Cross Domain Claim Analysis

K V Pankaj
Information and
Communication Technology
DA-IICT
Gandhinagar, Gujarat, India
201701068@daiict.ac.in

Prof Prasenjit Majumder
IR LAB, DA-IICT
Gandhinagar, Gujarat, India
p_majumder@daiict.ac.in

ABSTRACT

Argument mining has been a popular area of research in NLP. Arguments can be identified by various components of which claims are of the essence. One definition of claim introduced in the 1950's was that an assertion that needs attention. Claims form the central component of any argument. Claim Identification is an important prerequisite for various applications like fact checking, in legal systems etc. In this experiment, I have performed classification of claims as the primary task across a variety of 6 datasets that each have conceptualized claims differently and contain claims from different domains. Through this experiment we can see the effects of the different ideologies of claims and how they affect the model in predicting the claims.

INTRODUCTION

Argument mining is the computational counterpart of manual argument analysis. Main focal point of the analysis is the identification of a claim in a sentence or a document. There are two types of argument analysis:

1. Discourse level
2. Multi-Document level

To keep the uniformity among the different datasets, in the experiment we have worked on the discourse level. Primarily the task was to predict whether or not the sentence is a claim. Each sentence is classified as a claim if it contains tokens that have been labelled as claim and if no tokens are classified as a claim then the sentence is not a claim.

I have used two deep learning approaches for the text classification task. First approach is using a Convolutional Neural Network (**CNN**) with Glove pre-trained word embeddings and the second method involves the use of Recurrent Neural Network based (**LSTM**) for the sequential data. I have performed extensive In-Domain experiments i.e. training and testing on the same dataset and Cross-Domain experiments which involve training on one dataset and testing on the other. I have also performed an experiment which involves training on all datasets leaving one dataset and testing on it. **F1-score** of the claimed class and the **Macro F1 score** have been calculated for the evaluation and comparison with the results of the original paper. In some cases, the model has beaten the score

of the paper which is discussed in detail in the results section of the report.

KEYWORDS

Deep Learning, Convolutional Neural Network (**CNN**), Recurrent Neural Network (**RNN**), **LSTM**, Text Classification

DATASETS

The experiment involves a dataset from various domains to capture the differences between the notions of claim between them. The **AraucariaDB** corpus or **VG**(Various Genres) corpus contains data from newspaper editorials, parliamentary records and judicial summaries. The annotation scheme structures arguments as trees and distinguishes between claims and premises at the clause level.

The corpus from **Habernal and Gurevych** includes user generated **WD**(Web Discourse) taken from blog posts, user comments annotated according to the Toulmin's definition of a claim i.e. an assertion that needs attention.

The PE(Persuasive Essays) from (**Stab and Gurevych)** includes 402 student essays and has been widely used in the NLP community for argument mining.

Biran and Rambow annotated claims and premises on online comments from blog threads of LiveJournal. In a subsequent paper they have applied their annotation scheme to documents from the wikipedia talk page (**WTP**) and annotated 118 threads.

Peldszus and Stede created a corpus of German microtexts (**MT**) of controlled linguistic and rhetorical complexity. Each document includes a single argument and does not exceed five argument components. In the first annotation study, 26 untrained annotators annotated 23 microtexts in a classroom experiment. . In a subsequent work, the corpus was largely extended by expert annotators. Recently the corpus was translated into english resulting in the first parallel corpus for computational argumentation. The experiments have been carried out using the english version of the texts.

METHODOLOGY

The task here is to take a sentence and classify whether or not it is a claim. Hence we need to perform sentence classification as that is the only way to ensure compatibility between the datasets. The classification in my experiments have been carried using State of the art deep learning models such as **CNN** and **LSTM.**

The problem faced while training the models is the **F1 score** of the claimed class to be zero in many cases due to the imbalance in the dataset.

