# An Entity-based Claim Extraction Pipeline for Real-world Biomedical Fact-checking

**Amelie Wührl, Lara Grimminger,** and **Roman Klinger**
Institut für Maschinelle Sprachverarbeitung, University of Stuttgart, Germany
{amelie.wuehrl, lara.grimminger, roman.klinger}@ims.uni-stuttgart.de

## Abstract

Existing fact-checking models for biomedical claims are typically trained on synthetic or well-worded data and hardly transfer to social media content. This mismatch can be mitigated by adapting the social media input to mimic the focused nature of common training claims. To do so, Wührl and Klinger (2022a) propose to extract concise claims based on medical entities in the text. However, their study has two limitations: First, it relies on gold-annotated entities. Therefore, its feasibility for a real-world application cannot be assessed since this requires detecting relevant entities automatically. Second, they represent claim entities with the original tokens. This constitutes a terminology mismatch which potentially limits the fact-checking performance. To understand both challenges, we propose a claim extraction pipeline for medical tweets that incorporates named entity recognition and terminology normalization via entity linking. We show that automatic NER does lead to a performance drop in comparison to using gold annotations but the fact-checking performance still improves considerably over inputting the unchanged tweets. Normalizing entities to their canonical forms does, however, not improve the performance.

## 1 Introduction

Fact-checking models trained on synthetic, well-worded and atomic claims struggle to transfer to colloquial content (Kim et al., 2021). There are multiple ways to address this problem: We can build custom datasets and models that verify medical content shared online (Saakyan et al., 2021; Mohr et al., 2022; Sarrouti et al., 2021) and tackle related tasks (Sundriyal et al., 2022; Dougrez-Lewis et al., 2022). Alternatively, we can adapt the input before addressing other fact-checking tasks. Bhatnagar et al. (2022) create claim summaries and find that this improves the detection of previously fact-checked claims. Similarly, Wührl and Klinger (2022a) extract concise claims from user-generated text in an effort to mimic the focused, well-structured nature of the claims the fact-checking models were originally trained on. They find that this improves the accuracy of pre-trained evidence-based fact-checking models in the biomedical domain.

| | Claim | Evidence |
|---|---|---|
| orig | medicines causes blood clots | drospirenone may significantly increase chances of developing venous thromboembolic events |
| norm | pharmaceutical preparations causes thrombus | |

Table 1: Example claim represented with original and normalized entities together with evidence.

However, the study by Wührl and Klinger (2022a) is limited in two ways: (1) Their claim extraction method relies on gold-annotated, claim-related entities. For a realistic evaluation, such an oracle needs to be replaced by an entity recognizer. Only then it is possible measure the impact of potential error propagation which may ultimately render the method unfeasible. (2) The claim entities are represented by the original token sequence. This is problematic as medical mentions on Twitter potentially contain imprecise, abbreviated, or colloquial terminology. This is in contrast to the terminology in the original model input as well as the documents that we provide as evidence (cf. Table 1). We hypothesize that for a successful fact-check we need to close this gap by normalizing medical terminology in the input. Previous work suggested leveraging entity linking for evidence retrieval (Nooralahzadeh and Øvrelid, 2018; Taniguchi et al., 2018; Hanselowski et al., 2018) leading us to believe that it could also be beneficial for aligning claim and evidence.

We address both limitations and evaluate a real-world, fully-automatic claim extraction pipeline for medical tweets which incorporates an entity rec-

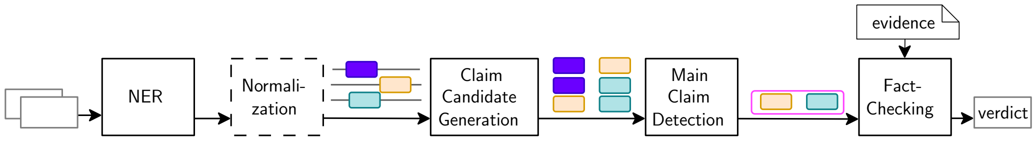


Figure 1: Overview of the claim extraction pipeline. Input documents go through entity recognition (NER), normalization, claim candidate generation, main claim detection and fact-checking. Colored boxes represent the entities which we use to extract claim candidates. Note that we evaluate the normalization module separately from the evaluation of the rest of the pipeline (see §3).

ognizer. It only relies on the original text as input that contains the claim. We further evaluate the impact of an entity linker for normalizing entity mentions to canonical forms based on the Unified Medical Language System (UMLS, Bodenreider, 2004). Our pipeline improves the fact-checking performance over tasking models to check unchanged tweets. Normalizing entities to overcome the terminology mismatch does not improve fact-checking, potentially due to limitations of biomedical entity linking for social media.

