Combatting Disinformation via Interactive Evidence Detection

Chris Stahlhut
Ubiquitous Knowledge Processing Lab (UKP-TUDA)
Research Training Group KRITIS
Department of Computer Science, Technische Universitt Darmstadt
https://www.ukp.tu-darmstadt.de/

Abstract

Correcting deeply held beliefs in false claims, such as vaccines cause autism, is incredibly hard. However, when discovered early enough, it might still be possible to debunk it before it reaches a critical mass, thereby stopping it from spreading any further. To critically evaluate any claim, a fact-checker needs evidence that either supports or contradicts it. This is very time consuming and requires a high level of expertise. In this paper, we present a tool which aims at supporting a fact-checker in finding evidence to support or contradict a claim by using interactively trained evidence detection methods. These methods learn directly from the user what is and isn't evidence as well as which claim it supports or contradicts. We illustrate the benefit of our tool with the description of two use cases; one in quickly reacting to new claims and one in studying social aspects of controversial topics in more detail. Furthermore, we present pre-trained evidence detection models that a user can fine-tune, thereby reducing the amount of training data required until the models benefit the user. This allows fact-checkers to react quicker to up-and-coming claims and thereby reducing the spread of incorrect ones.

1 Introduction

Once a particular false claim, such as *vaccines cause authism*, has been assimilated by a large group of people, it can be extremely hard to correct; it caused many parents to not vaccinate their children leading to a reemergence of measles despite innumerous research contradicting the claim. Therefore, when a new false claim appears it is best to limit its spread right from the start so that it does not get incorporated into the self-identity of a large group of people. Unfortunately, coming up with any claim is significantly easier than evaluating it, especially when not providing any evidence at all. This is particularly apparent in the Gish Gallop, a debating technique in which the user of this technique is making as many claims as possible so that their opponent cannot refute them all in time. Similar techniques are used not only by creationists, but also by climate change deniers and anti vaccine activists.

The risk of autism and other autistic-spectrum disorders did not differ significantly between children vaccinated with thimerosal-containing vaccine and children vaccinated with thimerosal-free vaccine (RR, 0.85 [95% confidence interval CI, 0.60-1.20] for autism; RR, 1.12 [95% CI, 0.88-1.43] for other autistic-spectrum disorders.

Furthermore, we found no evidence of a dose-response association (increase in RR per 25 g of ethylmercury, 0.98 [95% CI, 0.90-1.06] for autism and 1.03 [95% CI, 0.98-1.09] for other autistic-spectrum disorders).

Figure 1: Two examples of evidence contradicting the claim that thimerosal-containing vaccines, a mercury based preservative, have adverse health effects. Taken from Hviid et al. (2003).

For example, the claim *vaccines contain mercury and are therefore poisonous* is commonly repeated by people opposing vaccination. To evaluate this claim one has to first assess whether vaccines contain mercury and then find the best evidence regarding its safety. The *Centers for Disease Control and Prevention* already provides an overview regarding this particular claim[1] which states that research into the mercury based preservative *thimerosal* found no adverse health effects. Figure 1 shows two pieces of evidence contradicting the popular claim. Creating such an overview is time consuming; therefore, for many claims it will not be available, especially when this claim is not yet widely distributed and no one has had the time to evaluate it. However, modern Natural Language Processing (NLP) and machine learn-

[1] https://www.cdc.gov/vaccinesafety/concerns/thimerosal/index.html

ing techniques offer support in such an endeavour, particularly in finding evidence in large collections of documents.

Evidence Detection (ED) is a task within NLP in which a machine is supposed to find evidence related to a human defined hypothesis or claim. Figure 1 shows two examples of evidence that contradict the claim that vaccines are poisonous due to a mercury-based preservative. Building an ED model that distinguishes evidential from non-evidential sentences requires labeled data, that is by nature not available when a new claim is made. We therefore propose to interactively train two ED models that support a user in finding and curating evidence; one for finding evidential sentences or *evidence detection* and one for linking them to user defined topics or *evidence linking*.

To support users in finding and curating evidence, we obtained the source code for EDoHa (Evidence Detection fOr Hypothesis vAlidation) from Stahlhut et al. (2018) and integrated machine learning components. EDoHa is an annotation tool that aims at supporting researchers in finding and curating evidence. It allows a user to label sentences and link them to a self-defined hypothesis. In our modified EDoHa, we treat the user created data as training data for the machine learning components so that it can learn what a particular user is interested in.