Corpus	*Reference*	*Genre*	*#Docs*	*#Tokens*	*#Sentences*	*#Claims*
VG	Reed et al. (2008)	various genres	507	60,383	2,842	563 (19.81%)
WD	Habernal and Gurevych (2015)	web discourse	340	84,817	3,899	211 (5.41%)
PE	Stab and Gurevych (2017)	persuasive essays	402	147,271	7,116	2,108 (29.62%)
OC	Biran and Rambow (2011a)	online comments	2,805	125,677	8,946	703 (7.86%)
WTP	Biran and Rambow (2011b)	wiki talk pages	1,985	189,140	9,140	1,138 (12.45%)
MT	Peldszus and Stede (2015)	micro texts	112	8,865	449	112 (24.94%)

Table 1: Overview of the employed corpora.

As we can see above the ratio of the positive classes to the negative classes is at max 30% and is even as low as 5% in case of the **WD** dataset. In such a case direct training of the model on the dataset makes the model always predict the negative class giving 0 **F1 score.**

To overcome the problem of insufficient data, for each model, negative samples have been randomly dropped so that the ratio of the positive and the negative classes remains the same. To cover up the loss of information on dropping the data, 20 such models have been made for each classification task and during the testing of the models the average of the **Macro F1 score** has been calculated as the final score of the problem. This methodology has led to a good consistent result of the F1 score of the claimed class in all the experiments made.

Each model was trained on 15 iterations and 10% of the training data was kept as the unseen test data.

RESULTS

<u>In Domain Experiments:</u>

The **F1 score** of the claimed class and the **Macro F1 score** of each CNN and RNN models have been used for comparison. The paper used CNN with the word2vec model whereas I used CNN with Glove embeddings.

CNN models

The authors of the original paper achieved a Macro F1 score of 0.737 and an F1 score of 0.609 on the claimed class on the **MT** where my model got a Macro F1 score of 0.67 and an F1 score ranging from 0.57 to 0.78 clearing beating the score of the paper.

On the **OC** dataset the results of the paper was an Macro F1 score of 0.582 and an F1 score of 23.7 on the claimed class. The model which I used gave a Macro F1 score of 0.512 and an F1 score ranging from 0.40 to 0.57 which is far higher than the paper's model.

On the **PE** dataset the results of the paper was an Macro F1 score of 0.74 and an F1 score of 0.617 on the claimed class. The model which I used gave a Macro F1 score of 0.67 and an F1 score ranging from 0.65 to 0.72 which is better than the paper's model.

On the **VG** dataset the results of the paper was an Macro F1 score of 0.63 and an F1 score of 0.33 on the claimed class. The model which I used gave a Macro F1 score of 0.54 and an F1 score ranging from 0.40 to 0.59 which is better than the paper's model.

On the **WD** dataset the results of the paper was an Macro F1 score of 0.62 and an F1 score of 0.28 on the claimed class. The model which I used gave a Macro F1 score of 0.64 and an F1 score ranging from 0.56 to 0.79 which is far better than the paper's model.

On the **WTP** dataset the results of the paper was an Macro F1 score of 0.65 and an F1 score of 0.48 on the claimed class. The model which I used gave a Macro F1 score of 0.53 and an F1 score ranging from 0.49 to 0.61 which is far better than the paper's model.

LSTM models

On the **MT** dataset the results of the paper was an Macro F1 score of 0.57 and an F1 score of 0.24 on the claimed class. The model which I used gave a Macro F1 score of 0.7 and an F1 score ranging from 0.57 to 0.84 which is far better than the paper's model.

On the **OC** dataset the results of the paper was an Macro F1 score of 0.58 and an F1 score of 0.22 on the claimed class. The model which I used gave a Macro F1 score of 0.54 and an F1 score ranging from 0.37 to 0.58 which is better than the paper's model.

On the **PE** dataset the results of the paper was an Macro F1 score of 0.71 and an F1 score of 0.6 on the claimed class. The model which I used gave a Macro F1 score of 0.66 and an F1 score ranging from 0.57 to 0.67.

On the **VG** dataset the results of the paper was an Macro F1 score of 0.61 and an F1 score of 0.4 on the claimed class. The model which I used gave a Macro F1 score of 0.65 and an F1 score ranging from 0.56 to 0.67 which is better than the paper's results.

On the **WD** dataset the results of the paper was an Macro F1 score of 0.61 and an F1 score of 0.25 on the claimed class. The model which I used gave a Macro F1 score of 0.67 and an F1 score ranging from 0.53 to 0.69 which is far better than the paper's results.

