Model explainability - in Context of Argument Mining

Student Name: Anunay Yadav
Roll Number: 2018021

BTP report submitted in partial fulfillment of the requirements
for the Degree of B.Tech. in Computer Science & Engineering

on 12 Dec 2021

BTP Track: Research

BTP Advisor
Tanmoy Chakraborty
BTP Evaluator
Md. Shad Akhtar
Ganesh Bagler

Indraprastha Institute of Information Technology
New Delhi

Student's Declaration

I hereby declare that the work presented in the report entitled **Model explainability - in Context of Argument Mining** submitted by me for the partial fulfilment of the requirements for the degree of *Bachelor of Technology* in *Computer Science & Engineering* at Indraprastha Institute of Information Technology, Delhi, is an authentic record of my work carried out under the guidance of **Tanmoy Chakkraborty**. Due acknowledgements have been given in the report to all material used. This work has not been submitted anywhere else for the reward of any other degree.

.... **Place & Date: IIITD, 12 Dec, 2021**
Anunay Yadav

Certificate

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

.... **Place & Date: IIITD, 12 Dec, 2021**
Tanmoy Chakraborty

Abstract

Argument mining is a rising research area in natural language processing, the goal of which is to extract argumentative structures from natural language texts. Such components contain a lot of information not only limited to objective questions such as finding the location, etc., but can also answer many subjective questions as to why someone holds this opinion. Argument mining has already been applied in social media platforms, legal, and newspapers as a qualitative assessment tool, providing a powerful tool for analysis to analysts without prior knowledge of the domain.

Being such a complex task, little research is done in explaining the state-of-the-art models in this domain. In this project, we are trying to analyze the workings of these models as to why they behave in this way and verify it. We expect to give a combined algorithm that does the above and presents it in an explainable and human-comprehensible format so that users without any prior knowledge can understand the model's inner workings and verify it according to their respective tasks.

Keywords: Argument Mining, NLP, LIME, Explainable AI

Acknowledgments

We would like to thank our mentor Tanmoy Chakraborty (Assistant Professor, IIIT Delhi), for giving me this opportunity to take up the project. We thank him for his constant guidance, immense patience, and encouragement. We would also like to thank Subhabrata Dutta (PhD student, IIITD) for his continuous guidance, motivation, and parallel collaboration.

Work Distribution

Anunay has done all the work mentioned in this report in the Monsoon semester of 2021. The first part of the semester was used for codebase setup and going through many papers in the field to get the current research and understanding of argument mining and model explainability. The second part involves running extensive experiments to analyze the models' working and drawing insights from them.

Contents

Chapter 1

Introduction

Due to such advanced developments on the internet, now the internet serves as a primary media for arguments and debates. But even after many developments in data science, many existing approaches have struggled to analyze arguments and debates. Extracting the complex structure and justifications between the claims has always been the domain of argumentation theory. Argument mining tries to structure these texts by extracting complex structures and dependencies between components. Manual extraction [7] is not possible because of such an increase in data; it is time-consuming and requires skilled individuals to perform the task. Hence, the need to automatically extract the properties of natural language texts is justified. Argument mining has many applications; researchers in many different domains use the extracted properties to analyze data and perform further research. For example, extracting important information from legal documents can greatly help a lawyer; similarly, for a quant researcher or news article writer, similar extractions can help a lot.

Recent developments in argument mining solve the tasks, but how can one trust these models. Do they work as expected? Can we deploy and use them in critical situations? Can we trust their predictions? These questions are equally important as solving the problem, but little research is done in this domain. Lime [6] was one of the major developments in Model explainability though it tries to explain any classifier model (considers it a black box) hence losing the local context of a specific domain.

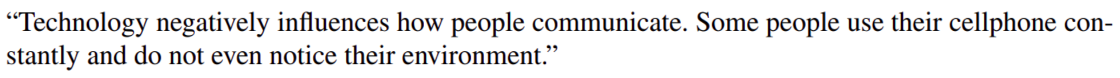
To explain models specifically in argument mining, we need to check many things such as attention to local, global, and how the model behaves with attacks that augment the data etc. We have designed many experiments to analyze such behavior and drawn insights about state-of-the-art NLP models from them.

Chapter 2

Literature Review and Background

2.1 Argument Mining

As stated in [5], Argument mining is an intelligent task where it tries to parse the structure information from the natural language texts. Automatically extracting this information not only makes the problem physically feasible but also makes it possible use these advancements without user involvement. Arguments are a set of phrases ”premise” and ”claim” when together clubbed count as an argument. Argument mining tries to segment out these arguments and find the relation between them, therefore extracting the structural dependencies between each component. Recognizing ”premise - claim” components have been considered tough for both humans and machines; they are often identified by discourse markers ”because,” etc. but these discourse markers have ambiguous meanings, and the claim and premise components can be far apart from each other and finding a relation between any of claim component to any of premise components is difficult, thus these relations from a graph like structure. People manually solve these tasks on the basis of their common sense, world sense, or domain knowledge; this is a major problem for argument mining models as they are not able to understand common world sense and domain knowledge correctly to identify components. For example,



> “Technology negatively influences how people communicate. Some people use their cellphone constantly and do not even notice their environment.”

Here the second statement is a supporting premise for the claim in the first statement, which can be identified in many ways, but the machine has to identify this relationship by knowing the fact that telephones are used as a form of communication. Argument mining models have major unidentified drawbacks. These could be solved if there was a way to explain their behavior or a set of empirical analyses that could help the user understand where our model is focusing.

2.2 Model Explain-ability technique - Lime

Lime is an important advancement in Model explainability; It tries to explain any machine learning model in a format that any person can understand without any prior knowledge. Machine learning models have been core for many recent developments, and most people don't know whether to trust a model or a prediction. There has not yet been a common way of evaluating or explaining every machine learning model irrespective of context. Lime proposes an algorithm that helps users differentiate between a trustworthy model/prediction and an untrustworthy one. Trusting our Model is an important part of the process as in medical science or terrorism detection, we cannot blindly agree with the predictions made by the Model.

