Identifying discourse elements in writing by fine-tuning BERT, LongFormer, and GPT-2 models for NER Token Classification

Brad LaMontagne

Abstract—Current automated writing feedback systems cannot distinguish between different discourse elements in a students' writing, and this is a problem because, without this skill, these systems give guidance that is too broad for what the student is trying to accomplish. This is concerning because automated writing feedback systems are a great tool that can be used to combat the decline of writing ability among students. According to the National Assessment of Educational Progress, less than 30% of high school seniors are adept writers. If we can improve on automated writing feedback systems, then we can improve the quality of students' writing and stop the decline of proficient writers among students. Solutions have been proposed to solve this problem, with the most popular approach being to fine-tune a Bidirectional Encoder Representations from Transformers model that can identify the different discourse elements in a student's writing, however, these approaches have their setbacks. For instance, these approaches do not compare the strengths and weaknesses of different models, and these solutions encourage the model to be trained by sequences(sentences) rather than whole essays. In this paper, I redesign the Persuasive Essays for Rating, Selecting, and Understanding Argumentative and Discourse Elements corpus so that models can be trained on entire essays, and I also fine-tuned the Bidirectional Encoder Representations from Transformers, Long Document Transformer, and Generative Pre-trained Transformer-2 models on discourse classification under the framework of a named-entity-recognition token classification problem. Overall, the Bidirectional Encoder Representations from Transformers model train using my pre-processing method of sequence merging out-performed the standard model by 17% and 41% in overall accuracy. I also found that the Long Document Transformer model is best for discourse classification with a total f-1 score of 54%, however, an increasing validation loss from 0.54 to 0.79 suggests that the model is over-fitting. Since the models are over-fitting, some improvements can still be made, such as implementing early stopping techniques and more examples of rare discourse elements during training.

Index Terms—BERT - Bidirectional Encoder Representations from Transformers, NER - Named Entity Recognition, Longformer - Long Document Transformer, GPT2 - Generative Pre-trained Transformer 2, NLP - Natural Language Processing, GSU - Georgia State University

1 Introduction

1.1 The importance of writing

Writing is a crucial skill that is needed in all professions because it allows you to express your thoughts in a clear and precise way. Being able to express one's thoughts and ideas will give a person an advantage when writing business emails, proposals, or arguing against/in favor of a new policy. As the website "Source Expert" states in their article "43 Reasons Why Writing is Important for Students", "There are different ways to communicate with others, but writing is the most important. In other words, it is always going to be a part of your everyday life. . . " [1]. Even though writing is an important skill for people to have, many high school students still suffer from a lack of writing proficiency. The National Assessment of Educational Progress states that less than 30 percent of high school seniors are adept writers [2] and they also show that this problem is even worse in low income communities who have an adept writing rate of less than 15 percent [2]. As the researchers at Georgia State University point out, this problem is mainly due to the fact that many schools, especially those in low-income communities, do not have the resources to provide personalized feedback on a student's writing [2], fortunately, one way to solve this problem is through automated writing feedback. Automated writing feedback systems are programs which can analyze and critique a students' writing if a teacher is not present to do so. These programs are already popular in many applications such as Microsoft Outlook auto suggestions and Grammarly. In fact, Trey from the website "apoven", exclaims how writing feedback systems like Grammarly can be used to expand a person's vocabulary and provide them mini-grammar lessons immediately [3]. In reaction to this, many institutions have taken steps to improve on our current automatic feedback systems.