## 2 Methods

Figure 1 visualizes our pipeline. It takes text as input and performs *named entity recognition* and optionally term *normalization* via entity linking. Each unique entity pair forms the building blocks for a potential claim (*claim candidate generation*). The *main claim detection* identifies the core claim among the candidates that presumably represents the most important aspect of the text. The resulting claim is the input to the fact-checker. In our setting, we assume this to be a frozen pre-trained fact-checking model. We describe the modules in the following and the fact-checker in Section 3.2.

**NER.** We use the SpaCy environment[1] to train a custom NER model that detects medical entities. This framework relies on a transition-based parser (Lample et al., 2016) to predict entities in the input. In a preliminary study, we found that relying on an off-the-shelf model for biomedical NER, i.e., ScispaCy (Neumann et al., 2019), does not transfer to medical texts from social media. Refer to Appendix B.1 for a comparison of the two models.

**Claim candidate generation.** Wührl and Klinger (2022a) propose two extraction methods, i.e., $\text{condense}_{\text{seq}}$ and $\text{condense}_{\text{triple}}$. The first represents the claim as the token sequence from the first entity to the last entity, while the second relies on gold-annotated causal relations which they use to build the claims. We use the sequence method $\text{condense}_{\text{seq}}$ in our pipeline because both methods show on par performances (difference in 1pp $F_1$) and, in contrast to $\text{condense}_{\text{triple}}$, it does not require relation classification.

Following the $\text{condense}_{\text{seq}}$ method, we therefore extract the sequence from the character onset of the first entity to the character offset of the second entity for all pairs of entities found by the NER module.

**Entity linking.** To normalize entities, we use the *EntityLinking* component in ScispaCy (Neumann et al., 2019). This model compares an entity mention to concepts in an ontology and creates a ranked list of candidates, based on an approximate nearest neighbor search. For text normalization, we retrieve the canonical name of the top concept. For entities which could not be linked, we use the original mention instead. As the knowledge base, we use UMLS (Bodenreider, 2004).

**Main claim detection.** For tweets with more than two predicted entities, claim generation produces multiple claim candidates. To identify the claim to be passed to the fact-checking module, we train a text classifier to detect the main claim for a given input. We build on RoBERTArg[2], a RoBERTA-based text classification model trained to label input texts as ARGUMENT or NON-ARGUMENT. We fine-tune this model to classify texts as CLAIM vs. NON-CLAIM and to fit the social media health domain. At inference time, the claim candidate with the highest probability for the claim class constitutes the main claim. We refer to this as *ner+core-claim*.

## 3 Experiments

### 3.1 Data

**CoVERT.** We use the CoVERT dataset (Mohr et al., 2022) to test our pipeline. It consists of

[1] https://spacy.io/api/architectures#TransitionBasedParser

[2] https://huggingface.co/chkla/roberta-argument

| | Input Claim | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Gold entities | | | | | | | Fully automatic (Ours) | | | | | | | | | | | |
| | condense$_{seq}$ | | | | full tweets | | | ner+rand-ent-seq | | | | ner+core-claim | | | | | | | |
| model | P | R | $F_1$ | $\Delta_{full}$ | P | R | $F_1$ | P | R | $F_1$ | $\Delta_{full}$ | P | R | $F_1$ | $\Delta_{full}$ | | | | |
| fever | 83.3 | 1.9 | 3.7 | +3.7 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | +0 | 100 | 0.4 | 0.8 | +0.8 | | | | |
| fever_sci | 87.2 | 15.5 | 26.4 | +18.4 | 91.7 | 4.2 | 8.0 | 92.3 | 4.7 | 9.0 | +1.0 | 82.4 | 5.6 | 10.4 | +2.4 | | | | |
| scifact | 90.9 | 7.6 | 14.0 | +13.2 | 100 | 0.4 | 0.8 | 100 | 2.4 | 4.6 | +3.8 | 100 | 2.4 | 4.7 | +3.9 | | | | |
| covidfact | 55.6 | 28.4 | 37.6 | +29.7 | 30.8 | 4.5 | 7.9 | 53.3 | 9.4 | 16.1 | +8.2 | 58.1 | 14.3 | 23.0 | +15.1 | | | | |
| healthver | 85.9 | 48.5 | 62.0 | +16.8 | 82.8 | 31.1 | 45.2 | 75.6 | 23.2 | 35.5 | −9.7 | 77.4 | 28.7 | 41.9 | −3.3 | | | | |
| average | 80.6 | 20.4 | 28.7 | +16.3 | 61.1 | 8.0 | 12.4 | 64.2 | 7.9 | 13.0 | +0.6 | 83.6 | 10.1 | 16.2 | +3.8 | | | | |

