Creating a dataset for Fallacy Detection

Computational Semantics for Natural Language Processing - Project Report, Spring 2022

Gauthier Boeshertz, Alvaro Caudéran, and **Franz Nowak**
ETH Zürich - Swiss Federal Insitute of Technology
{gboeshertz, acauderan, fnowak}@ethz.ch

Abstract

Faulty reasoning can lead someone to reach the wrong conclusions from correct premises. Just as spell checkers have been assisting writers for decades in avoiding typographical errors, a *fallacy checker* could aid in recognising and avoiding fallacious arguments and thereby make discourse more rational and more instructive. This work introduces the task of fallacy detection, as well as a custom dataset for training a binary classifier to distinguish sound reasoning arguments from fallacious arguments. 10,364 statements labeled as either fallacious or non fallacious were aggregated from multiple sources to create a balanced and useful dataset for this task. A classifier that we trained on the training subset of this data achieves an accuracy of 89% on the test set.

1 Introduction

The information age has brought about many advances in the way we interact with each other and the world. Instant access to vast amounts of information could in theory make every human alive today better informed than scholars would have been only a few centuries ago. However, the flip side of the flood of available information is a vast increase in misinformation and bad reasoning which, spreading e.g. through social media (Hristakieva et al., 2022), can be used to reach and influence millions of people. One part of this problem is fake news, i.e. factually wrong information, which needs to be debunked by trusted sources such as human or automated fact checkers (see Thorne et al. (2018) and others).

However, even with correct information, one can reach the wrong conclusion through faulty reasoning. Arguments that sound reasonable can contain logical fallacies which convince people to accept wrong conclusions from correct premises. Take for example the following statement:

> "Jeanne Calment smoked until she was 117 years old. Hence smoking isn't actually bad for you."

This is a faulty generalisation, because a single case where smoking did not cause health problems in a person does not mean that on average it does not cause health problems.

If there was an automatic way to recognise such instances of fallacious reasoning reliably, it could be used by social media or news sites to alert readers that the arguments they are reading might not be correct.

Another use for such technology would be that journalists, politicians, lawyers and anyone else who aims to write sound logical arguments could use it to make better arguments and avoid fallacies. When people started using computers to write text, the advent of spell checkers soon helped them avoid typographical errors. More recently, there have been applications that can detect grammatical errors as well, and now automated fact checkers even allow analysing the factual correctness of any text in real time. The next step in this "hierarchy of computer aided writing" would be a tool that indicates whether an argument just written might contain fallacious lapses of reasoning.

Recent work was done on the task of fallacy *classification* by Jin et al. (2022), who trained a classifier to recognise between 13 different types of fallacies using tools such as Natural Language Inference (NLI) and combining them with transformer language models. Specifically, argument mining (Lawrence and Reed, 2020) is used as a preprocessing step to extract the underlying logical form of the text and use the resulting features to train a classifier.

The datasets introduced by Jin et al. (2022) contained *only* fallacious statements, meaning it was a classification of the fallacy type under the assump-

tion that the argument is a fallacy.

Our contributions in this work can be summarised as follows:

- We introduce the task of *fallacy detection*, i.e. the binary classification of arguments into fallacies and non-fallacies.
- We supplement the fallacy datasets from Jin et al. (2022) with counterexamples (non-fallacious statements).
- Furthermore, we also add data of both fallacies and non-fallacies we collected ourselves from real world online discussions.
- Finally, we use the preprocessing and modeling techniques from Jin et al. (2022) to show how this data can be used to train high accuracy classifiers for this task.

In future work, combining detection and classification of fallacies will allow building an end-to-end pipeline that could be used, e.g. in the form of a plugin, on any real world English language text to detect whether a statement is a fallacy, and if so, what type of a fallacy it is.

2 Data collection

One of our main contributions is creating a high quality dataset that can be used to train a classifier to perform the task of binary fallacy detection. In the following, we outline the makeup of this dataset.

