Reproducibility Report for CSE 481N

Griffin Golias, Alex Dundarov, Charles Immendorf, Eric Yeh
Team 4 - Paul G. Allen's Card
{goliagri, alexd02, chazi, yehe}@uw.edu

May 5, 2023

Project Report

Paper: Logical Fallacy Detection [1]

Progress

1. Ran thresholding experiments on the Electra-StructAware to see how the metrics differ (precision, recall, etc). Worked on understanding the code that the author's used to generate these values.
2. Tried running experiments on the forked version of the repo to see if they are closer to the ones in the paper.
3. Sent a followup email to the authors, no response yet.

Next Steps

1. Try running a small number of inferences and see how the metrics are derived. It seems somewhat inconsistent, there are multiple places in the code which calculate precision and recall values and they do so in different ways.
2. Hear back from the authors on feedback for how to get the results closer to theirs
3. try training and testing on models outside of the Electra ones.

1 Introduction

The paper we are trying to replicate runs tests to demonstrate that introducing structure awareness to a model will improve its ability to detect logical fallacies. Additionally, the paper tries to evaluate if the model trained on the main LOGIC dataset can generalize to the LOGICCLIMATE dataset.

2 Scope of Reproducibility

This paper introduces a structure-aware model called Electra-StructAware and claims that this model classifies logical fallacy arguments better than other models. The reasoning is that the structure of an argument is more likely to determine the logical fallacy type than the content words, therefore it would be ideal to mask the content words out to simplify classification. To corroborate the claim, the paper ran experiments using the Electra-StructAware model and compared the results against other zero-shot and fine-tuned models and outperforms the best one, giving an F1 score of 58.77%. Since the other models are no longer available, we will aim to solely reproduce the results of the Electra scores to determine if the claim holds.

3 Methodology

3.1 Model Descriptions

The Electra-StructAware model produces in a structure-aware premise and a structure-aware hypothesis and passes those into a NLI model that will determine if the structure-aware premise matches the structure-aware hypothesis.

The structure-aware premise consists of the original sentence whose fallacy type that is to be predicted. The sentence is modified using coreference resolution to mask out the similar content words. After masking, the sentence is simplified to the basic structural elements that might make it easier to determine the fallacy type.

3.2 Datasets

The dataset we will train and evaluate is a collection of news articles that may or may not contain a variety of logical fallacy types. The links to these articles can be found on the Github repo.

3.3 Implementation

We have utilized the code that is provided on the Github repo. For training and evaluating, we ran the scripts found in codes_for_models/logicedu.py.

3.4 Experimental Setup

We ran the experiments on the NLP machines, which have 2 GPUs. We also set up a virtual Python environment with all the dependencies needed to run the scripts.

The following command was run to evaluate the Electra model:

```
$ python logicedu.py  -t "google/electra-large-discriminator"
    -m "../../saved_models/electra-logic/"
    -ts 1 -ds 1 -nt T -mp base
```

The following command was run to train and evaluate the Electra-StructAware model:

```
$ python logicedu.py -t "google/electra-large-discriminator"
    -m "howey/electra-base-mnli"
    -ts 1 -ds 1 -nt F -mp masked-logical-form -w 12
    -s ../../saved_models/new-struct-aware-model/
```

4 Results

We were successfully able to reproduce the results of the Electra model, getting fairly similar results to the original paper. However, we have found that the results of rerunning the Electra-StructAware model do not reflect the results reported in the original paper, even after retraining the model from scratch. We will need to explore further as to why our results are not matching and how we can adjust our configuration to reproduce successfully.