Our contributions are three fold: (1) A tool that supports a user in finding evidence by learning what is and isn't evidence, (2) two use cases to illustrate how such a tool can be beneficial, and (3) software and data to pre-train evidence detection and evidence linking models in English.[2]

2 Related Work

Evidence detection Current research in evidence detection is often focussed on supporting decision making by finding evidence related to a user specified topic (Rinott et al., 2015), or to find and categorise supporting evidence (Hua and Wang, 2017). Other focal points have been the detection of claims and evidence in medical abstracts (Mayer et al., 2018a) and the classification of evidence into *comparative*, *significance*, *side-effect*, and *other* (Mayer et al., 2018b). However, these approaches do not consider the differences in interpretation of evidence between individual users, which Stahlhut et al. (2018) showed to vary. These differences are currently not addressed by existing research.

Claim validation Claim Validation (CV), or the automatic evaluating of claims, is becoming more important with the increased speed in which information can be disseminated. In the FEVER shared task (Thorne et al., 2018), CV is split into three separate tasks, namely document retrieval to find relevant documents from a large collection, ED to find the relevant pieces of evidence, and textual entailment to decide whether the claim follows from the evidence. Other approaches, such as TwoWingOS (Yin and Roth, 2018) and DeClarE (Popat et al., 2018) use end-to-end methods to jointly find evidence and determine how well it supports the claim. Another end-to-end approach was presented by Ma et al. (2019) which uses separate components for coherence, so that irrelevant evidence candidates are removed early on, and entailment for the final predictions. Both components are initially pre-trained and then further trained jointly with the other components of the end-to-end model. Other approaches focus more on the entailment part. Wang (2017) released a dataset in which the validity of a claim can be evaluated based on the claim and metadata, such as the context of the claim and the claim holder. This dataset was extended by Alhindi et al. (2018) who added the justification of the original annotators and showed that adding this information improves the performance of a CV model. While these approaches show promising results, neither of them are intended to be trained interactively which does limit their applicability to genuinely new claims.

End user systems Enabling users to directly access complex NLP components has become common in the closely related field of argument mining. MARGOT (Lippi and Torroni, 2016) allows a user to input multiple sentences and classifies sequences of words as argumentative or not. Others offer to search large web corpora for arguments related to a user specified topic in English (Stab et al., 2018) or German (Stahlhut, 2018). A similar tool named *args.me* was presented by Ajjour et al. (2018) allows the user to navigate overarching topics of arguments. So far, these tools have remained static without learning from the user's input.

[2]The source code of EDoHa and the data as well as the code for the pre-trained models is available under `https://github.com/UKPLab/EDoHa`