B. LaMontagne is with the College of Computing and Software Engineering, Kennesaw State University, Marietta, GA, 30060 USA. Emails: blamonta@students.kennesaw.edu

1.2 The current machine learning approach

A machine learning based approach to improve on automated feedback systems has been investigated by both institutions GSU and The Learning Agency Lab. Their belief is that a machine learning model can be trained to accurately categorize the discourse elements in a work of writing. This model can then be added to a pre-existing feedback system to help the system give out better and more constructive feedback for the students. The Learning Agency Lab took the first step towards making this model by creating The

Persuasive Essays for Rating, Selecting, and Understanding Argumentative and Discourse Elements (PERSUADE) corpus. The PERSUADE corpus is a collection of 25,000+ argumentative essays that have been collected from students ranging in 6-12th grade [4]. All the essays have been annotated by professional English teachers so that the different discourse elements are known [4]. After the creation of this dataset, machine learning researchers had the ground truth they needed to start training their models. Specifically, they hoped to fine-tune pre-existing Natural Language Processing(NLP) models for the discourse classification task with a large emphasis being placed on Google's Bidirectional Encoder Representations from Transformers(BERT) model. I agree with the current approach to fine-tune this model for discourse classification; however, I believe a few steps are needed to make these models more accurate.

1.3 The Discourse Elements

This list of discourse elements was put together by a team of teachers and professional writers at The Learning Agency Lab [4]. They believe that this list encompasses all of the major discourse elements that make up a students writing and they used this list as a template while constructing the PERSUADE corpus. I will be using the same rubric when fine-tuning my own models:

- Lead - an introduction that begins with a statistic, a quotation, a description, or some other device to grab the reader's attention and point toward the thesis
- Position - an opinion or conclusion on the main question
- Claim - a claim that supports the position
- Counterclaim - a claim that refutes another claim or gives an opposing reason to the position
- Rebuttal - a claim that refutes a counterclaim
- Evidence - ideas or examples that support claims, counterclaims, or rebuttals.
- Concluding Statement - a concluding statement that restates the claims

1.4 My approach & potential outcomes

Like the current machine learning approach, I believe a transform-based model like Bidirectional Encoder Representations from Transformers(BERT) [5] can be fine-tuned to successfully solve the discourse classification problem. However, in this paper, I want to investigate other transformer models as well and compare/contrast the different results. Additionally, more improvements can be made to the The Persuasive Essays for Rating, Selecting, and Understanding Argumentative and Discourse Elements(PERSUADE) corpus as well. The current corpus splits the essays into sequences where each sequence is a different discourse type, however, I will redesign the dataset so that the models can be fed entire essays during training. In this paper, I hope to prove that transformer-based models should be trained on entire essays at a time to make full use of their architecture. I also hope to prove that it is useful for Machine Learning researchers to assess different models apart from the Bidirectional Encoder Representations from Transformers(BERT) model, and I hope to prove that the Long Document Transformer(Longformer) [6] model is a better model when it comes to discourse classification.

1.5 Outline

To justify my approach, first I will go over other projects that focus on discourse classification. Then I will go more into detail about my approach to the discourse classification problem and the steps that I took to fine-tune the transformer-based models Bidirectional Encoder Representations from Transformers(BERT), Long Document Transformer(Longformer), and Generative Pre-trained Transformer-2(GPT-2). After this I will go over some promising results that my research has uncovered and interpret what those results mean for the task of discourse classification. Finally, I will conclude and bring to light some improvements that could be made to future fine-tuning attempts.

2 Related works

2.1 Towards Automatic Classification of Discourse Elements in Essays

Researchers Burstein et al [7] attempt to use a Bayes Classifier to identify thesis statements in a student's writing. Their model was able to achieve an overall mean accuracy of 43%, but most importantly they were able to prove that thesis statement classification was generalizable. That is to say, the model does not have to be retrained for every new essay prompt, once trained, the model can detect thesis statements in all essay topics. One con however was the Bayes Classifier's low training set size of only 100 essays, which the authors acknowledge could be holding back their model.