Table 2: Performance (precision, recall and $F_1$) of MultiVerS-based models (*fever, fever_sci, scifact, covidfact, healthver*) on CoVERT data. Model inputs are the full tweets, the entity-based sequence claims (condense$_{seq}$ (Wührl and Klinger, 2022a)), and claims from the fully automatic pipeline, *ner+rand-ent-seq* and *ner+core-claim*. $\Delta_{full}$ : difference in $F_1$ between the full tweet and performance for the respective input claim. We report the average across all models in the last row.

medical tweets labeled with fact-checking verdicts (SUPPORTS, REFUTES, NOT ENOUGH INFORMATION) and associated evidence texts. We follow the same filtering and preprocessing as Wührl and Klinger (2022a) which leaves us with 264 tweets. For 13 tweets, the NER model predicts only one or no entities. In these cases, we cannot generate claim candidates thus we can only consider 251 claims.

**BEAR.** We require an independent dataset to train the NER component. We find the BEAR dataset (Wührl and Klinger, 2022b) to be closest in domain and text type to the target data from CoVERT. BEAR provides 2100 tweets with a total of 6324 annotated medical entities from 14 entity classes. We use 80% of the data for training and 20% for testing the model.

**Causal Claims.** To build a classifier that identifies the core claims, we use the CAUSAL CLAIMS data from SemEval-2023 Task 8, Subtask 1.[3] It consists of medical Reddit posts and provides span-level annotations for *Claim*, *Experience*, *Experience based claim* and *Question*. Our goal is to differentiate claims from non-claims. Consequently, we extract all spans labeled as *Claim* and *Experience based claim* as positive instances for the claim class and use the remaining text spans as negative examples. This leads to 1704 claim and 6870 non-claim spans. We use a train/test split of 90/10%.

### 3.2 Evaluation

The fact-checking module serves as a by-proxy evaluation for the claim representations. Provided with a claim–evidence pair, the system predicts a fact-checking verdict that indicates if the evidence SUPPORTS or REFUTES the claim. We assume that the fact-checker is a frozen model for which we adapt the claim input. To gauge the checkability of a particular input, we compare the performance for predicting the correct verdict when the model is presented with claims of this type. This follows the evaluation in Wührl and Klinger (2022a).

The fact-checking models we employ stem from the MultiVerS architecture (Wadden et al., 2022).[4] This framework is designed for scientific fact-verification and provides five models (*fever, fever_sci, scifact, covidfact, healthver*), differing in training data. We report precision, recall and $F_1$ for predicting the correct fact-checking verdict (SUPPORTS, REFUTES, NOT ENOUGH INFORMATION) for a given claim-evidence pair.

### 3.3 Exp. 1: Impact of NER

In Exp. 1, we aim to understand the impact of automatic NER and main claim detection in the pipeline, instead of relying on gold-labeled entities.

Table 2 reports the results for our fully automatic claim extraction pipeline. Each column reports the performance for a specific type of input claim. *Full tweets* is the performance as reported by Wührl and Klinger (2022a) for the unchanged input tweets. The results denoted with condense$_{seq}$ describe their results with gold annotations, to which we compare. Our main results are in the last column (*ner+core-claim*). To understand the impact of the main claim detection, we compare against a purely random

[3] https://causalclaims.github.io/

[4] https://github.com/dwadden/multivers

selection of the main claim from all candidates in the tweet (*ner+rand-ent-seq*).

The rows correspond to the various fact-checking models. Δ columns report the difference in $F_1$ between the performance of checking the full tweet and the respective claim representation.

*ner+core-claim* shows an average performance of $F_1$ =16.2. The performance varies across the models. The *healthVer* model performs the best (41.9$F_1$). The average is considerably higher than using the full tweets (Δ=3.8 pp $F_1$). This improvement is consistent across all models, except for *healthVer*, presumably because it already shows a high performance for the original texts. To better understand the model behavior, we provide an analysis of its prediction in Appendix B.3. We see a particularly strong impact for the *covidfact* model, with Δ=15.1 pp. Despite this positive result, we see a performance drop when integrating entity recognition instead of building claim extraction on gold entity annotations. This decrease is not surprising since we expect some error propagation from an imperfect entity recognizer. Nevertheless, the results show that entity-based claim extraction also increases the fact-checking performance even under some error propagation throughout the real-world pipeline.

