.4 % in the virtual conditions.	2.92	5
Metarrestin, an inhibitor of tumor metastasis, discovered thru team science @ KU, @username, @username, and @username and more. Congrats to first author Kevin Frankowski and special thanks to Udo Rudloff, Juan Maruguan, and Sui Huang.	Evaluation of apoptotic index showed less than 1% of cells undergoing apoptosis in response to metarrestin treatment.	2.15	4.2
Today in @username a graphene transfer approach using paraffin as a support layer to obtain wrinkle-reduced, clean, large-area graphene retaining high mobility	Similar to previous reports, our PMMA-transferred CVD monolayer graphene on Si/SiO 2 substrate experienced compressive strain and p-doping 30 .	2.64	1
When the Going Gets Tough: The "Why of Goal Striving Matters. An excellent article by @username + colleagues.	Practitioners who aim to facilitate effective goal setting in sport, business, and educational settings would benefit from guidelines for developing autonomous motivation.	2.00	3.6
Thosewho were sociosexually unrestricted reported lower stress and greater overall emotional health after casual sex.	Simple slope analyses indicated that high-SOI participants who had casual sex over the academic year had higher self-esteem (B $\frac{1}{4}$ 0.14, SE $\frac{1}{4}$ 0.06, p $\frac{1}{4}$ .025) and marginally lower depression (B $\frac{1}{4}$ A0.12, SE $\frac{1}{4}$ 0.07, p $\frac{1}{4}$ .091) and anxiety (B $\frac{1}{4}$ A0.11, SE $\frac{1}{4}$ 0.06, p $\frac{1}{4}$ .086) than high-SOI participants who did not have casual sex (Figure 3) .	2.84	4.4

Table H.9: Top-5 biggest errors made by RoBERTa on <tweet, paper> pairs in terms of absolute error.

Tweet	Paper Finding	Prediction	Ground Truth
Mixed reality variations improve learning, over screen-only options. CMU researchers.	The overall improvement from pre to post was 11.3 % in the mixed-reality conditions and 2.4 % in the virtual conditions.	2.45	5
'Physical observation + interactive feedback improved childrens learning by 5x' via Nesra Yannier @username	These results show that mixed-reality led to more learning than screen only, for both the mousecontrol and physical-control conditions (Figure 10).	2.19	4
Today in @username a graphene transfer approach using paraffin as a support layer to obtain wrinkle-reduced, clean, large-area graphene retaining high mobility	Similar to previous reports, our PMMA-transferred CVD monolayer graphene on Si/SiO 2 substrate experienced compressive strain and p-doping 30 .	2.80	1
Metarrestin, an inhibitor of tumor metastasis, discovered thru team science @ KU, @username, @username, and @username and more. Congrats to first author Kevin Frankowski and special thanks to Udo Rudloff, Juan Maruguan, and Sui Huang.	Evaluation of apoptotic index showed less than 1% of cells undergoing apoptosis in response to metarrestin treatment.	2.61	4.2
Super happy to present our latest paper on global food webs: Years of work on predator-prey body-mass ratios and the first use of the GATEWAy data base.	Predators typically exert the strongest feeding pressure on prey that are 1-2 orders of magnitude smalle, while weaker interaction strengths are realized with prey that are smaller or larger than this size.	1.92	3.4

Table H.10: Top-5 biggest errors made by MPNet-FT on <tweet, paper> pairs in terms of absolute error.