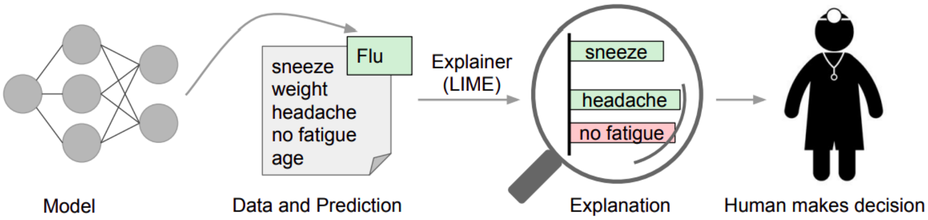
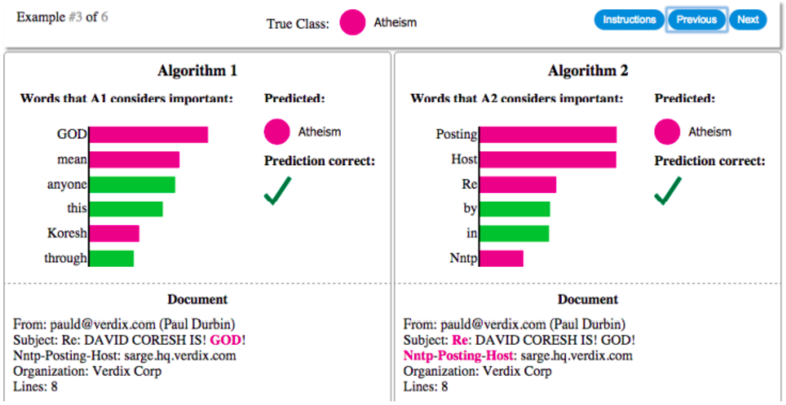


Figure 2: Explaining individual predictions. A model predicts that a patient has the flu, and LIME highlights the symptoms in the patient's history that led to the prediction. Sneeze and headache are portrayed as contributing to the "flu" prediction, while "no fatigue" is evidence against it. With these, a doctor can make an informed decision about whether to trust the model's prediction.

In Figure 2, Lime identifies the features that the Model thought was important in the classification of FLU, and because these explanations are in a human-readable format, human without any prior knowledge can verify the working of a model. As in Figure 2, Sneeze and headache are common symptoms of flu and hold high weightage to the prediction of Flu, but No fatigue is not a symptom and hence holds negative weightage towards the prediction. The good thing about the explanations provided is that a human who is actually making the decision can apply his prior knowledge of the domain without any knowledge on how the Model works and inner working of Lime; he can make correct decisions on the basis of explanations and explanation only.

Lime is not limited to the Medical domain, it can explain models from NLP as well as CV.

Use of Lime in NLP: Lime for NLP is quite tricky since Lime depends on data generations and finds boundaries on the basis of that. In NLP, to clearly understand the working of a model, Lime should be able to generate a variety of language data. Lime uses data augmentation techniques and generates a variety of meaningful texts to assess the knowledge of models. One example of Lime working in NLP is the following.



Example #3 of 6

True Class: ● Atheism

Algorithm 1

Words that A1 considers important:

Predicted: ● Atheism

Prediction correct: ✓

Document

From: pauld@verdix.com (Paul Durbin)
Subject: Re: DAVID CORESH IS! **GOD**!
Nntp-Posting-Host: sarge.hq.verdix.com
Organization: Verdix Corp
Lines: 8

Algorithm 2

Words that A2 considers important:

Predicted: ● Atheism

Prediction correct: ✓

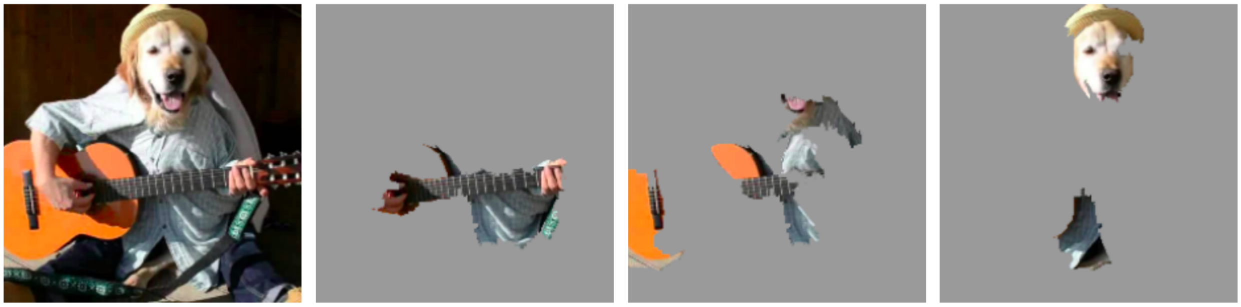
Document

From: pauld@verdix.com (Paul Durbin)
Subject: **Re**: DAVID CORESH IS! GOD!
Nntp-Posting-Host: sarge.hq.verdix.com
Organization: Verdix Corp
Lines: 8

Figure 3: Explaining individual predictions of competing classifiers trying to determine if a document is about "Christianity" or "Atheism". The bar chart represents the importance given to the most relevant words, also highlighted in the text. Color indicates which class the word contributes to (green for "Christianity", magenta for "Atheism").

In Figure 3, given full context as a text email without any feature engineering, Lime was able to extract tokens that hold high weightage to "Atheism." Another thing to consider is how Lime is able to highlight the tokens that have had a major impact on the decision.