We further see that main claim detection is a required module – the performance for a randomly selected claim (*ner+rand-ent-seq*) is substantially lower. This indicates that using the same evidence and fact-checking model, not all potential claims in a tweet would receive the same verdict.

### 3.4 Exp. 2: Impact of Entity Normalization

In Exp. 2, we investigate if it is beneficial to assimilate the linguistic realizations of medical mentions to the expected input of the fact-checking models. More specifically, we suggest normalizing entity strings in the input. In contrast to Exp. 1, in which we evaluate the overall pipeline, we focus on the aspect of the entities here and therefore do not make use of the core claim detection method or the entity recognizer. Instead we build on top of gold annotations and, consequently, employ $\text{condense}_{\text{triple}}$ described in Section 2.

We use entity linking for term normalization and use ScispaCy's entity linking functionality with *en_core_sci_sm* as the underlying model (Neumann et al., 2019). For each (gold) entity, we use the canonical name of the concept with the highest linking score. Subsequently, we follow the $\text{condense}_{\text{triple}}$ method to represent claims.

| | $\text{condense}_{\text{triple}}$ Claims | | | | | |
|---|---|---|---|---|---|---|
| | surface string | | | normalized ent. | | |
| model | P | R | $F_1$ | P | R | $F_1$ |
| fever | 81.8 | 3.4 | 6.5 | 75.0 | 1.1 | 2.2 |
| fever_sci | 89.8 | 20.1 | 32.8 | 93.9 | 11.7 | 20.9 |
| scifact | 86.4 | 7.2 | 13.3 | 94.4 | 6.4 | 12.1 |
| covidfact | 65.0 | 30.3 | 41.3 | 61.8 | 20.8 | 31.2 |
| healthver | 79.7 | 41.7 | 54.7 | 85.7 | 31.8 | 46.4 |
| average | 80.5 | 20.5 | 29.7 | 82.2 | 14.4 | 22.6 |

Table 3: Performance (precision, recall and $F_1$) of MultiVerS-based fact-checking models (*fever, fever_sci, scifact, covidfact, healthver*) on CoVERT claims built with non-normalized (surface string) vs. normalized entities. We report the average across all models in the last row.

Table 3 reports the results for claims built with non-normalized (*surface string*) vs. normalized entities (*normalized ent.*). The results indicated as $\text{condense}_{\text{triple}}$ *surface string* are analogue to the results in Wührl and Klinger (2022a). We see that normalization does not have the desired effect: The verdict prediction performance drops across all of the fact-checking models (from 29.7 to 22.6 in avg. $F_1$). We assume that this is, to a considerable extend, due to entity linking being a challenging task which leads to a limited performance of the employed linking module. We present an error analysis in Appendix B.4.

## 4 Conclusion & Future Work

We propose a fully automatic claim extraction pipeline that is capable of handling real-world medical content. We show that entity-based claim extraction has a positive effect on the performance of multiple fact-checking models – even after replacing the entity oracle with automatic NER. While we observe a negative impact of error propagation from NER and a performance drop as a result, fact-checking the extracted claims is more successful than checking unchanged tweets. Future research may therefore focus on improving the pipeline components as this clearly has the potential to further strengthen the verdict prediction performance. In particular, we expect an improved entity recognizer to have a considerable impact.

Our work focuses on the biomedical domain and builds upon the assumption by Wührl and Klinger (2022a) that claims in this domain are strongly centered around entities. Claims from other domains may share this property which could make entity-based claim extraction applicable for such claims as well. We leave the evaluation for future work.

We find that normalizing entity mentions does not improve the fact-checking performance. However, our analysis shows that the off-the-shelf linking module might be too unreliable. To fully gauge the potential of normalizing entities, future work needs to ensure correct mappings (creating gold links or building a reliable linker) before evaluating the downstream fact-checking performance.

## Acknowledgments

This research has been conducted as part of the FIBISS project which is funded by the German Research Council (DFG, project number: KL 2869/5-1). We thank the anonymous reviewers for their valuable feedback.

## Limitations

Our work focused on evaluating the impact of putting together a set of components to achieve a real-world system for fact-checking. For answering the research question at hand, the components offered themselves as appropriate choices. This being said, to some degree, the particular selection may limit the expressiveness of the experiments.