2.1 LOGIC and LOGICCLIMATE

Jin et al. (2022) already provided two datasets containing fallacies, LOGIC and LOGICCLIMATE. For the LOGIC dataset, around 1700 statements were automatically crawled from the educational websites Quizziz, and study.com, and ProProfs. Around 600 more examples were taken from manual google searches. Notably most of the examples are quite artificial, in that they were used to teach students about logic, but probably don't occur in that way in the wild. For instance, the dataset contains the example:

> "Jack is a good athlete. Jack comes from Canada. Therefore, all Canadians are good athletes."

As is apparent, a newspaper or politician's speech is unlikely to contain this, however it is very clear what the logical structure is and hence where the fallacy lies.

On the other hand, for the LOGICCLIMATE dataset, Jin et al. (2022) collected news articles from the Climate Feedback website, asking human annotators native in English to extract and annotate fallacies. Naturally, these fallacies are much more difficult to identify as such as they are usually much longer and less logically clear. For example, the following instance from the LOGICCLIMATE dataset was classified by annotators as an *Appeal to emotion* (a type of fallacy):

> "There are now, trapped in Arctic ice, diseases that have not circulated in the air for millions of years — in some cases, since before humans were around to encounter them. Which means our immune systems would have no idea how to fight back when those prehistoric plagues emerge from the ice."

However, it is by no means obvious from the logical structure of this sentence that it is a fallacy, rather the annotation stems from the emotive connotations of the words "disease", "fight", and "plague". This exemplifies why this dataset is referred to as the *challenge set*.

2.2 LOGICVALID and LOGICCLIMATEVALID

In order to be able use the two datasets above for fallacy detection, we extended them by creating contrasting examples for each of them.

Inspired by works by Agarwal et al. (2022) and Boschi et al. (2019), we sourced most of our own data from Kialo[1]. Kialo is an online discussion platform where users can write, respond to, and rate arguments as part of debates on a variety of topics.

We decided on Kialo for a large part of our data sourcing because it contains arguments from actual debates, i.e. real exchanges between people, enabling us to train and validate our models on more realistic examples than the artificial statement's from LOGIC.

One source of a limited number of already downloaded Kialo discussions was created by Agarwal et al. (2022). It has the advantage that it also contains scraped quality ratings, given by Kialo users to each statement, which we use as a proxy for logical validity, keeping only the best rated ones.

[1] https://www.kialo.com/

Using this rating and some filtering, we selected the 2,200 highest rated arguments from the whole corpus of arguments for the LOGICVALID dataset, to be used as contrast against the LOGIC fallacies, and 721 arguments that are climate related for the LOGICCLIMATEVALID dataset, to be used against the LOGICCLIMATE fallacies dataset.

The filtering was necessary to address the fact that many arguments were too short and thus did not make for good contrasting examples. To remedy this, we only selected longer sentences by filtering the number of punctuation marks (specifically, commas and full stops) in each argument. We also removed arguments that contained normative words such as "should" which we assumed to be used in normative statements such as opinions, rather than logical statements.

To identify climate related arguments from the corpus for LOGICCLIMATEVALID, we selected them by looking for keywords, including "climate change", "emissions" and "global warming". As both climate and non climate related counterexamples were taken from the same corpus, there was some overlap between the two datasets, however, the number of duplicates is a single digit number.

2.3 KIALO and KIALOVALID

Finally, the last two datasets we sourced from Kialo ourselves to have a larger amount of discussions to use, since the corpus from Agarwal et al. (2022) was collected on 28 January 2020 and hence has fewer content. As a comparison, they collected 1,560 discussion threads, whereas we collected 2,622.

The hierarchical structure of the debates on Kialo allows us to use one user's response to another user's argument as a label for that argument. For instance, one user might make the following statement (taken from our data):

> "Solipsism has been proven to be nonsensical by some of the greatest minds in history. As such, discussing it is similarly non-sensical."

to which another user replies:

> "An argument from authority is [...] a fallacy"

Given that the parent statement was thus marked as a fallacy by another user, we can use it as a positive example in our dataset. Similarly, to find negative counterexamples, we can choose one of the fallacious argument's siblings in the discussion tree, and if it has no annotations claiming it is a fallacy we assume it is valid.

To get enough discussions from Kialo, we downloaded all discussions that were linked on the explore page through a list of 2,152 tags, via the download API endpoint.