Paper Finding	News Finding	Prediction
Increase in the body size of dicynodonts across the Late Triassic may have been driven by selection pressure to reach a size refuge from large predators (24) .	Researchers believe selection pressures–potentially to protect themselves from larger predators–may have been the driver behind their giant size, but more research will be needed to understand Lisowicia and its place in the evolutionary tree.	3.0008
The best option among the three is the EPS container with the lowest impacts across the 12 categories.	The study found that the styrofoam container was the best option among the disposable containers across all the impacts considered, including the carbon footprint.	3.1120
As media coverage started to increase, water demand decreased and the models with media correctly captured the downward trend, but the models without media forecasted increasing demand.	Strikingly, the models also found that for every 100-article increase over a two-month period, there was an 11 percent to 18 percent decrease in demand for water.	3.1537
For example, of the 63 negative precipitation years during 1896-2014, 15 of the 32 warm-dry years (47%) produced 1-SD drought, compared with only 5 of the 31 cool-dry years (16%)	Their analysis revealed that the years that were both warm and dry were about twice as likely to produce a severe drought as years that were cool and dry.	3.2569
Our study shows that low-dose BPA and BPS exposure has physiological effects.	Although the levels were low, the scientists soon saw that both BPA and BPS caused changes in the brain development of the zebra fish embryos.	3.3331
Use of multiple prescription medications with these potential effects was associated with greater likelihood of concurrent depression.	About 15 percent of participants who simultaneously used three or more of these drugs were depressed.	3.3692
We also found that renewal submission rate was the factor most predictive of sustained funding for either gender, and that gender differences in survival disappear when genders were matched on renewal submission rate and first year of funding.	On average, women submitted eligible grants for renewal 42% of the time and won funding 36% of the time, compared with 45% and 39%, respectively, for men.	3.4132
Among those completing the 12-month survey, 60 nonsmokers (55.6%) and 29 smokers (26.6%) were reemployed at 1 year.	After 12 months, the re-employment rate of smokers was 24 percent lower than that of nonsmokers.	3.5151
This suggests behaviour consistent with moral licensing: participants who refrained from cheating at higher stakes seem to have subsequently licensed themselves to donate less to charity, thereby "balancing" their moral behaviour over time.	However those who cheated the least when tempted with high stakes were more likely to license themselves not to behave so charitably in another task.	3.5481
Lack of Panx1 increases adipocyte hypertrophy and reduces adipocyte numbers in subcutaneous fat in vivo.	With both a normal diet, and a a high-fat diet, a lack of Panx1 increases cell size.	3.5618

Table H.11: Borderline IMS Model prediction samples. We note that 3 appears to be a good threshold for matching, as pairs with an IMS over 3 tend to discuss the same scientific findings.

Model:	MixedLM	Dependent Variable:	paper_sentence_score
No. Observations:	1111150	Method:	REML
No. Groups:	6705	Scale:	0.1084
Min. group size:	31	Log-Likelihood:	-349944.7797
Max. group size:	67063	Converged:	Yes
Mean group size:	165.7		

	β Coef.	Std.Err.	z	P> |z|	[0.025	0.975]
Intercept	3.299	0.007	489.729	0.000	3.286	3.312
Outlet Type: Press Release	0.037	0.001	31.187	0.000	0.035	0.039
Outlet Type: Science & Technology	0.034	0.001	30.581	0.000	0.032	0.036
Field: Biology	-0.018	0.020	-0.904	0.366	-0.056	0.021
Field: Psychology	0.040	0.018	2.168	0.030	0.004	0.076
Field: Medicine	0.206	0.017	11.813	0.000	0.171	0.240
Field: Computer_science	0.050	0.024	2.132	0.033	0.004	0.096
Group Var	0.009	0.001				

Table H.12: Regression table for RQ1

Model:	MixedLM	Dependent Variable:	paper_sentence_score
No. Observations:	182735	Method:	REML
No. Groups:	1360	Scale:	0.1525
Min. group size:	31	Log-Likelihood:	-89654.8514
Max. group size:	89523	Converged:	Yes
Mean group size:	134.4		

	β Coef.	Std.Err.	z	P> |z|	[0.025	0.975]
Intercept	3.777	0.013	292.571	0.000	3.752	3.803
Is Verified User?	-0.047	0.004	-11.044	0.000	-0.056	-0.039
Is Organizational Account?	0.042	0.002	-19.026	0.000	-0.046	-0.037
User Metric: log(Followers)	-0.003	0.001	-5.059	0.000	-0.004	-0.002
User Metric: log(Following)	0.000	0.001	0.369	0.712	-0.001	0.002
User Metric: Account Age (in years)	0.004	0.000	10.824	0.000	0.003	0.005
Field: Biology	-0.025	0.030	-0.850	0.395	-0.083	0.033
Field: Psychology	0.308	0.028	11.052	0.000	0.254	0.363
Field: Medicine	0.206	0.026	7.826	0.000	0.155	0.258
Field: Computer_science	-0.352	0.035	-10.158	0.000	-0.420	-0.284
Group Var	0.059	0.006				

Table H.13: Regression table for RQ2