By instantiating the pipeline components with the set of models and underlying data that we chose, our findings are limited to this setting. However, the analysis that we provide in Appendix B dissects the pipeline results and allows us to draw more general conclusions about the impact of replacing individual components.

We propose that the main claim detection receives more attention in future research. This may mitigate the issue that this module is potentially the most in-transparent component. Compared to the NER, this task can be modeled in various ways. We rely on the output probabilities to identify the claim candidate the model is most confident about. While this is a straight-forward approach and we show that it works as intended, prediction probabilities – especially for deep models – may not always be a distinctive indicator of model confidence. To overcome this limitation, alternative ways of detecting the main claim should be evaluated.

## Ethical Considerations

A real-world fact-checking pipeline presents itself as a valuable tool. However, we advise against using the pipeline purely automatically that at this point in time. Unless they are used hand-in-hand with a human expert performing or supervising the fact-check, such systems are not reliable enough yet.

Potential issues are the result of the inherent opaqueness of sophisticated automatic analysis pipelines. In the system that we propose, it is important that the impact of each module needs to explain itself to the user. While there is recent work on explainability particularly in the area of fact checking, this work did not yet focus on entity-based approaches. It is important that a user can clearly understand which claim in a statement is checked and which risks potential error propagation might lead to. Therefore, before deploying such systems for fully automatic filtering or labeling of problematic messages in a social media content, there needs to be more research on explainability and transparency of such systems.

## References

Varad Bhatnagar, Diptesh Kanojia, and Kameswari Chebrolu. 2022. Harnessing abstractive summarization for fact-checked claim detection. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2934–2945, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Olivier Bodenreider. 2004. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32:D267–270.

John Dougrez-Lewis, Elena Kochkina, Miguel Arana-Catania, Maria Liakata, and Yulan He. 2022. PHEMEPlus: Enriching social media rumour verification with external evidence. In *Proceedings of the Fifth Fact Extraction and VERification Workshop (FEVER)*, pages 49–58, Dublin, Ireland. Association for Computational Linguistics.

Andreas Hanselowski, Hao Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. 2018. UKP-athene: Multi-sentence textual entailment for claim verification. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 103–108, Brussels, Belgium. Association for Computational Linguistics.

Byeongchang Kim, Hyunwoo Kim, Seokhee Hong, and Gunhee Kim. 2021. How robust are fact checking systems on colloquial claims? In *Proceedings of* *the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1535–1548, Online. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Isabelle Mohr, Amelie Wührl, and Roman Klinger. 2022. CoVERT: A corpus of fact-checked biomedical COVID-19 tweets. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 244–257, Marseille, France. European Language Resources Association.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 319–327, Florence, Italy. Association for Computational Linguistics.

Farhad Nooralahzadeh and Lilja Øvrelid. 2018. SIRIUS-LTG: An entity linking approach to fact extraction and verification. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 119–123, Brussels, Belgium. Association for Computational Linguistics.

Arkadiy Saakyan, Tuhin Chakrabarty, and Smaranda Muresan. 2021. 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 (Volume 1: Long Papers)*, pages 2116–2129, Online. Association for Computational Linguistics.

Mourad Sarrouti, Asma Ben Abacha, Yassine Mrabet, and Dina Demner-Fushman. 2021. Evidence-based fact-checking of health-related claims. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3499–3512, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Megha Sundriyal, Atharva Kulkarni, Vaibhav Pulastya, Md Shad Akhtar, and Tanmoy Chakraborty. 2022. Empowering the fact-checkers! automatic identification of claim spans on twitter. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, page 7701–7715, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Motoki Taniguchi, Tomoki Taniguchi, Takumi Takahashi, Yasuhide Miura, and Tomoko Ohkuma. 2018. Integrating entity linking and evidence ranking for fact extraction and verification. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 124–126, Brussels, Belgium. Association for Computational Linguistics.

David Wadden, Kyle Lo, Lucy Lu Wang, Arman Cohan, Iz Beltagy, and Hannaneh Hajishirzi. 2022. MultiVerS: Improving scientific claim verification with weak supervision and full-document context. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 61–76, Seattle, United States. Association for Computational Linguistics.

Amelie Wührl and Roman Klinger. 2022a. Entity-based claim representation improves fact-checking of medical content in tweets. In *Proceedings of the 9th Workshop on Argument Mining*, pages 187–198, Online and in Gyeongju, Republic of Korea. International Conference on Computational Linguistics.