We then parsed these discussions into a suitable custom data structure we crated for kialo discussion data that preserves the hierarchy and allows efficient look-ups of claims, their parents, children, siblings, and meta data[2].

Next, we extracted statements of interest for the KIALO dataset containing fallacies by extracting all statements whose children statements contain at least one keyword from a list of 35 which correspond to different type of fallacies.

As we did not fully trust the users' assessment nor our extraction strategy (for instance the sentence "The above is not a fallacy" would result in a fallacy label because it contains the word "fallacy") we manually went through the KIALO fallacies and re-labelled them as fallacy or non-fallacy.

For the contrasting dataset, KIALOVALID, we extract counterexamples by selecting siblings of the previously found fallacies which themselves are not marked as fallacies by users. The reason for using siblings is to get counterexamples that are topically close to the fallacious examples.

Finally we cleaned both the KIALO and KIALOVALID datasets, removing unsuitable examples as outlined in section 2.2. Also we removed links and newlines within arguments in all the datasets.

2.4 Dataset overview

In total, we collected 10,364 data points, with 4,376 fallacies and 5,988 non-fallacies. Table 1 shows the size of each dataset, with each fallacy dataset paired with its corresponding contrast dataset.

Dataset name	number of statements
LOGIC	2,449
LOGICVALID	2,200
LOGICCLIMATE	1,079
LOGICCLIMATEVALID	721
KIALO	848
KIALOVALID	3,067

Table 1: Number of examples in each dataset

[2]https://github.com/franznowak/kialoparser

3 Modeling techniques

Like Jin et al. (2022), our approach for modeling the classifier utilises Natural Language Inference (NLI) models (Yin et al., 2019). NLI models can determine whether a given hypothesis "follows" from a specific premise (entailment), "unfollows" (contradiction) or whether they are "undetermined" (neutral) with regard to each other. This allows us to construct zero-shot classifiers since the task we are trying to solve is answering a question that can be formulated as a hypothesis.

Furthermore, it can also useful for pretraining a model, as more binary questions can be asked without changing the structure of the model.

For obtaining the logical structure of the statements, we used the *Structure Aware Premise* module from Jin et al. (2022), which makes the model attend to the word order and their logical structure. We defer to section 3 of their paper for the details.

As the hypothesis, we used the statement "This statement is a fallacy". Depending on whether the model predicts entailment or contradiction, our classifier selects the output label (fallacy or non-fallacy, respectively).

One thing that is important to mention is that, in our model, we discarded the neutral case as we are working in a binary classification setting, since an argument is either a fallacy or it is not. This means that both contradiction and undetermined entailment count as non-fallacy for our purposes.

To create our classifier, we fine-tuned four different transformer models: BERT (Devlin et al., 2018), Roberta (Liu et al., 2019), DeBERTa small (He et al., 2021) and Electra (Clark et al., 2020). The models were taken from and fine tuned using Huggingface's transformer libraries (Wolf et al., 2019). We will compare the results obtained with each model in the following section.

4 Experiments

We are interested in testing how accurately a classifier can predict (binary) fallaciousness of statements when trained on our data. Given that our data comes from multiple sources, we used different ways of splitting the data into train, validation, and test sets, taking the provenance of the different subsets of the dataset into account.

To this end we designed 4 experiments. The first, which yields our main results, uses all of the data we acquired and splits it into training, validation and a holdout test set.

		Non-fallacies	Fallacies
Combined data	Train set	4804	3487
	Val set	572	464
	Test set	612	425
Climate test	Train set	4201	2650
	Val set	1066	647
	Test set	721	1079
Logic climate test	Train set	2466	1946
	Val set	601	503
	Test set	721	1079
Logic reduced climate test	Train set	2440	693
	Val set	627	157
	Test set	721	1079
Kialo climate test	Train set	2441	691
	Val set	626	157
	Test set	721	1079

Table 2: Description of the dataset splits used in the experiments

For the second experiment, we trained the different classifier models on all data unrelated to climate change, and then tested the ability to generalise to the unseen climate related domain. Finally, the last two experiments examine how well the model trains on the fallacies and counterexamples we obtained by scraping real world sources, compared to the Logic (Jin et al., 2022) dataset (and the same counterexamples). For each of these experiments a quantitative description of the dataset is shown in table 2.