Use of Lime in CV: Lime can highlight the pixels that hold high weightage to the predictions given an image and classifier model that classifies this image. for example-

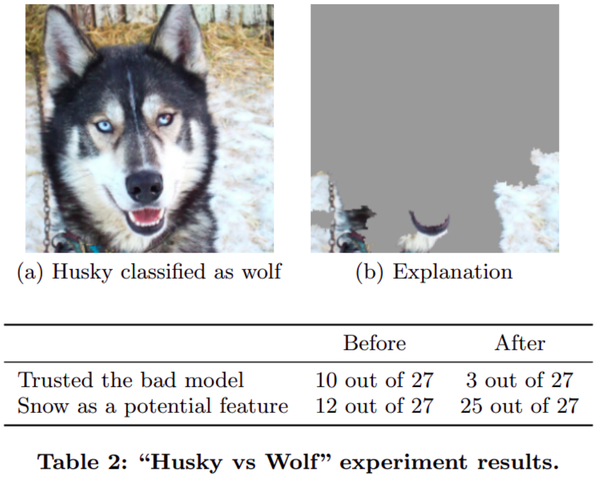


(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

Figure 4: Explaining an image classification prediction made by Google's Inception neural network. The top 3 classes predicted are "Electric Guitar" (p = 0.32), "Acoustic guitar" (p = 0.24) and "Labrador" (p = 0.21)

In figure 4: Considering the Original Image, the Electric guitar classification focused exactly on guitar strings and similarly for labradors. An important thing to note is that Anyone with Little knowledge can now understand the workings of the Model; they can verify if their models are actually focusing on important features or not.

Lime Even was able to identify mistakes in many SOTA models, one example of which is in Figure 5, as the Model classified an image as Wolf and not as Husky based only on the background and not on the properties of the dog. So for Model, snow played a major role in classification, which is not correct. Many people were still using this Model. They had no idea if the Model was working correctly or not; after the survey, because of Lime, the number of people who trusted the Model reduced drastically.



(a) Husky classified as wolf

(b) Explanation

	Before	After
Trusted the bad model	10 out of 27	3 out of 27
Snow as a potential feature	12 out of 27	25 out of 27

Table 2: "Husky vs Wolf" experiment results.

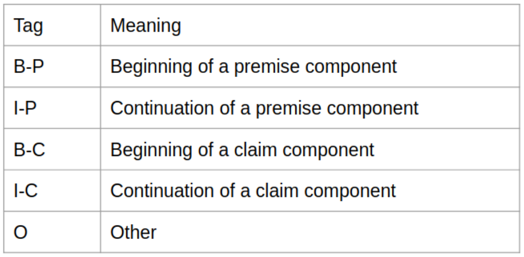
Figure 5: Raw data and explanation of a bad model's prediction in the "Husky vs Wolf" task and Table after the survey.

Chapter 3

Experiment Setting

3.1 Dataset

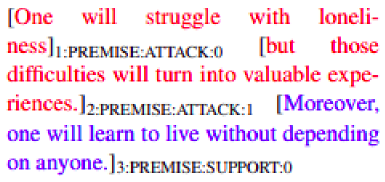
+ We picked two popular NLP Argument Mining datasets (one big and one small) to perform our experiments on. We picked the following models due to the easy availability of multiple classification models present for these datasets, So we can analyze many models and perform experiments on many SOTA models with ease. Tags and Their meaning is given below.



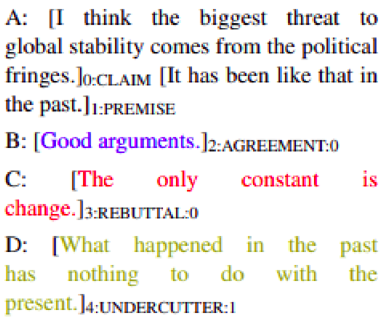
Tag	Meaning
B-P	Beginning of a premise component
I-P	Continuation of a premise component
B-C	Beginning of a claim component
I-C	Continuation of a claim component
O	Other

3.1.1 Change my views subreddit

Change my views subreddit [2] is very popular for discussions and their threads contain a long chain of arguments that is the discussion between many people on the subreddit. This dataset used the CMV subreddit for dataset and segmented threads in CMV modes in different components such as claim and premise and in sub relation types which explain the relationship between these components. Also, the dataset contains interrelations between each component, constituting a complex graph. An example of an annotated thread is given below.



[One will struggle with loneliness]1:PREMISE:ATTACK:0 [but those difficulties will turn into valuable experiences.]2:PREMISE:ATTACK:1 [Moreover, one will learn to live without depending on anyone.]3:PREMISE:SUPPORT:0



A: [I think the biggest threat to global stability comes from the political fringes.]0:CLAIM [It has been like that in the past.]1:PREMISE

B: [Good arguments.]2:AGREEMENT:0

C: [The only constant is change.]3:REBUTTAL:0

D: [What happened in the past has nothing to do with the present.]4:UNDERCUTTER:1

3.1.2 Dr Inventor

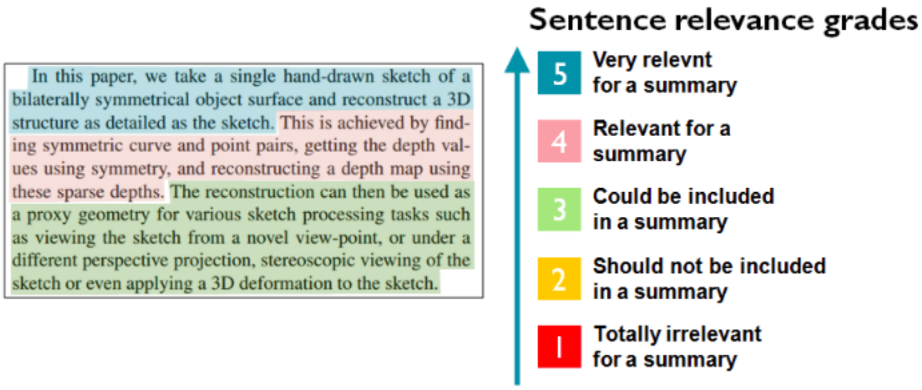
Dr Inventor is a combined corpus of 40 computer Graphics papers, selected by the domain experts. Each paper of the corpus has been annotated by three annotators. All three provided the following Layers of annotations, each representing the core aspect of the paper.