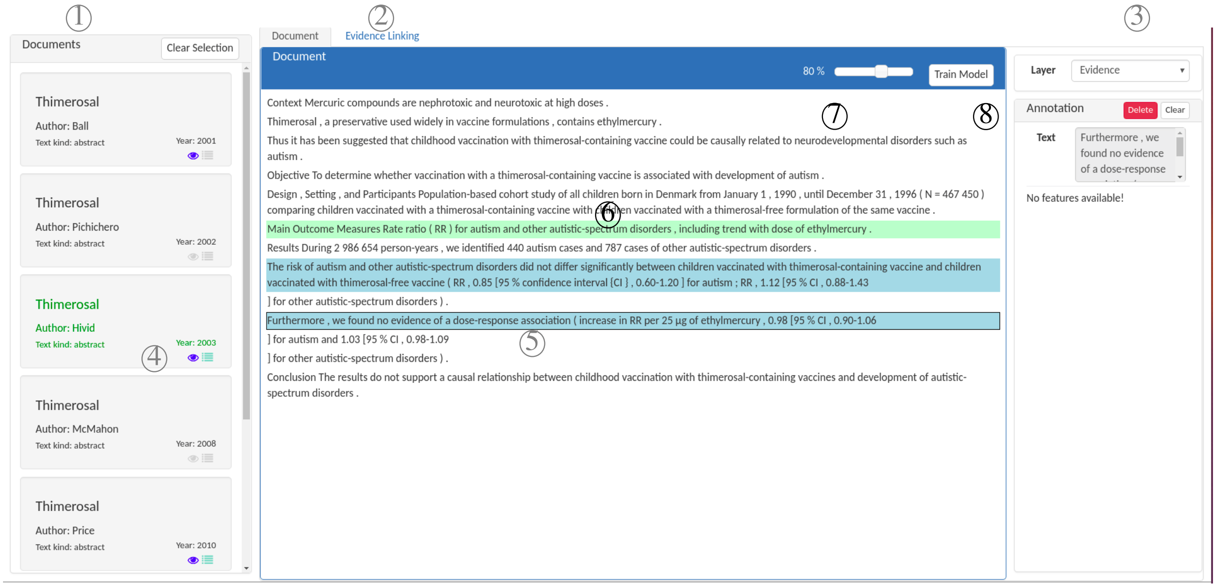

Figure 2: Screenshot of EDoHa. The Document view allows the user to label sentences as evidential. The grey number indicate components that where already implemented by Stahlhut et al. (2018) and the black numbers indicate our additions.

Interactive machine learning for NLP Learning directly from a user is a common approach to reduce the cost of creating labeled datasets. For instance, active learning focusses on reducing the number of times a machine learner needs to query an oracle. This approach has already been used for multi-domain sentiment detection (Huang et al., 2017) and to learn a reward function for summarisation (Gao et al., 2018). Another motivation for interactive learning is to speed up human annotators, for instance in dependency parsing (Ulinski et al., 2016). However, speeding up a human annotator without providing them with an additional benefit is still focussed on the machine side of an interactive system. On the human side however, the purpose of machine learning is to serve requirements of the user and not the system. Yimam et al. (2017) learns from a human user who labels entities in medical abstracts, making suggestions after a manual annotation and also learns to suggest relations between entities. Generally, these approaches aggregate the data across users which does influence their annotations (Fort and Sagot, 2010) which is not beneficial when supporting individual users.

3 EDoHa

EDoHa is a web based annotation tool on top of the INCEpTION platform (Klie et al., 2018). The INCEpTION platform is primarily an annotation tool, therefore offering a variety options. A user can choose between different types for a specific label, e.g. named entities or POS tags, and different granularities, such as individual or multiple tokens within and across sentences. It also offers support in aggregating the annotations of multiple annotators. Since our goal is to support the individual user in their work for which annotation is just a necessary step, we modified the user interface considerably. Our extension of EDoHa consist of modification of both the Document view, as well as the Hypotheses/Evidence view, which we renamed to Evidence Linking, and the inclusion of interactively trainable evidence detection and evidence linking models. Figure 2 shows EDoHa's Document view in which a user can label sentences as evidence. The grey circled numbers mark components already presented by Stahlhut et al. (2018) while the black circled numbers mark components added by us.

3.1 Evidence Detection

The user interface of EDoHa consists of two parts. The list of available documents on the left ① and the Document or Evidence Linking view ②. In the Document view it is also possible to show the status of the currently selected piece of evidence ③ on the right hand side. The currently opened document is highlighted with green colour in the Documents collection ④.