2.2 Kleczek Model

Another model which should be mentioned is the fine-tuned Longformer model created by programmer Darek Kleczek [8]. Kleczek's approach to solve this problem was to fine-tune a pre-existing Longformer model on the huggingface website [8] which was originally trained by Machine Learning engineers at allenai. By fine-tuning the Longformer model, Kleczek was able to obtain an accuracy of 61.4

2.3 Rhetorical Structure Theory: looking back and moving ahead

Taboada et al [9] go over the history of Rhetorical Structure Theory(RST) and its merits today. They find that RST is useful for a variety of applications (including discourse classification) and is a "healthy, well-tested theory". Most importantly, they establish that there is a relationship between the different discourse elements, a relationship that our models can hopefully pick up on. The researchers do not attempt to specifically define the relationship between models nor create a machine learning model that can, rather they leave it as an open-ended question to be solved by others.

2.4 The Peller Sentence Classifier

In this notebook, machine learning researcher Julián Peller [10] tackles the discourse element classification problem by fine-tuning Google's BERT model. He approaches the problem as a named-entity-recognition(NER) token classification problem where the essay is a list of tokens and the discourse elements are different classes. He also sampled 10,000 essays from the PERSUADE corpus for training and his overall F-1 score accuracy was 0.226.

2.5 Habiby RoBERTa Q&A model

In this project, machine learning researcher Ali Habiby [11] solves the discourse element classification problem in a rather unique way. Instead of defining the problem as a NER token classification problem, Habiby frames the problem as a Q&A problem which allows him to use a Q&A model. The transformer model that Habiby chose to fine-tune was the RoBERTa, a model created by facebook which was inspired by BERT. Habiby used a max length of 448 tokens, a stride length of 192 for his model, and trained his model for 3 epochs. His overall F-1 score was 0.453.

2.6 Deotte Weighted Box Fusion and Post process

In this project, researchers Deotte, Lee, and Bamba [12] utilized several machine learning techniques in their approach to the discourse element classification problem. The first technique they employed was weighted box fusion in order to combine the output of 10 different models into one decision. Most of the models used were variations of the deberta model as well as the longformer model. After getting the model results, the team used post processing such as repairing span predictions and discourse specific rules to clean up the output of the model after predictions were made. The overall F-1 score was 0.74 and the model was trained for 5 epochs on Nvidia's V100 32GB GPUs and A100 40GB GPUs.

2.7 Habiby Random Forest model

In this project, machine learning researcher Ali Habiby [13] solves the discourse element classification problem with the random forest model instead of his previous Q&A model. One benefit to this model is that it is very simple to understand and replicate. The training/test split that Habiby decided to use for this model was 70% for training and 30% for testing and the overall f-1 score for the model was 0.25 [13]. Although this model is very easy to replicate and understand, I believe that this model was too simple to pick up on how different discourse elements are related to each other given the low f-1 score.

2.8 Lonie Keras Model

In this notebook, Lonnie [14] uses the Keras library in order to create an LSTM network that can classify discourse elements in a student's essay. Some notable layers that are included in Lonnie's model is the embedding layer of length 1024 [14]. This is important because most other solutions are fine-tuned versions of the BERT model, however, the BERT model can only take in 512 tokens at a time. Thus, Lonnie's model is better able to accommodate larger student essays than most other solutions, however, Lonnie still trains the data one sequence at a time which I believe is holding back his model from its full potential. Overall, Lonnie's model had an f-1 score of 0.214 [14].

2.9 Drakuttala Fine-tuned RoBERTa model

In this project, machine learning researcher Drakuttala [15] fine-tuned the RoBERTa base model to solve the discourse element classification problem. One detail that stands out in Drakuttala's approach is the way he defines the different elements during his model training. Unlike most researchers who used 7 classes(Lead, Position, Claim, CounterClaim, Rebuttal, Evidence, and Concluding Statement). Drakuttala split these elements into two parts, B and I. The B class was for words at the beginning of an entity and the I class was for words inside an entity, so for example, instead of having one Lead class, Drakuttala has two Lead classes B-Lead and I-Lead. Drakuttala was able to achieve an overall f-1 score of 0.54 while training on 3 epochs, a 1e-5 learning rate, and a token length of 512 [15].