4.1 Main results

The combination of all the data results in a dataset of 10,364 arguments, of which 4,376 are fallacious. We split this data into training, validation and testing sets, resulting in 8,291 arguments for training, 1,036 for validation and 1,037 for testing. On this data we tested two zero shot classifiers trained on MNLI tasks, namely, Bart large (Lewis et al., 2020) and Roberta large (Liu et al., 2019). Then, we fine-tuned BERT (Devlin et al., 2018) and performed an ablation study to see whether using Structure-Aware Premises as in (Jin et al., 2022) was a good idea. We also fine-tuned 3 other transformer models, Roberta (Liu et al., 2019), DeBERTa small (He et al., 2021) and Electra (Clark et al., 2020).

The results of these experiments, with macro averages, are shown in table 3. As we can see, the zero shot classifiers perform barely better than a random classifier. This is expected as classifying

	P	R	F1	Acc
Bart-MNLI	51	51	51	53
Roberta-MNLI	52	52	52	54
Bert	87	85	86	87
Bert + SA P	86	84	85	86
Bert + Hypo	89	85	86	87
Bert + SA P + Hypo	87	85	86	87
Electra	88	86	87	87
Roberta	**90**	**88**	**89**	**89**
deBERTa	**90**	**88**	**89**	**89**

Table 3: Macro average of model performance in % on the combined dataset

arguments as fallacies is a challenging task, even for humans. However, fine tuning models drastically improves their performance over zero shot models. Indeed, BERT achieves an accuracy of 87%, which is surprisingly high given the complexity of the task. The ablation study on the inputs shows that it performs better when only the hypothesis is used. This is expected as fallacies rely not only on the structure of the arguments but also on the semantics of the words themselves. Masking the words removes crucial information, especially for real world online arguments that do not have as clearly fallacious structure as textbook examples. We thus decided to do all of the following experiments with the unmasked arguments and the hypothesis as inputs.

We find that, while the pure fine tuned BERT models using structure aware preprocessing achieve accuracy and F1 scores in the high eighty percent range, using more specialised fine tuning approaches and architectures such as Roberta and deBERTa perform best out of all models, with 89% accuracy and macro F1.

The fact that the models performs well in the same training and testing distribution shows that with a larger dataset these models could indeed be used to effectively detect fallacious online arguments.

4.2 Generalizing to unseen domains

Following Jin et al. (2022), we also tested the models on out of domain data. For that, we trained models with all of the arguments we gathered that were unrelated to climate and tested only on climate related arguments. The test set consists of 1,079 fallacies and 721 valid arguments. We again fine tuned BERT, Roberta, deBERTa and Electra.

	P	R	F1	Acc
BERT	**72**	**67**	**61**	**62**
Electra	71	65	57	58
Roberta	71	65	58	59
deBERTa	70	63	55	57

Table 4: Macro average of model performance in % on the climate test set

The results of this experiment are shown in table 4. All of the metrics drop in comparison to the values reported in the previous experiment, meaning that domain distribution does matter for classification accuracy. This is not unexpected since classifying fallacies heavily relies on knowledge about the background and context of the arguments being classified. Moreover, the writing style in journal articles from which the LogicClimate fallacies come from are very distinct from the Kialo arguments and the fallacies in the Logic corpus, which were authored by anyone interested in taking part rather than trained journalists. These results show the importance of having a dataset which covers as many different subjects and domains as possible.

4.3 Fallacy data comparison

As pointed out above, the fallacies we collected on kialo.com (KIALO) are different from the fallacies in the LOGIC dataset. Therefore, we wanted to see whether this has ramifications in how well the models can train on them. We did this by creating two training sets, the first comprising of the scraped (KIALO) fallacies and scraped counterexamples (KIALOVALID), the second with LOGIC fallacies and KIALOVALID as well (so the training sets only differ in their fallacies).