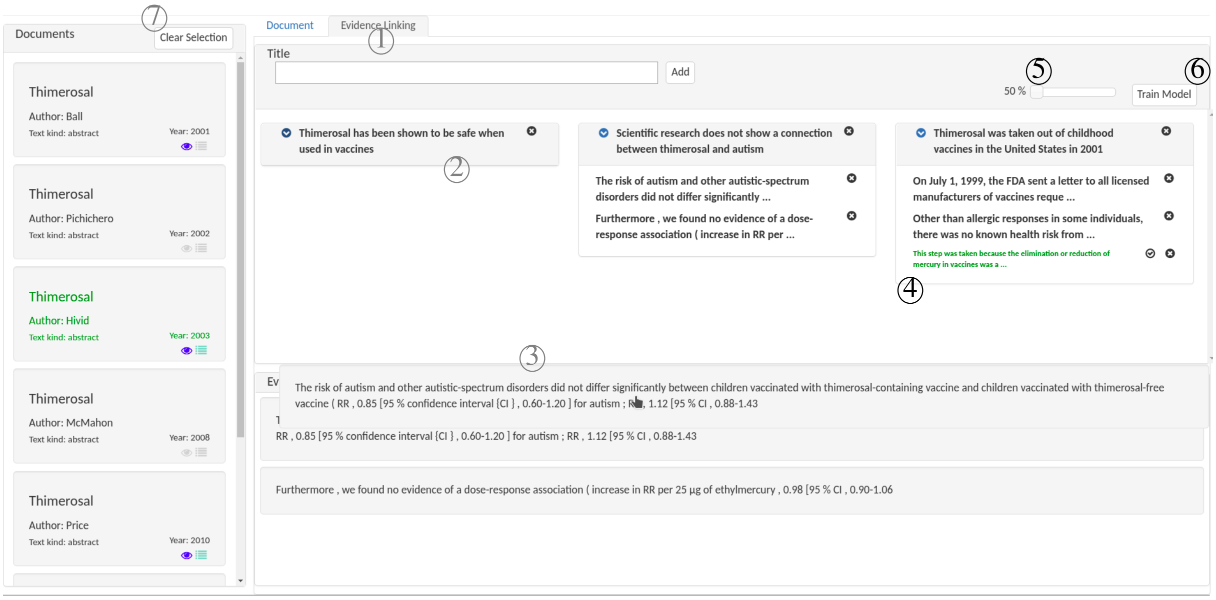

Figure 3: Screenshot of the Evidence Linking view of EDoHain which pieces of evidence the user selected or accepted can be linked to topics. Black numbers indicate our modifications.

While reading a document, the user and EDoHa interact as follows: The user can annotate individual sentences as evidence by clicking on them ⑤, EDoHa can suggest sentences that the user might be interested in with green background colour ⑥ given the previously annotated documents. If the user decides to accept the annotation, they only need to click on the sentence and the background colour changes to blue. To avoid being overloaded with bad suggestions, the user can adjust a confidence threshold between 50% and 100% ⑦. Given that a model will very rarely give a 100% confidence, setting the threshold to this value practically turns off the suggestions entirely. The user can also trigger the training of a new model by clicking the `Train Model` button ⑧. If the user decides that a sentence they previously deemed evidential is not, they can select this sentence and click on the red `Clear` button in the status column on the right hand side.

3.2 Evidence Linking

In the Evidence Linking view, figure 3, the user can group pieces of evidence together, e.g. to link evidence to a particular hypothesis or group evidence by similar aspects, and give each group a title. The user can create a group by defining a title ①, e.g. *Thimerosal has been shown to be safe when used in vaccines*, which is then added to the group view ② where the user can also change the title if necessary. The user can then link pieces of evidence to this group by dragging them from the list of available evidence at the bottom and dropping them into a group. Similarly to the Document view, the Evidence Linking view also uses the data the user creates to train a machine learning model. This model can also make suggestions, which are visible in a smaller green font ④ and the user can accept or reject this suggestion by clicking on the ✔ or ✖ respectively. The confidence threshold can also be adjusted by the user ⑤ and a new model can be trained by clicking the `Train Model` button ⑥. By clicking the `Clear Selection` button ⑦, the user indicates that they do not want to be limited to the evidence from the currently selected document and the list of available evidence is extended by the manually labeled sentences from all documents.

3.3 Interactive Learning

To support the user in either task, EDoHa can interactively train two separate models; one for each task the user performs.

Evidence detection Each time the user clicks on the `Train Model` button, EDoHa takes all documents the user has opened and treats all sentences as labeled training data. It then re-trains its internal ED model on this data and afterwards this model is used to suggest sentences as evidence to the user. Figure 4 illustrates these interactions. The output of the evidence class for each sentence is then used as a confidence value which will be compared against the user defined confidence threshold in the Document view.