Amelie Wührl and Roman Klinger. 2022b. Recovering patient journeys: A corpus of biomedical entities and relations on Twitter (BEAR). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4439–4450, Marseille, France. European Language Resources Association.

# A Implementation details

In the following, we provide implementation details for the individual model components described in Section 2.

## A.1 Named Entity Recognition

In a preliminary experiment, we use a pre-trained model for biomedical NER, i.e., the *en_core_sci_sm* model by ScispaCy (Neumann et al., 2019), that was trained on scientific, biomedical and clinical text to identify sequences of biomedical entities. We find that the off-the-shelf model transfers poorly to our target data which stems from social media. We provide the evaluation results for this experiment in Appendix B.2.1. Therefore, we train a custom NER model in spaCy on the BEAR dataset. We create an empty model using spacy.blank() and pass the language ID "en" for English. We provide the train/test splits and configuration file we use to train the model which includes all settings and hyperparameters here: `https://tinyurl.com/bear-ner`

## A.2 Main Claim Detection

We fine-tune RoBERTArg[5] to classify texts as CLAIM VS. NON-CLAIM using the Causal Claim data. We create a train-validation split of 85/15 %. We train for 5 epochs with a batch size of 16, 409

[5] `https://huggingface.co/chkla/roberta-argument`

training steps per epoch, 136 warmup steps and a weight decay of 0.01. We use the same learning rate that was used in fine-tuning the underlying RoBERTArg model, i.e., a learning rate of 2.3102e-06. We evaluate the model every 500 steps using the validation set. After training, we use the model with the best performance on the validation set to make a prediction for each claim candidate.

### A.3 Entity linking

We use the *EntityLinking* component in ScispaCy (Neumann et al., 2019) and *en_core_sci_sm* as the underlying model[6]. For each entity, the model maps the mention to the associated concept within UMLS (Bodenreider, 2004). We include the option to resolve abbreviations and leave the other configuration parameters at their default values.

## B Analysis

We provide an evaluation and analyses of individual pipeline components to better understand the capabilities of the modules.

### B.1 Evaluation Setup

**NER.** Entity recognition consists of two subtasks: (a) identifying the span of an entity and (b) predicting the entity class. Consequently, we evaluate the NER component of our pipeline in two modes. In the *strict* mode, the entity span and the entity class have to be identical to the gold data. In the *relaxed* mode, the entity span has to be identical to the gold data, entity class labels is ignored.

Note that the off-the-shelf ScispaCy (Neumann et al., 2019) model that we compare against only labels the entity span and not the entity class. Therefore, we can only evaluate its performance in the *relaxed* mode.

Further note that we need to map certain entity classes between the CoVERT and the BEAR dataset. To align CoVERT with BEAR, we map *Medical Condition* to *med_C*, *Treatment* to *treat_therapy*, and *OTHER* to other, respectively. The CoVERT dataset further contains the class *Symptom/Side-effect*, which corresponds to the class *med_C* of the BEAR dataset. Therefore, we map the class *Symptom/Side-effect* to the class *med_C*. Entities which have been labeled in BEAR, but not in CoVERT, are ignored for the evaluation.

We report the macro-average of precision, recall and $F_1$ for both modes.

[6]https://allenai.github.io/scispacy/

**Main claim detection.** We evaluate the prediction of the model on the held-out test set from the CAUSAL CLAIM data. We report precision, recall and $F_1$ for both classes (CLAIM VS. NON-CLAIM) the as well as the macro-average.

### B.2 Results

#### B.2.1 NER

We evaluate the performance of the NER component within our pipeline. Table 4 reports the results for the strict and relaxed evaluation mode. First, we evaluate the performance on the unseen test split of the BEAR data – the dataset we use for training the model. To gauge how well it transfers to our target data, we evaluate the performance for the entity predictions in CoVERT. We compare the performance of our custom model to the performance of the pre-trained ScispaCy model.

For the BEAR data, our model reaches an average $F_1$ of 0.41 for the strict evaluation mode. Note that in this mode only exact span and entity type matches count as true positives. If we relax this condition and disregard the entity type, the model achieves an $F_1$-score of 0.51. When moving to a slightly different type of input text, i.e., the CoVERT data, the average $F_1$-scores for the strict and relaxed evaluation modes reach 0.34 and 0.38, respectively.