We then tested the models on the climate test set as in the previous experiment. The results are shown in table 5. We can see that the model trained on LOGIC has an accuracy of 68%, so it performs better than the one trained on KIALO, there the accuracy is 59%, on the same test set. However LOGIC also has a larger number of fallacies (2449 vs. 848) so this was expected.

To have a fair comparison, therefore, we reduced the LOGIC dataset to have roughly the same number of fallacies as KIALO, and the resulting accuracy is in fact 1% higher for Kialo than LOGIC.

This seems to imply that our scraped real-world data is similarly effective in training the classifier as the textbook logical fallacies, which is surprising

	P	R	F1	Acc
Kialo	60	60	59	59
Logic	74	72	68	68
Logic reduced	71	64	56	58

Table 5: Comparison of our fallacies with Logic fallacies when testing on climate data, using macro average

since the real-world examples are much noisier and were also harder to validate for us during data cleaning.

5 Limitations

The main limitation in our work is that the data we have comes from different sources (some examples are contrived and others more realistic), and each source had its own criteria for classification and its own data cleaning process. As we are not linguists nor logicians, our judgement in assuring correctness in the fallacy examples is imperfect.

Furthermore, after we had collected our data and manually cleaned it, it was no longer perfectly balanced.

We have seen in the first experiment that the classifier performs well in the same training and testing distribution. However, to achieve better generalisation results we would need to create a bigger dataset using the same classification criteria, which is very time consuming and hence lay outside of the scope of our project.

Also, since our main focus was on the proof of concept and the data set, there are more combinations of models and preprocessing we did not try due to time constraints, and we did not do hyper parameter optimisation.

6 Future Work

As we previously mentioned, an extension of our work could be to combine our fallacy detection with the fallacy classification from Jin et al. (2022) to get a complete end to end pipeline. This would lend more credence to the models judgement of statements as fallacies because one could understand what type of fallacy exactly the model thinks it sees.

Additionally, the dataset could be changed to include non-arguments. As it is now, the no-fallacy class mostly contains valid arguments, so it is not certain how the classifier would work on sentences containing no logical argument at all. Alternatively, another binary classifier that does argument detection could be appended in the pipeline before the fallacy detection step.

The complete pipeline could then be made available to the public through the use of software or web extensions.

7 Conclusion

We have introduced the task of fallacy detection as a binary classification task. For this, we created a custom dataset which consists of 10,364 statements labeled as either fallacious or non fallacious. These statements were aggregated from multiple sources to create a roughly balanced dataset which can be used to train a classifier on a training subset of this data.

We fine tuned some state of the art language models on our data and achieved an accuracy of 89 percent on the hold out test set.

While there are some performance implications for completely out of distribution data, and there is some way to go to achieve human level accuracy, we would argue that being able to automatically tell in 9 out of 10 cases whether a statement is fallacious could already provide very useful guidance in a user application, provided an uncertainty warning is given.

References

Vibhor Agarwal, Sagar Joglekar, Anthony P. Young, and Nishanth Sastry. 2022. Graphnli: A graph-based natural language inference model for polarity prediction in online debates. In *The ACM Web Conference (TheWebConf)*.

Gioia Boschi, Anthony P. Young, Sagar Joglekar, Chiara Cammarota, and Nishanth Sastry. 2019. Having the last word: Understanding how to sample discussions online. *CoRR*, abs/1906.04148.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing.

Kristina Hristakieva, Stefano Cresci, Giovanni Da San Martino, Mauro Conti, and Preslav Nakov. 2022. The

spread of propaganda by coordinated communities on social media. In *14th ACM Web Science Conference 2022*. ACM.

Zhijing Jin, Abhinav Lalwani, Tejas Vaidhya, Xiaoyu Shen, Yiwen Ding, Zhiheng Lyu, Mrinmaya Sachan, Rada Mihalcea, and Bernhard Schölkopf. 2022. Logical fallacy detection. *arXiv preprint arXiv:2202.13758*.

John Lawrence and Chris Reed. 2020. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. 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*, pages 7871–7880, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2019. Huggingface's transformers: State-of-the-art natural language processing.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3914–3923, Hong Kong, China. Association for Computational Linguistics.