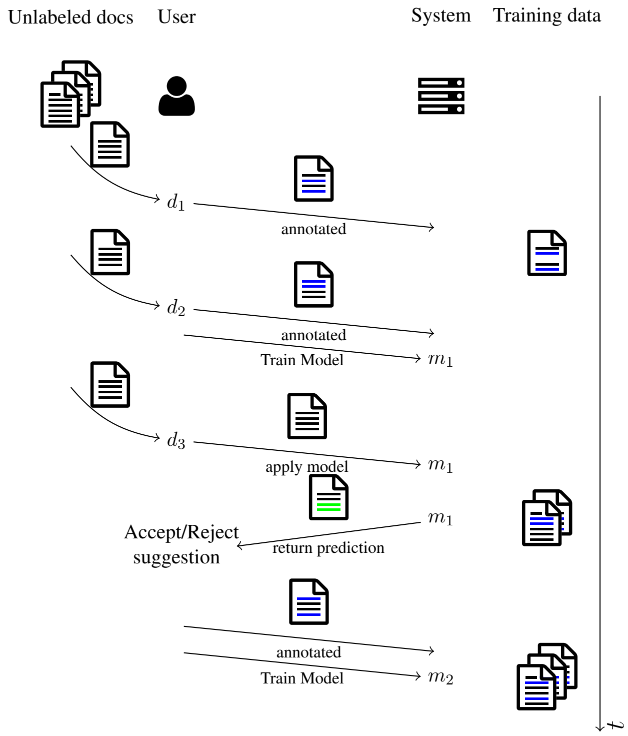

Figure 4: After reading the first two documents d_1 and d_2, the user clicks on `Train Model` to train the first model m_1. This is then used to suggest evidence on the third document d_3, which the user then corrects and after clicking again on `Train Model`, the system trains the second model m_2 on the training documents d_1, d_2, and d_3.

Evidence linking In the Evidence Linking view, when the user clicks on the `Train Model` button, EDoHa first generates as many random links between pieces of evidence and topics as there are manually created ones. The purpose of the weakly labeled data is to keep the dataset balanced until the user has created enough non-links to train an evidence linking model. To avoid creating a link the user has not yet had the time to create, the weakly labeled data will only contain pieces of evidence linked to at least one group for the creation of the random non-links. The model's output for the link class can also be used as a confidence value. If the user rejects a suggested link, it will also be stored and used as training data for any future evidence linking model. The number of weakly labeled non-links is then decreased by one so that it is still a balanced dataset. If the user rejects as many or more links than they accept or create, EDoHa will not generate any weakly labeled data. Figure 5 illustrates this interactive training.

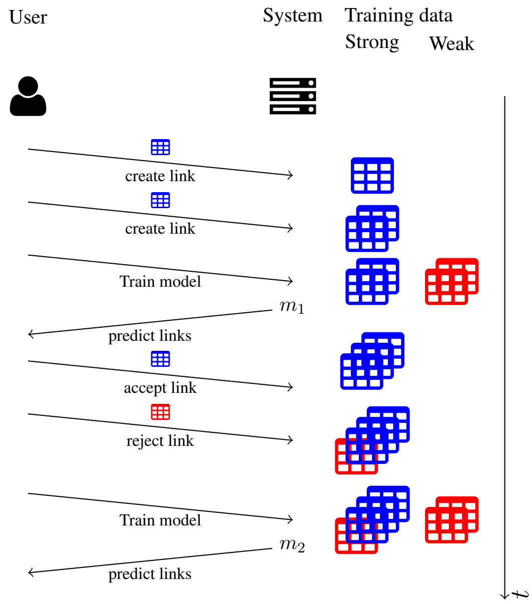

Figure 5: The user first creates two links and clicks the `Train Model` button. Then EDoHa generates two weakly labeled non-links and trains the model m_1 which EDoHa then uses to predict links between evidence and titles. After the user accepts one and rejects one link, they again train a new model. To keep the dataset balanced, EDoHa only needs to generate two weakly labeled non-links and train the model m_2.

3.4 Model definition

For both evidence detection and evidence linking, we chose to use TensorFlow[3] based models. This allows us to select from a wide variety of architectures, as long as the following conventions are held:

Evidence detection The evidence detection model m_{ed} is a sentence classification model that classifies a candidate sentence s as either evidential or not $m_{\text{ed}} : s \rightarrow [0,1]^2$, where the value of the last index represents the confidence output for the evidence class.

Evidence linking The evidence linking model m_{el} requires that the header τ as well as the candidate evidence e are presented and classifies them as either linked or not. To allow for the inclusion of cross attention between the topic and the candidate evidence, we also added the length of both sequences to the input. Hence, the model

[3] https://www.tensorflow.org/

$m_{\text{el}} : (\tau, \text{len}(\tau), e, \text{len}(e)) \rightarrow [0,1]^2$, where again the last dimension represents the confidence of the two input candidates being linked.

Both models can also be pre-trained on external data before being used by EDoHa.

4 Use Cases

We envisioned two different uses cases in which EDoHa offers a benefit to the user. First, in supporting a human fact-checker to evaluate the validity of an up-and-coming claim; second, in supporting a researcher in the political or social sciences in analysing interviews.