Scientific discourse: each sentence has been associated to a specific scientific discourse category (Background, Approach, Chalenge, Future Work, etc.).

Subjective statements and novelty: each sentence has been characterized with respect to advantages, disadvantages and novel aspects presented.

Citation purpose: to each citation has been associated a purpose specifying the reason why the authors of the paper cited the specific piece of research.

Summary relevance of sentences and hand written summaries: each sentence of the paper has been characterized by an integer score ranging from 1 to 5, to point out the relevance of the same sentence for its inclusion in the summary of the paper. Sentences rated as 5 are the most relevant ones to summarize a paper. For each paper three hand-written summaries (max 250 words) are provided.



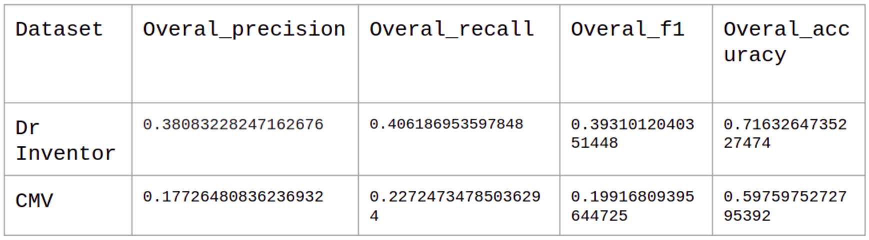
In this paper, we take a single hand-drawn sketch of a bilaterally symmetrical object surface and reconstruct a 3D structure as detailed as the sketch. This is achieved by finding symmetric curve and point pairs, getting the depth values using symmetry, and reconstructing a depth map using these sparse depths. The reconstruction can then be used as a proxy geometry for various sketch processing tasks such as viewing the sketch from a novel view-point, or under a different perspective projection, stereoscopic viewing of the sketch or even applying a 3D deformation to the sketch.

Sentence relevance grades
5 Very relevnt for a summary
4 Relevant for a summary
3 Could be included in a summary
2 Should not be included in a summary
1 Totally irrelevant for a summary

Chapter 4

Experiments and Analysis

We have used Models such as Bert [4], LongFormer [1] etc and performed All the experiments stated below on these models. Models were trained for 35 epochs with 50-50 or 80-20 train-test split. Models are downloaded from Huggingface and custom coded, using Pytorch and Tensorflow framework.



Dataset	Overal_precision	Overal_recall	Overal_f1	Overal_accuracy
Dr Inventor	0.38083228247162676	0.406186953597848	0.3931012040351448	0.7163264735227474
CMV	0.17726480836236932	0.22724734785036294	0.19916809395644725	0.5975975272795392

Figure 6: Results of the model - bert on these datasets. task - Argument Mining.

4.1 Attention Weights Analysis

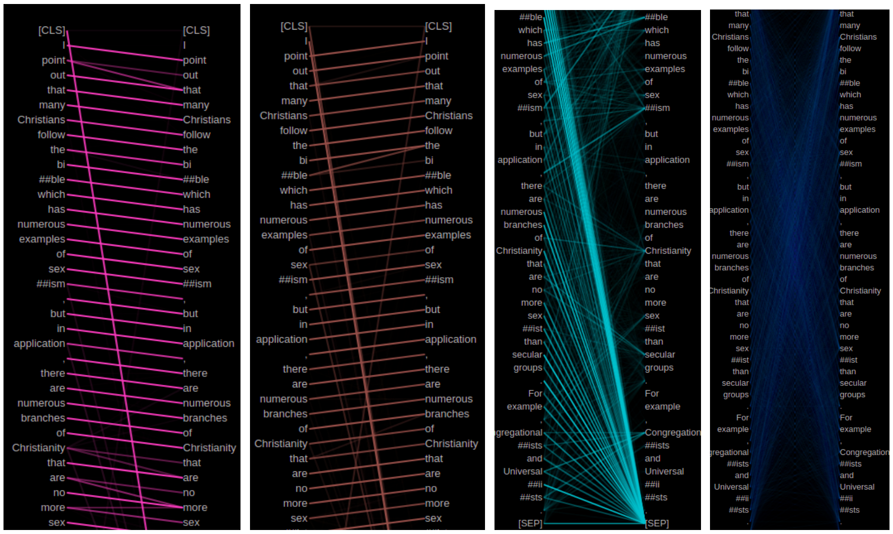


Figure 7: Attention weights - Leftmost focusing on next consecutive token, middle focusing just previous, Second rightmost focusing on the [SEP] token and Rightmost Focusing on broadly whole context.

Bert's attention [3] weights follow many patterns, but mainly these patterns are related to language structure and not to the context. As can be seen from Figure 7, each head focuses on a different aspect of the task, some focusing on the next consecutive token, some on the Separator token added and end of sentences, and global context. Mainly the focus goes to periods and separator tokens which can be because they contain the local context of the sentence. Other than Separator and Period, many language structures are seen, such as the high attention weights of noun modifiers and nouns, similarly for Prepositions and their objects. Attention weights are of use, but we could not draw any task-based conclusions from them since they only exhibit language-based attention patterns. Some attention heads followed a random pattern, which could be because of task-specific reasons.