3 Approach (and technical correctness)

3.1 PERSUADE corpus

The training and testing data that I used for fine-tuning my models was the PERSUADE corpus, a dataset which was created by the Learning Agency Lab. I chose this dataset because it was a dataset designed specifically for the discourse classification problem. This corpus holds 25,000+ student essays that all have been annotated by writing professionals [4]. To make sure that the dataset was as accurate as possible, each essay was annotated using a double-blind rating process as well as an adjudication from another 3rd writing professional [4]. The contents of this dataset are outstanding and very useful for training/testing models; however I believe that some changes can be made to the formatting of the dataset through data preprocessing.

3.2 Data Preprocessing

To preprocess the data for this model, I decided to merge the individual sentence sequences back into a collective essay. Inside of the PERSUADE corpus, the essays are split up into sequences where each sequence is a different discourse element. I believe that this is not the best way to fine-tune a transformer-based model because these models employ positional encoding. Positional encoding was a technique added to transformer architecture because the model does not recurrence meaning that the sequences "Hello World" and "World Hello" look the same to the transformer model [16]. With positional encoding added to the word embeddings, the transformer model was able to learn that different word positions have different meanings in a text, and I believe that this tool can be exploited for the purpose of discourse classification. This is because some discourse elements, like the concluding statement, are heavily correlated to their position in an essay; Merging the sequences before fine-tuning starts gives the model a chance to learn how the placement of a sequence in an essay might correlate to its discourse type.

3.3 Three different models (BERT, Longformer, and GPT-2)

The three models that I chose to fine-tune in this paper were the BERT, Longformer, and GPT-2 models. I chose to fine-tune multiple models because I wanted to see how different model architectures react to the discourse classification problem; I was also curious as to whether different models would be better at classifying different discourse elements. I chose the BERT model because this model is one of the most popular when it comes to NLP tasks. According to the Huggingface database, the BERT model is the 2nd most popular NLP model and was downloaded by researchers 15.8 million times as of April 2022 [17]. In order to compare my results to the work of other researchers I have decided to include this model in my own research. Another model that I will fine-tune is the GPT-2 model. I included this model in my project because it is a popular model, but most importantly because of the model's design. Unlike the BERT model which has stacking transformer encoding layers, the GPT-2 architecture has stacking transformer decoding layers [18]. In this paper, I hope to discover whether this slight change in design has any effect on the output of the discourse classification results. The last model that I will fine-tune and the model that I believe has the most promise is the Longformer model. The Longformer model is an extension of the BERT model that is designed to manage larger input values without losing quality [6]. This feature is important to my research because my data preprocessing will result in longer input values which can cause most models to forget information they learned at the beginning of a sequence. The Longformer model is important to my research because it can reap the benefits of my data preprocessing without suffering in quality. I believe this model will show just how far my preprocessing technique can improve a model.

3.4 Hyper-parameters

For the model hyper-parameters I used:

- batch-size = 1
- Learning-rate = 5e-5
- Epochs = 7
- Warm-up ratio = 0.1
- Gradient-accumulation = 8
- Weight-decay = 0.01

3.5 F-1 score

To evaluate the models, I will be using the f-1 score. To calculate the f-1 score using the following formula:

$$F-1score = TP/(TP + 0.5 * (FP + FN))$$

Before we can use this formula, we need to find the values for true positive, false positive, and false negative as defined by the GSU researchers. As seen in this post by the GSU team [19], each model evaluation will have a ground truth and predictions. The ground truth is what discourse classes the sequences (group of words) belong to, and the predictions are what classes the model thinks the sequences belong to. If the prediction sequence has a 50 percent or greater overlap with the ground truth sequence, then that will count as one true positive. If there is an unmatched prediction sequence then I count that as a false positive, and if there is an unmatched ground truth sequence then I count it as a false negative. Figure 5 shows an example of these prediction sequences and goes into greater detail as to how they are calculated.