4.1 Faster Debunking

Suppose the claim that *vaccines contain mercury and are therefore poisonous* is increasing in popularity and there is not yet any good overview that debunks this claim. A fact-checker who is interested in evaluating this claim might start by searching into the background of this claim. They find that older vaccines contain a component called *thimerosal* which acts as a preservative that protects the vaccine from bacterial contamination. They then search the PubMed database[4] for medical abstracts and other sources for documents which might contain evidence to support or contradict the original claim. Once the fact-checker starts reading these documents, they notice that some of the research focusses on general health outcome and others specifically on autism.

They then import their documents into EDoHa and start labeling the evidence they find regarding health effects and thimerosal, such as *The adjusted odds ratios (95% confidence intervals) for ASD [Autism Spectrum Disorder] associated with a 2-SD increase in ethylmercury exposure were 1.12 (0.83–1.51) for prenatal exposure, 0.88 (0.62–1.26) for exposure from birth to 1 month, 0.60 (0.36–0.99) for exposure from birth to 7 months, and 0.60 (0.32–0.97) for exposure from birth to 20 months.*[5] in the Document view.

After changing into the Evidence Linking view, they create multiple groups, such as *Scientific research does not show a connection between thimerosal and autism*[6], or *Thimerosal has been shown to be safe when used in vaccines* and link the previously found evidence to it.

The fact-checker then triggers the training of the evidence selection and evidence linking models. When opening the next document in EDoHa, it already shows sentences that might be evidential. The fact-checker accepts the suggestions they agree with and ignores the ones they don't consider evidence. In the Evidence Linking view, EDoHa also shows possible links between topics and evidence the fact-checker accepted or labeled manually. Of these suggestions the fact-checker again accepts the correct ones but also rejects the incorrect ones.

After being satisfied with the results of the search, the fact-checker proceeds to write an overview presenting the claims and evidence in a dispassionate form that can be referenced in any discussion if the claim appears. Afterwards, they can approach a new claim with new documents and pieces of evidence.

4.2 Research in Political and Social Science

When conducting qualitative research in the political or social sciences, researchers often first conduct structured or semi-structured interviews and then extract important statements of the interviewees that are supporting or contradicting the researchers hypotheses. They generally first transcribe the interviews and then use an annotation tool (or coding tool in their terminology) to highlight and extract statements which they find relevant for their research. The latter task is one for which EDoHa is well suited to support the researcher.

Say a researcher investigates the self perception of activists against Genetically Modified Organisms (GMO). They might start by conducting interviews with many different activists. After transcribing the interviews they load them into EDoHa and start by labeling interesting statements. The researcher can then aggregate these statements into groups, such as *Complete rejection of any GMO* which contains statements such as *I don't want genes in my food*. However, other groups contain more nuanced points, such as *Unregulated GMO crops might out-compete domestic ones* or *GMO crops lock farmers into exclusive contracts with one supplier*. The researcher can then take these results and conduct more detailed interviews with a select group of activists.

[4] https://www.ncbi.nlm.nih.gov/pubmed/

[5] The example is taken from McMahon et al. (2008).

[6] The examples are taken from the CDC webpage Thimerosal in Vaccines: https://www.cdc.gov/vaccinesafety/concerns/thimerosal/index.html

In this case, EDoHa can support the researcher by learning what kind of statements the researcher is interested in and how to link them to groups.

5 Pre-trained Evidence Detection Models

To jump-start the interactive training process, EDoHa can load pre-trained models which a user can then further fine-tune interactively. These models can be trained on any dataset and use any internal architecture as long as they are implemented in TensorFlow and their input, output, target, and training operations comply with the convention used by EDoHa. As an example, we prepared an ED dataset by splitting it into evidence detection and evidence linking tasks and trained two models as a starting point for interactive fine-tuning.

5.1 Dataset and Data Preparation

As data for the pre-trained models we selected the ED dataset published by Shnarch et al. (2018). It consists of around 4.000 training sentences that are either evidential regarding a particular topic or not and 1.700 test sentences with different topics. The topics are worded as statements for action, e.g. *We should increase gun control* or *We should abolish temporary employment*.

For the evidence linking task, we first extracted all topic evidence pairs and then created equally many random non-evidential topic evidence pairs. A non-evidential pair consists of a sentence that is evidential to one topic and a randomly chosen topic to which this sentence is not evidential. We conducted this for the training and testing datasets separately to avoid any contamination of the testing data. Table 1 shows the statistics of the training and testing data for both tasks.