4.1 Electra

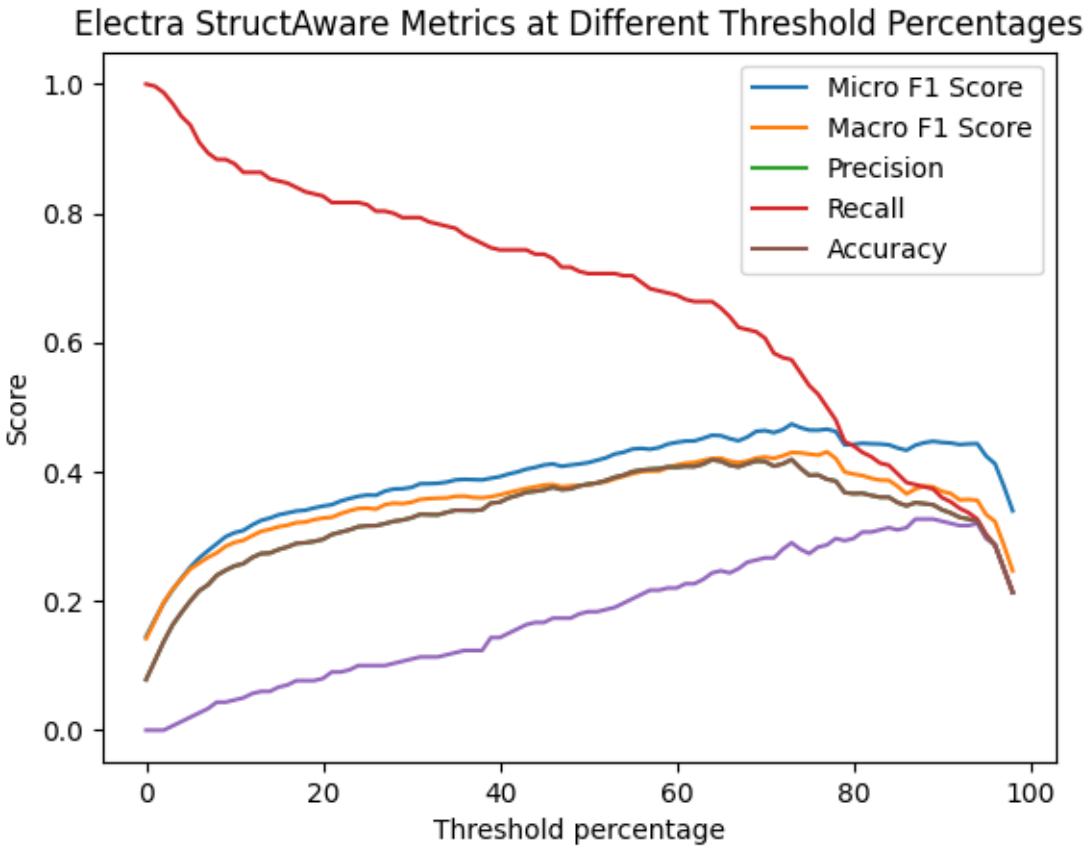
	Reproduced	Original
F_1	50.58	53.31
P	51.23	51.59
R	80.33	72.33
Acc	33.33	35.66

4.2 Electra-StructAware

	Reproduced	Original
F_1	47.95	58.77
P	47.11	55.25
R	74.00	63.67
Acc	29.33	47.67

4.3 Electra-StructAware

Because our model was reaching the desired results, we decided to test out the impact of different thresholds on the accuracy scores to see if changing it would get us closer to the reproduced results. The scores from adjusting the thresholds are reflected in the graph below.



Unfortunately, we were not able to get our accuracy score much higher than 0.4, in comparison to the desired result of over 0.5. We will continue conducting other experiments to see if we can make further progress in reproduction while we wait for the authors to respond.

Communication with Original Authors

We have contacted the lead author of the paper, Zhijing Jin, and the main contributor to the Github repo, Abhinav Lalwani. We have learned that the reproducibility issue on the Github repo has not yet been resolved, and they encourage us to work towards getting it working properly.

Questions to Ask:

1. Explanation of the results, and why some of the accuracies are 0s?
2. More context on how the model can be trained and evaluated, which scripts to run (including what commands to pass in)
3. What further work can be done to build on the results from the paper?

References

[1] Z. Jin, A. Lalwani, T. Vaidhya, X. Shen, Y. Ding, Z. Lyu, M. Sachan, R. Mihalcea, and B. Schoelkopf. Logical fallacy detection. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7180–7198, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. 1