On the **WTP** dataset the results of the paper was an Macro F1 score of 0.58 and an F1 score of 0.28 on the claimed class. The model which I used gave a Macro F1 score of 0.66 and an F1 score ranging from 0.52 to 0.66 which is far better than the paper's results.

Cross Domain Results - 1

One of the experiments carried out was training on a dataset and testing on the other. The CNN model used by the paper was Kim's CNN with random embeddings but the model used by me is CNN with Glove embeddings.

CNN models

Training on **MT** dataset and testing on the **OC** dataset. Macro F1 score of the paper was 0.51 whereas the model I used gave a score of 0.43.

Training on **PE** dataset and testing on the **VG** dataset. Macro F1 score of the paper was 0.57 whereas the model I used gave a score of 0.53.

Training on **WD** dataset and testing on the **WTP** dataset. Macro F1 score of the paper was 0.46 whereas the model I used gave a score of 0.50.

LSTM models

LSTM model results have not been shown by the paper. Below are the results of my model on the datasets.

Training on **MT** dataset and testing on the **OC** dataset. Macro F1 score of the model I used gave a score of 0.51.

Training on **PE** dataset and testing on the **VG** dataset. Macro F1 score of the model I used gave a score of 0.54

Training on **WD** dataset and testing on the **WTP** dataset. Macro F1 score of the model I used gave a score of 0.51.

Cross Domain Results -2

Another set of experiments performed was that of training on all datasets leaving one and testing on it. In this scenario, the neural network systems seem to benefit from the increased amount of training data and thus gave the best results

CNN models

The model used by the paper was CNN with random embeddings but the model that I have used is CNN with Glove embeddings.

Leaving the **MT** dataset. The Macro F1 score of the paper is 0.62 whereas the results of my model gave a score of 0.49.

Leaving the **OC** dataset. The Macro F1 score of the paper is 0.57 whereas the results of my model gave a score of 0.50.

Leaving the **PE** dataset. The Macro F1 score of the paper is 0.59 whereas the results of my model gave a score of 0.51

Leaving the **VG** dataset. The Macro F1 score of the paper is 0.58 whereas the results of my model gave a score of 0.503.

Leaving the **WD** dataset. The Macro F1 score of the paper is 0.54 whereas the results of my model gave a score of 0.42.

Leaving the **WTP** dataset. The Macro F1 score of the paper is 0.56 whereas the results of my model gave a score of 0.47.

LSTM models

The paper had not produced the results using LSTM models. Below are the results of the model that I have used.

Leaving the **MT** dataset. The Macro F1 score of the model is 0.50.

Leaving the **OC** dataset. The Macro F1 score of the model is 0.52

Leaving the **PE** dataset. The Macro F1 score of the model is 0.54.

Leaving the **VG** dataset. The Macro F1 score of the model is 0.53.

Leaving the **WD** dataset. The Macro F1 score of the model is 0.40.
Leaving the **WTP** dataset. The Macro F1 score of the model is 0.51.

The CNN and the LSTM models have outperformed all the baseline model's results and sometimes produced a better result than the deep learning models of the paper. Although the Macro F1 score of the model I used and the one used in the paper are very comparable, there is a big disparity in the F1 score of the claimed class of the model I used and the one in the paper.

Also the LSTM models have even beaten the CNN models in recognising the sequential data and mining the claims.

ACKNOWLEDGEMENT
This work would not have been possible without the help of Prof Prasenjit Majumder and our teaching assistants at the IR LAB at DA-IICT. I thank my professor and the TA's for all their support through this journey.

REFERENCES

Paper

1. Daxenberger, Johannes & Eger, Steffen & Habernal, Ivan & Stab, Christian & Gurevych, Iryna. (2017). What is the Essence of a Claim? Cross-Domain Claim Identification.
2. Nguyen, Son & Nguyen, Le-Minh & Tojo, Satoshi & Satoh, Ken & Shimazu, Akira. (2018). Recurrent neural network-based models for recognizing requisite and effectuation parts in legal texts. Artificial Intelligence and Law. 1-31. 10.1007/s10506-018-9225-1.