Evidence detection	Topics	Sentences	Evidence
Train	83	4065	1499
Test	35	1718	683

Evidence linking	Topics	Links	No Links
Train	83	1499	1499
Test	35	683	683

Table 1: Statistics on the ED datasets used to pre-train the evidence detection and evidence linking models.

5.2 Model Selection and Evaluation

Evidence detection To detect evidence, we selected a Bidirectional Long-Short Term Memory (BiLSTM) with 100 nodes as encoder and a dense layer for classification. As input features, we used 50 dimensional GloVe embeddings (Pennington et al., 2014). We pre-trained the model for 10 epochs with a learning rate of 0.001.

Evidence linking To suggest the membership of a piece of evidence in a particular group, we built a siamese BiLSTM in which both the topic of a group as well as the candidate evidence are encoded by a separate BiLSTM each. Both BiLSTMs have 100 nodes which are concatenated and used as input to a dense layer. As with the evidence detection model, we also used 50 dimensional GloVe embeddings as input features. We trained the evidence linking model for 20 epochs with a learning rate of 0.001.

Model Evaluation We evaluated the pre-trained models in the testing data of both, evidence detection and evidence linking datasets. Table 2 shows the macro averaged scores, as well as the scores of the evidence class for the evidence detection task and the macro averaged scores for the evidence linking task. The evidence detection model performs especially well in evidence recall. The model for evidence detection performs significantly better than the model for evidence linking, indicating that the evidence linking task is significantly more complicated. However, the purpose of these models is to address the cold-start problem for a model that adapts to an individual user and not to be state-of-the-art on the pre-training datasets.

	Precision	Recall	F1
Evidence detection			
Macro	0.644	0.650	0.635
Evidence	0.533	0.710	0.609
Evidence linking			
Macro	0.542	0.542	0.542
Link	0.543	0.537	0.540

Table 2: Performance of the pre-trained models for evidence detection and evidence linking. The evidence linking results are macro-averaged.

6 Conclusion

In this paper we presented an extension of EDoHa which learns directly from a user to find evidence and link them to topics. We laid out two use cases in which such a tool can be beneficial. First, in speeding up the debunking of an up-and-coming claim before it reaches a critical mass of believers and becomes almost impossible to correct; and second, in studying the activism around GMO which contains multiple different kinds of concerns which can be easier found and grouped together using EDoHa. Additionally we provided two pre-trained models which can already distinguish evidential from non-evidential sentences and link them to controversial claims. These models can then be further fine-tuned by a user towards their specific needs.

Our next steps are to evaluate to what degree the interactively learned evidence detection and evidence linking models speed up the working process of actual fact-checkers and researchers.

Acknowledgements

This work has been supported by the German Research Foundation (DFG) as part of the Research Training Group KRITIS No. GRK 2222/1.

References

Yamen Ajjour, Henning Wachsmuth, Dora Kiesel, Patrick Riehmann, Fan Fan, Giuliano Castiglia, Rosemary Adejoh, Bernd Fröhlich, and Benno Stein. 2018. Visualization of the Topic Space of Argument Search Results in args.me. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 60–65, Brussels, Belgium. Association for Computational Linguistics.

Tariq Alhindi, Savvas Petridis, and Smaranda Muresan. 2018. Where is Your Evidence: Improving Fact-checking by Justification Modeling. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 85–90, Brussels, Belgium. Association for Computational Linguistics.

Karën Fort and Benoît Sagot. 2010. Influence of Pre-Annotation on POS-Tagged Corpus Development. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 56–63, Uppsala, Sweden. Association for Computational Linguistics.

Yang Gao, Christian M. Meyer, and Iryna Gurevych. 2018. APRIL: Interactively Learning to Summarise by Combining Active Preference Learning and Reinforcement Learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, volume Long Papers.

Xinyu Hua and Lu Wang. 2017. Understanding and Detecting Supporting Arguments of Diverse Types. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 203–208, Vancouver, Canada. Association for Computational Linguistics.

Lulu Huang, Stan Matwin, Eder J. de Carvalho, and Rosane Minghim. 2017. Active Learning with Visualization for Text Data. In *Proceedings of the 2017 ACM Workshop on Exploratory Search and Interactive Data Analytics*, ESIDA '17, pages 69–74, Limassol, Cyprus. ACM.