Compared to our custom model, the performance of the off-the-shelf model from ScispaCy is much lower. For the relaxed mode, we observe a $\Delta$ in $F_1$ of 0.21 and 0.12 for the BEAR and CoVERT data, respectively. This showcases the necessity of a customized model for NER in this setting.

Overall, this evaluation of the entity recognition shows moderate performance. Importantly, the results also indicate that improving this component is likely to improve the overall fact-checking performance.

#### B.2.2 Main claim detection

We evaluate the performance of the claim detection model on the held-out test set. We report the results in Table 5. We can see that the model successfully differentiates claims from non-claims ($F_1$-scores of 0.94 and 0.99, respectively).

### B.3 Analysis of *healthver* prediction

We want to understand why the *healthver* model behaves unexpectedly compared to the other models (refer to Table 2). We saw that providing the

| | | target data | | | | | |
|---|---|---|---|---|---|---|---|
| | | BEAR | | | CoVERT | | |
| model | eval. mode | P | R | $F_1$ | P | R | $F_1$ |
| ScispaCy | strict | - | - | - | - | - | - |
| | relaxed | .2 | .61 | .3 | .16 | .72 | .26 |
| Ours | strict | .46 | .37 | .41 | .26 | .51 | .34 |
| | relaxed | .56 | .46 | .51 | .29 | .57 | .38 |

Table 4: Evaluation of our NER module for the test split of the BEAR dataset and the CoVERT data. We report the macro average precision (P), recall (R) and $F_1$ across all entity classes. We report results for a strict and a relaxed evaluation mode. We compare against the performance of an off-the-shelf ScispaCy (Neumann et al., 2019) model (*en_core_sci_sm*). This model only labels the entity span, not the entity class. Therefore, we only evaluate in the relaxed mode.

| class | P | R | $F_1$ |
|---|---|---|---|
| Non-claim | 0.98 | 0.99 | 0.99 |
| Claim | 0.95 | 0.93 | 0.94 |
| macro av. | 0.97 | 0.96 | 0.96 |

Table 5: Performance (precision (P), recall (R), $F_1$) of the claim detection model for CAUSAL CLAIMS test set.

automatically extracted claim leads to a slight performance decrease compared to inputting the full tweet, while the claims extracted using gold entities were more successfully checked. We hypothesize that for this model, the automatic extraction either removed relevant pieces of the input that it relied on previously for a successful prediction or it may have introduced irrelevant noise. Therefore, we compare the predictions of this model for our *ner+core-claim* inputs to the claims built on gold-labeled entities $\text{condense}_{\text{seq}}$. Note that we compare the predictions which are not necessarily in line with the gold label.

**Label distribution.** Table 6 reports the distribution of predicted labels for both input types. The NEI class increases substantially (115 to 158 predicted instances) while SUPPORT and REFUTE become less frequent. This indicates that the claims become less checkable as NEI means a lack of information to support or refute the claim.

**Label flips.** To better understand which instances cause the model to predict a different verdict, we present the number of label transitions between the predictions for the gold-labeled entity claims and the predictions for our pipeline claims (*ner+core-claim*) in Table 7. From those results we can observe that for a substantial amount of instances (161) the predicted label actually does not shift. For 90 instances, we observe a label shift.

| | # Predicted labels | | |
|---|---|---|---|
| input claim | SUPPORT | REFUTE | NEI |
| gold entities | 110 | 39 | 115 |
| *ner+core-claim* | 69 | 24 | 158 |

Table 6: Labels predicted by the *healthver* model for claims extracted using $\text{condense}_{\text{seq}}$ based on gold entities and our pipeline (*ner+core-claim*). Note that there are 13 claims more in the gold-entity setting compared to *ner+core-claim* inputs. These are cases for which the NER module predicted $\leq 1$ entity.

Most notably, claims that were supported and refuted when inputting the gold-entity claims, get classified as NEI when we input our extracted claims (46 and 18, respectively). In an introspection of this transition type, we observe that in cases, the automatic pipeline failed to detect the main claim, potentially rendering the evidence useless. Refer to Claims 3 and 4 in Table 7 for examples.

Flipped verdicts (SUPPORTS to REFUTES or vice versa) are less frequent. We observe a total of 11 instances. Refer to Claims 1 and 2 in Table 7.

We observe 15 cases in which the label flips to the correct gold label when we input our claim as opposed to the gold-entity-based claim. In the manual introspection, we observe many cases in which the claim from the pipeline slightly extends the context compared to the gold entity claim. Refer to Claims 5 and 6 in Table 7 for two examples.