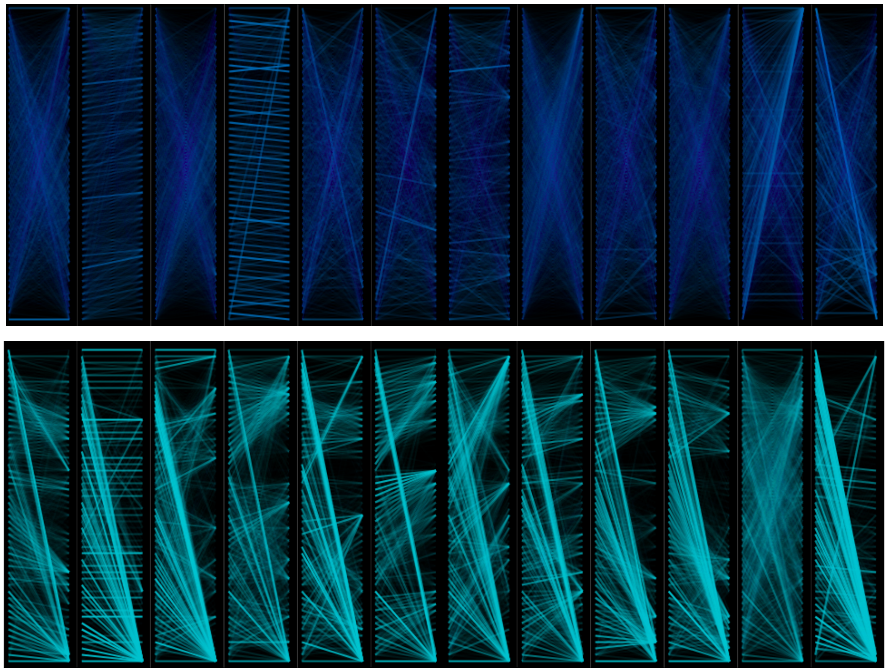


Figure 8: Attention weights - Top one corresponds to layer 0, bottom one - layer 9

Attention heads closer to the input were more random or followed a different pattern than focusing on the SEP token; on the other hand, attention heads 9-11 were mostly focused on SEP token, which i think could be because the upper layer only solidifies claims made from lower layers hence sometimes they do not play a big role in deciding the class.

4.2 Using Lime to Analyse Models

We tried using lime, but we were restricted to two types of experiments because it was not computationally feasible. One gives the global picture of the models working on the dataset, and the second gives the local picture of the models working.

Global Picture: We analysed only Beginning of Claim and premise tags as they were less in number and feasible computationally. still it took around 10-15 days to run the experiment.

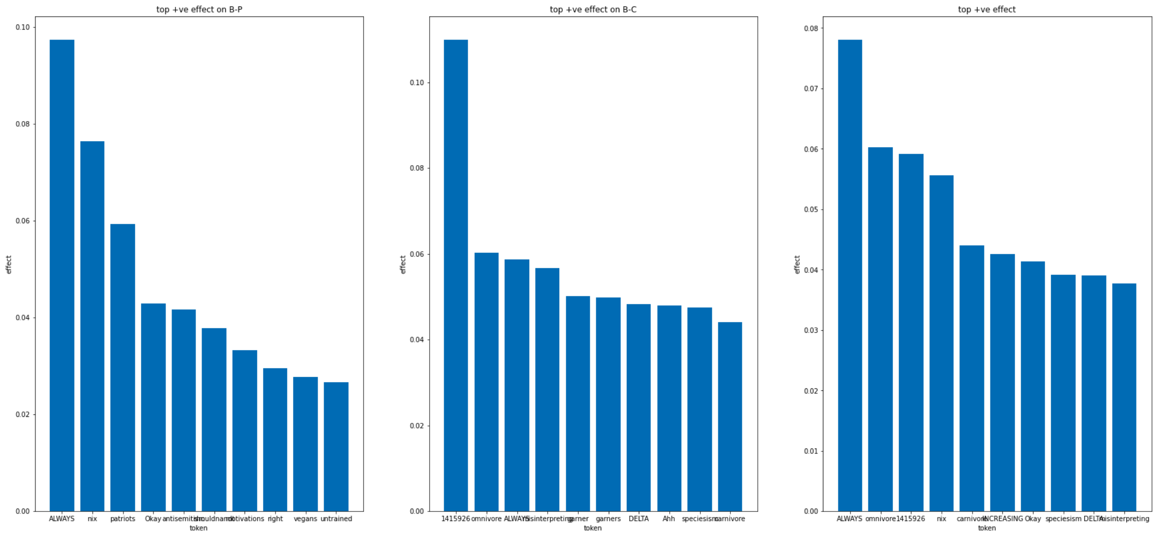


Figure 9: Lime weights class wise, positive Effects of tokens on the particular class prediction.

In Figure 9 it can be seen that the top positive effect on the beginning of the premise were nouns, adjectives and adverbs. As tokens like "ALWAYS", "partriots," and Vegan had a major impact positive impact on the prediction of the start of a premise. Similarly, for the beginning of claim tagging, the most positive impact was from nouns, numbers, and "Delta." Delta is a keyword used to state that a person's opinion changed after the discussion, which actually justifies its existence in the top positive effects list. Overall, the beginning of either claim or premise was majorly affected by Nouns and Adverbs, etc.