Anders Hviid, Michael Stellfeld, Jan Wohlfahrt, and Mads Melbye. 2003. Association Between Thimerosal-Containing Vaccine and Autism. *JAMA*, 290(13):1763–1766.

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The INCEpTION Platform: Machine-Assisted and Knowledge-Oriented Interactive Annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9, Santa Fe, New Mexico. Association for Computational Linguistics.

Marco Lippi and Paolo Torroni. 2016. MARGOT: A web server for argumentation mining. *Expert Systems with Applications*, 65:292–303.

Jing Ma, Wei Gao, Shafiq Joty, and Kam-Fai Wong. 2019. Sentence-Level Evidence Embedding for Claim Verification with Hierarchical Attention Networks. page 12.

Tobias Mayer, Elena Cabrio, Marco Lippi, Paolo Torroni, and Serena Villata. 2018a. Argument mining on clinical trials. In *Computational Models of Argument: Proceedings of COMMA 2018*, volume 305 of *Frontiers in Artificial Intelligence and Applications*, pages 137–148, Warsaw, Poland. IOS Press.

Tobias Mayer, Elena Cabrio, and Serena Villata. 2018b. Evidence Type Classification in Randomized Controlled Trials. In *Proceedings of the 5th Workshop on Argument Mining*, pages 29–34, Brussels, Belgium. Association for Computational Linguistics.

A. W. McMahon, J. K. Iskander, P. Haber, M. M. Braun, and R. Ball. 2008. Inactivated influenza vaccine (IIV) in children <2 years of age: Examination of selected adverse events reported to the Vaccine Adverse Event Reporting System (VAERS) after thimerosal-free or thimerosal-containing vaccine. *Vaccine*, 26(3):427–429.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

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

Ruty Rinott, Lena Dankin, Carlos Alzate Perez, Mitesh M. Khapra, Ehud Aharoni, and Noam Slonim. 2015. Show Me Your Evidence – an Automatic Method for Context Dependent Evidence Detection. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 440–450, Lisbon, Portugal. Association for Computational Linguistics.

Eyal Shnarch, Carlos Alzate, Lena Dankin, Martin Gleize, Yufang Hou, Leshem Choshen, Ranit Aharonov, and Noam Slonim. 2018. Will it Blend? Blending Weak and Strong Labeled Data in a Neural Network for Argumentation Mining. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 599–605, Melbourne, Australia. Association for Computational Linguistics.

Christian Stab, Johannes Daxenberger, Chris Stahlhut, Tristan Miller, Benjamin Schiller, Christopher Tauchmann, Steffen Eger, and Iryna Gurevych. 2018. ArgumenText: Searching for Arguments in Heterogeneous Sources. In *Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, New Orleans, USA. Association for Computational Linguistics.

Chris Stahlhut. 2018. Searching Arguments in German with ArgumenText. In *Proceedings of the First Biennial Conference on Design of Experimental Search & Information Retrieval Systems*, volume 2167 of *CEUR Workshop Proceedings*, page 104, Bertinoro, Italy.

Chris Stahlhut, Christian Stab, and Iryna Gurevych. 2018. Pilot Experiments of Hypothesis Validation Through Evidence Detection for Historians. In *Proceedings of the First Biennial Conference on Design of Experimental Search & Information Retrieval Systems*, volume 2167 of *CEUR Workshop Proceedings*, pages 83–89, Bertinoro, Italy.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2018. The Fact Extraction and VERification (FEVER) Shared Task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 1–9, Brussels, Belgium. Association for Computational Linguistics.

Morgan Ulinski, Julia Hirschberg, and Owen Rambow. 2016. Incrementally Learning a Dependency Parser to Support Language Documentation in Field Linguistics. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 440–449, Osaka, Japan. The COLING 2016 Organizing Committee.

William Yang Wang. 2017. "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

Seid Muhie Yimam, Steffen Remus, Alexander Panchenko, Andreas Holzinger, and Chris Biemann. 2017. Entity-Centric Information Access with the Human-in-the-Loop for the Biomedical Domains. In *Biomedical NLP Workshop*, Varna, Bulgaria.

Wenpeng Yin and Dan Roth. 2018. TwoWingOS: A Two-Wing Optimization Strategy for Evidential Claim Verification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 105–114, Brussels, Belgium. Association for Computational Linguistics.