For cases with consistent labels, we find that many instances either are identical to the claim extracted using gold entities (see Table 7, Example 7a) or only small amounts of context is added (see Table 7, Ex. 8).

This being said, we also observe cases in which the gold-entity and our predicted claim do not overlap and yet, the verdict stays consistent (Ex. 7b). This emphasizes the need to further improve the main claim detection step and leads us to hypothesize that this module may be another reason for the limited performance of this model. It appears that the *healthver* model is particularly sensitive to this component being somewhat unreliable and error propagation in general.

### B.4 Entity Linking

**Number of established mappings.** There are no gold annotated mappings for the medical entities

| id | transition | # inst. | example: gold-ent-claim | example: ner+core-claim | gold |
|---|---|---|---|---|---|
| 1 | S-R | 7 | Oral contraceptives cause more blood clots | blood clots and nobody is doing anything about that!!! Like 1 per 1,000 compared to basically 1 per MILLION with the Covid vaccine | S |
| 2 | R-S | 4 | COVID-19 vaccines can cause side effects | Vaccine reactions are rare. Info about side effects | S |
| 3 | S-NEI | 46 | COVID-19 1) directly causes viral pneumonia | pneumonia 3) can result in intubation | S |
| 4 | R-NEI | 18 | 5G causes covid | vaccines cause infertility & autism | R |
| 5 | NEI-S | 12 | live virus that causes covid-19 | vaccines don't use the live virus that causes covid-19 | S |
| 6 | NEI-R | 3 | masks cause plague | masks cause plague... fauci knows... masks promote bacteria... and not the good kind... sinus | R |
| 7a | S-S | 53 | covid vaccine doesn't cause fertility issues | covid vaccine doesn't cause fertility issues | S |
| 7b | S-S | | all brands of the vaccine can cause problems | death rate of COVID is said to be 10%. It is probable that some vaccines | S |
| 8 | R-R | 14 | Wearing a mask does cause disease | Wearing a mask does cause disease, harm the immune system | R |
| 9 | NEI-NEI | 94 | Auto-Immune disease causes the white blood cells that normally protect your body from invaders to turn around and attack your cells, tissues and organs | Auto-Immune disease causes the white blood cells that normally protect your body | S |

Table 7: Label transitions as predicted by the *healthver* model for claims extracted using condense$_{seq}$ based on gold entities (gold-ent-claim) and our pipeline (*ner+core-claim*). We provide example instances for each type of label transitions along with the gold label for the fact-checking verdict.

in the CoVERT dataset that would allow for a full evaluation. We therefore approximate one aspect of the quality of the entity linking module by analyzing the number of entities that are being linked to any concept in the first place. Out of 719 entity mentions the linking module established mappings for 495 instances (68.8 %). We provide insights from an error analysis in the following section.

**Error analysis.** We aim to understand the type of error patterns introduced by the entity linking module. We analyze predicted links for a randomly drawn sample of 25 entities. We manually categorize the predicted concepts with regard to four properties. Table 8 reports the results as well as examples. *correctly linked* instances are mapped to the appropriate concept within UMLS. *Incorrect but related link* include instances which are mapped incorrectly, but the concept is related. *incorrect and unrelated link* include cases in which the linking is incorrect and also unrelated.

The analysis shows that the majority of mentions are linked to the correct (15 out of 25 instances) or at least a related (6 out of 25 instances) UMLS concept. Four instances within our sample were mapped to an unrelated UMLS concept.

While the majority of cases within our sample are normalized correctly, this module potentially introduces many errors. Note that as pointed out before about 30 % of entities are not linked and consequently not replaced at all. In addition, an incorrectly mapped and replaced mention, even if the concept might be closely related, may change the meaning of a claim drastically. Take the following example claim: 'COVID cause of breathlessness'. While *breathlessness* is correctly mapped to *dyspnea*, *COVID* is linked to and subsequently replaced by an unrelated concept: 'Covi Anxiety Scale Clinical Classification cause of dyspnea'. This leads us to believe that the unreliability of the linking module is the main reason why the verdict prediction performance for the normalized claims is comparably low.

| error type | # | mention | pred. concept |
|---|---|---|---|
| *correctly linked* | 15 | glandular fever | Infectious Mononucleosis |
| *incorr., related* | 6 | fibro flare | Fibromyalgia |
| *incorr. & unrelated* | 4 | COVID | Covi Anxiety Scale [...] |

Table 8: Number of error types within a sample of 25 entities along with examples.