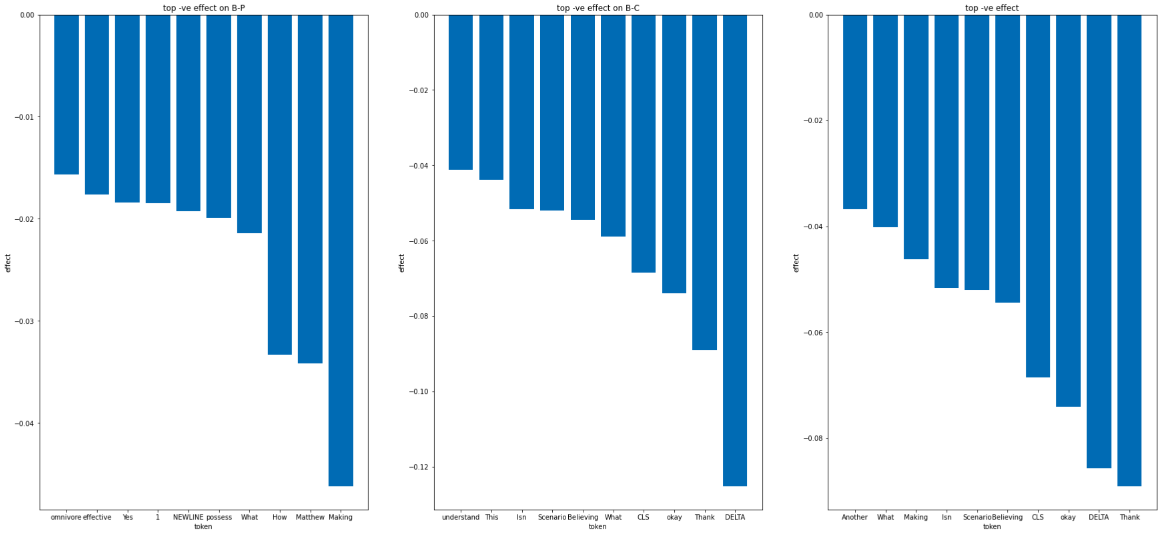
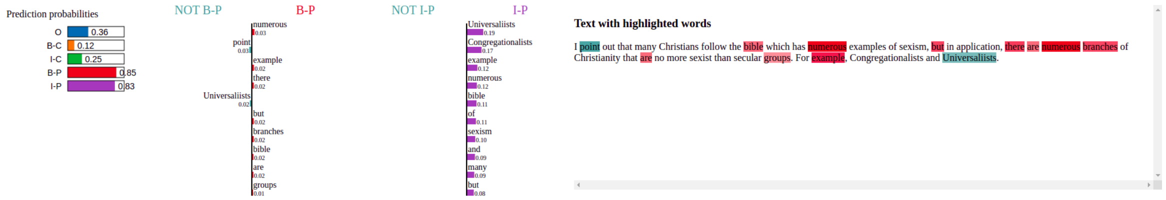


Figure 10: Lime weights class wise, negative Effects of tokens on the particular class prediction.

Tokens which had a most negative effect were pronoun, or a start of the question, unique names and Yes/No answer As Mostly claim and premise do not start with or near question and have ”Thank” in them, but still, some common words are seen in both top positive and negative. Most of the tokens found in the Top negative effect list give a slight indication of agreement such as Okay or a question as what, who, etc.

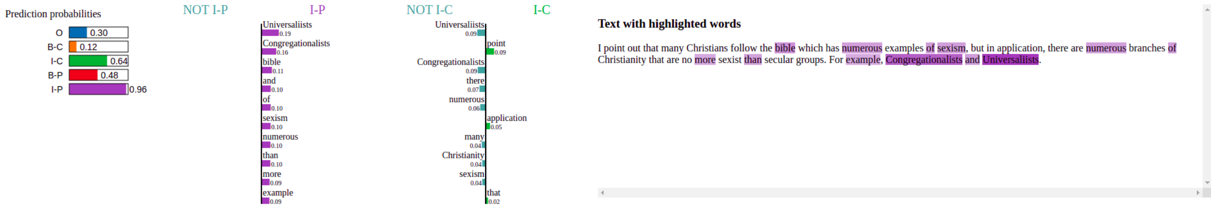
With these experiments, we were slightly able to understand what causes the start of a claim or premise component and what the Model focuses on when deciding the start of a component. Next, We try to analyze the local context by running one example thoroughly and analyzing its predictions.

Local Picture: Each label firstly is affected by the sentences that are very close to it and then the important keywords from other sentences, but mostly Classifications were affected by nouns and verbs, for each token, it was seen that every noun and verb was effecting its decision only the weight changes in the direction, we slide its position.



Text with highlighted words

I point out that many Christians follow the bible which has numerous examples of sexism, but in application, there are numerous branches of Christianity that are no more sexist than secular groups. For example, Congregationalists and Universaliists.

