CS 6120 Final Project Written Report

Rui Liu

Abstract

The project will be a WhoIsWho of Semantic Scholar. Semantic Scholar is a research tool for scientific literate. One of the issues with using this website is name disambiguation. This project is aimed to help improve the efficiency of distinguishing scientific literate under authors with the same name.

1 Introduction

In the ever-expanding landscape of academic research, the challenge of name disambiguation has become increasingly pronounced. With the exponential growth of research papers, distinguishing between individuals who share the same or similar names has become a fundamental problem in online academic systems.

To contribute to research, this project is to recurrent the RealTime Name Disambiguation (RND) task of WhoIsWho benchmark. The WhoIsWho benchmark is a substantial contribution to the field, comprising over 1,000,000 papers meticulously curated through an interactive annotation process. This large-scale benchmark is designed to represent the diversity and complexity of real-world scenarios, providing a robust foundation for evaluating and advancing name disambiguation algorithms.

In this project, I dived deep into the source code of WhoIsWho and compared the different transformers and classification methods.

2 Steps

The online academic systems have already constructed a large number of author profiles, so the most urgent problem is to assign newly-arrived papers to the existing author profiles efficiently.

Given a set of unassigned papers and a group of existing author profiles, researchers need to assign unassigned papers to the right author profiles or return NIL (When there are no right author profiles). To this end, the real-time name disambiguation becomes a classification problem.

To train the model, I pre-process the user's name, label every author as different classes and use papers relevant to the author to train, then classify paper to different classes.

- Randomly pick some paper and download more paper with same author as dataset
- Retrieve abstracts of all the paper and convert into vectors as the data
- Separate authors into classes
- Use Logistic and XGB Classifier
- Run test data and calculate accuracy

3 Results

Transformer	Classification	Accuracy
TF-IDF	XGB	0.65
BERT	XGB	0.68
BERT	Logistic	0.85

4 Limitations

Personal devices limit the memory and speed of training large scale data. The original author list size was 388 but I have to reduce it to 20. Otherwise processing the data takes more than 2 hours and the google colab runs out of memory before the model is trained.

5 Conclusions

- Data size has impact on accuracy but it depends.
- BERT yields a better score than TF-IDF in this experiment.
- Logistic Regression gives a high score compared to XGBClassifier in this experiment.

6 Extensions

- Increase the number of compared pair of transformers and classification models
- Increase the calculation ability to process large amount of data
- Conduct more research in data pre-processing

Acknowledgements

https://www.aminer.cn/whoiswho
https://github.com/THUDM/WhoIsWho
https://github.com/kwchurch/JSALT$_Better_Together/$