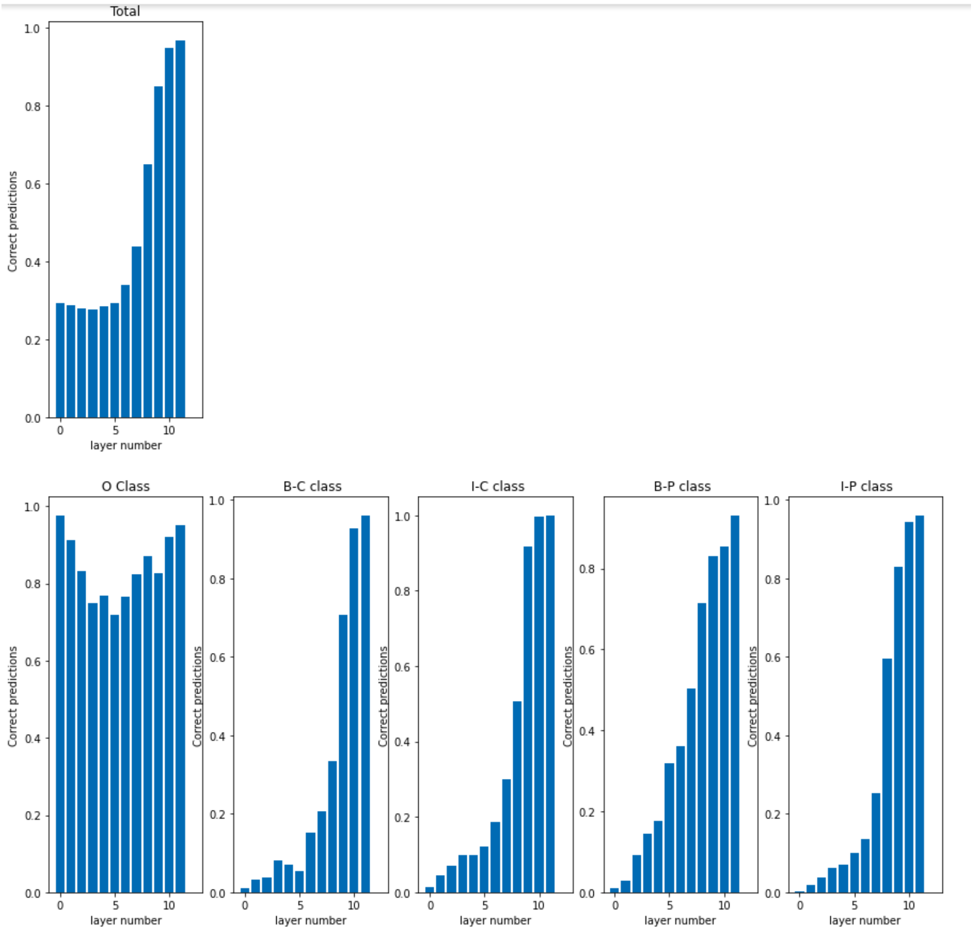


Text with highlighted words

I point out that many Christians follow the bible which has numerous examples of sexism, but in application, there are numerous branches of Christianity that are no more sexist than secular groups. For example, Congregationalists and Universaliists.

Highlighted tokens are the tokens that effect the class the most, Noun and verbs effects prediction the most.

4.3 Layer Wise effect of prediction



Layer Wise prediction accuracy for each class.

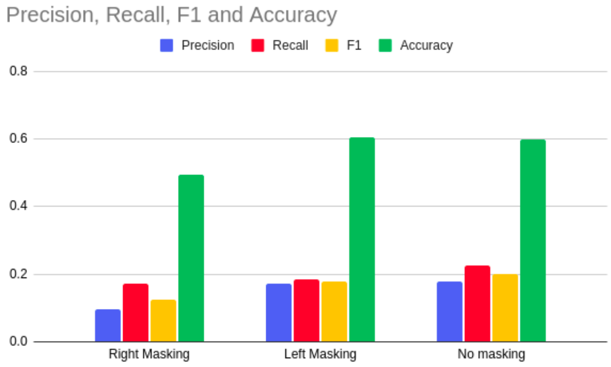
This experiment was to analyze whether the predictions made from the lower layers change as upper layers work on them, As seen from the figure, Actually claims made by lower levels are solidified by upper layers so we can say that lower layers play an important role in the start of the prediction and upper layers try to justify those claims as much as possible. O class does not have a fix distribution and is not the priority of learning hence shows an random nature by upper layers, Only lower layers are enough for the classification of O class.

4.4 Masking Techniques

We wanted to analyze the how components are affected by other components and hence started with masking strategies, Some classes were more effected by masking and some even benefited from the little context they got.

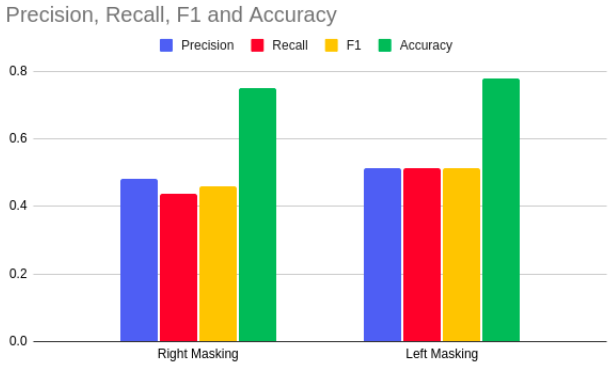
4.4.1 Masking - Right, Left and Nothing

Masking Algorithm: for a particular text, our masking algorithm would mask the left of each component or right of each component till the start or the end respectively on the basis of flag.



Masking strategies results on CMV modes.

Left masking had little to no effect on Precision, F1, and Accuracy, but right masking completely destroyed the working. This proves that for a model identifying the right ends of a component is easier compared to the left end, as with left masking, we give the model the left end of the component and similarly for right and since left masking performed better than left implies that finding right end is easier than left end of a component.



Masking strategies results on Dr Inventor.

Above figure supports our claim on a big dataset, insights from this experiment is that for segmentation model in argument mining it is easier to identify the right end of a component than left end.

4.4.2 Masking strategies in Prompt based Relation Prediction

We wanted to analyze the effect of masking out intercomponent on models, we were not able to analyze the same within the previous as there were many components involved in tagging, but with prompt-based relation prediction, since there are only two components and context, we can now analyze how masking out different components affect the classification.

In prompt-based relation prediction, a sentence is formed of the manner ”Whole- context” + ”User I said” + ”Component 1” + ”Mask tokens” + ”User J said” + ”Component 2”, in this way, mask tokens can be filled as ”supports” or ”agrees” etc. so basically there are three parts, whole context, component 1 and Component 2. ”Whole-context-” contains the discussion in which user i and j are part of, implying that component 1 and 2 already exist in Whole-context-

Strategies used are-

- Masking out the whole context.
- Masking out left of Component 1 in Whole context.
- Masking out between component 1 and 2.
- Masking out right of Component 2 in Whole context.
- Masking out each POS tag in the Whole context.

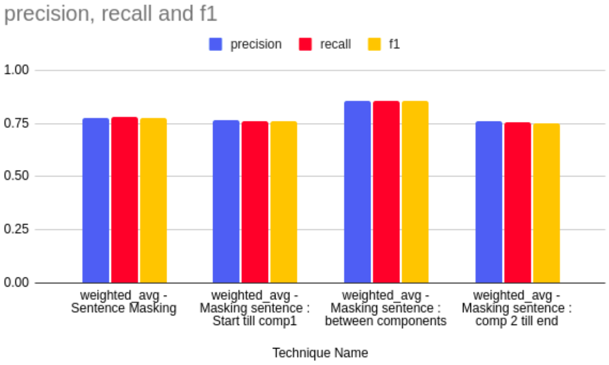
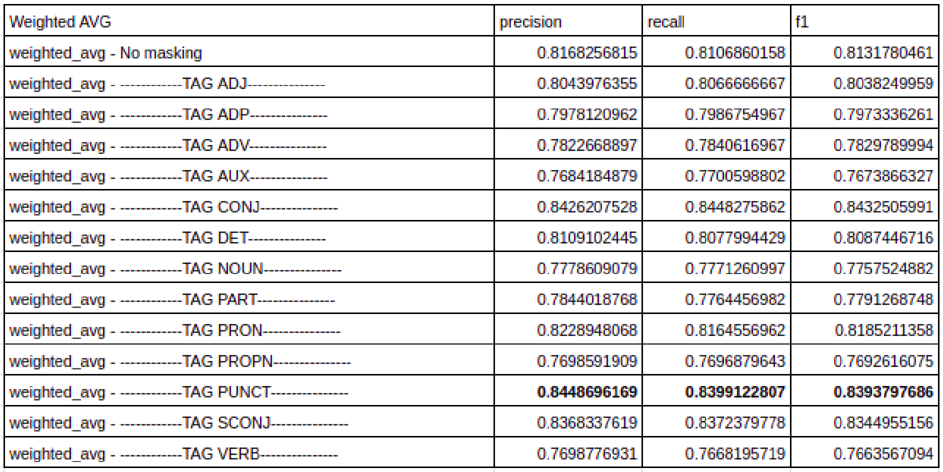


Figure 11: Masking strategies results on Dr Inventor.

Here Sentence is the Whole-context-, masking out the whole context results in drop but does not affect majorly. Masking between the components has had the least effect on the model, which explains that context between the components does not play a big role in the classification of relation. Similarly, the beginning of the context and end of the context both are important for the task.



Weighted AVG	precision	recall	f1
weighted_avg - No masking	0.8168256815	0.8106860158	0.8131780461
weighted_avg - ------------TAG ADJ---------------	0.8043976355	0.8066666667	0.8038249959
weighted_avg - ------------TAG ADP---------------	0.7978120962	0.7986754967	0.7973336261
weighted_avg - ------------TAG ADV---------------	0.7822668897	0.7840616967	0.7829789994
weighted_avg - ------------TAG AUX---------------	0.7684184879	0.7700598802	0.7673866327
weighted_avg - ------------TAG CONJ---------------	0.8426207528	0.8448275862	0.8432505991
weighted_avg - ------------TAG DET---------------	0.8109102445	0.8077994429	0.8087446716
weighted_avg - ------------TAG NOUN---------------	0.7778609079	0.7771260997	0.7757524882
weighted_avg - ------------TAG PART---------------	0.7844018768	0.7764456982	0.7791268748
weighted_avg - ------------TAG PRON---------------	0.8228948068	0.8164556962	0.8185211358
weighted_avg - ------------TAG PROPN---------------	0.7698591909	0.7696879643	0.7692616075
weighted_avg - ------------TAG PUNCT---------------	**0.8448696169**	**0.8399122807**	**0.8393797686**
weighted_avg - ------------TAG SCONJ---------------	0.8368337619	0.8372379778	0.8344955156
weighted_avg - ------------TAG VERB---------------	0.7698776931	0.7668195719	0.7663567094

Figure 12: Masking Part of speech from the whole context.

Masking out Punctuation's, Numbers, Pronouns have little effect on the model even improve the performance of these models. Most important tags were Verb, PropN, Participle, Auxillary, Noun etc, which actually verifies our experiments above as all the experiments show that noun adverb plays an important role in language models.

Chapter 5

Future Work

The following work still need to be done.

1. **Experiments with Text attack** Text attack is a python module that performs vulnerability tests on the models specific to the NLP domain; it does so by augmenting language data in many ways, such as changing the spelling or replacing it with a synonym. All in all, text attack is a great module to test models, and we should try to find vulnerabilities in the models related to using this.

2. **Propose an algorithm that explains it all:** The project aims to give an algorithm like LIME specific to Argument mining, which could easily explain the workings of Argument mining relations tasks even to a person without any prior knowledge of the domain. hence we should try to research more in this domain.

3. **Find weakness in the model:** After so many experiments, we are able to understand the workings of the model, but we still are not able to find exact weaknesses that we can tackle in our proposed model; hence finding such weaknesses is the main goal to move forward.

Bibliography

[1] Beltagy, I., Peters, M. E., and Cohan, A. Longformer: The long-document transformer, 2020.

[2] Chakrabarty, T., Hidey, C., Muresan, S., McKeown, K., and Hwang, A. AMPERSAND: Argument mining for PERSuAsive oNline discussions. 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)* (Hong Kong, China, Nov. 2019), Association for Computational Linguistics, pp. 2933–2943.

[3] Clark, K., Khandelwal, U., Levy, O., and Manning, C. D. What does bert look at? an analysis of bert's attention, 2019.

[4] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[5] Lawrence, J., and Reed, C. Argument Mining: A Survey. *Computational Linguistics 45*, 4 (01 2020), 765–818.

[6] Ribeiro, M. T., Singh, S., and Guestrin, C. "why should i trust you?": Explaining the predictions of any classifier, 2016.

[7] van Eemeren, F., Garssen, B., Krabbe, E., Snoeck Henkemans, A., Verheij, B., and Wagemans, J. *Handbook of Argumentation Theory.* Springer, 2014. M1 - Book, Whole.