4 Experimental results (and technical correctness)

4.1 Data preprocessing and sequence merging

As seen in the table "Trained Models and their F-1 scores", the BERT model that was trained without my data preprocessing method scored 17% lower in f-1 score[Figure1] and 41% lower in accuracy[Figure 7] than the model that used my version of data preprocessing. This is because Lead and Concluding statements almost always occur at the beginning and end of an essay respectively. My models were able to take advantage of positional encoding and learn the relationship between the lead and concluding statement and their position in a student's essay. My work has shown that the best way to train transformer-based architecture on discourse classification is to merge the sequences back into a full essay and let the model study the relation between discourse elements by utilizing their positional encoding.

4.2 Comparing transformer-based architecture for discourse classification

For my experiment, I fine-tuned 3 transformer models off of the Huggingface library: BERT [20], Longformer [21], and GPT2 [22]. From the table "Trained Models and their F-1 scores" we can see that out of all the models that were fine-tuned, the Longformer model performed best with an f-1 score of 0.535. The BERT model came in 2nd place with an f-1 score of 0.395 and the GPT2 model was the worst performing with a score of 0.362. My work here has shown that the best model for discourse classification is the Longformer model. I believe that the Longformer's ability to handle large data input without losing vital information is what made this model so successful in my experiments.

4.3 High Lead/Concluding Statement scores

All of the models score relatively highly on the Lead and Concluding Statement categories and low on the Counterclaim category. According to Figure 4, the average f-1 scores for Lead and Concluding statements were 0.751 and 0.587 respectively, these scores were the two highest out of all categories. This goes against conventional wisdom, since Leads and concluding statements are not as frequent as other categories such as claim; one would assume that the claim category would be highest since the models had more examples to train on. I believe that these results occurred because the Lead and Concluding statements have a close relationship to their position in an essay. That is to say, Lead and Concluding statements are almost always at the beginning and end of a work respectively, so the model would have an easier time learning these classes using positional encoding. Thus, my work here has shown that feeding the model entire essays allows the model to excel at categories that are not as common. However, for some categories such as rebuttal and counterclaim, more training examples might be needed.

4.4 Increasing validation loss

All models started to show an increase in validation loss after the 3rd epoch, for instance, the best performing Longformer model increased its validation loss from 0.54 to 0.79 across epochs 3 to 7 [Figure 2]. According to the javatpoint article "Overfitting in Machine Learning" [23], a clear sign of an overfitting model is increasing validation error during training and one way to prevent this is early-stopping. Early stopping as defined by the website "Elite Data Science" is the process of "...stopping the training process before the learner passes that point[the point where the variance starts to increase]..." [24]. I believe that for my models, I should implement early stopping around the 2nd or 3rd epoch since that is when the variance started to increase. Another approach I could try is increasing the examples during the training phase. According to Farhad Malik in his article "The Problem Of Overfitting And How To Resolve It" [25], passing in more training examples is a great way to solve the overfitting problem. Specifically, I should feed in essays that have plenty of counterclaim and rebuttal examples since that is where my models performed weakest. My work here has shown that when fine-tuning a model for discourse element classification, a greater emphasis needs to be placed on getting more examples rather than running the model on a high number of epochs.

5 Conclusion

In conclusion, writing is an important skill to have and it is crucial for young adults to develop their writing skills. By using automated writing feedback systems, we can help students foster their writing talent by providing detailed analysis of a student's writing. One way to improve upon current automated writing feedback systems is to integrate them with machine learning models that distinguish between different writing elements in a student's essay. In this experiment I demonstrated that Longformer models are better at discourse classification than BERT or GPT2 models. I also showed how passing in the entire essay during fine-tuning leads to the model learning the positional relations between discourse elements, especially for the classes Lead and Concluding Statement. However, positional encoding will not solve the discourse classification problem alone, a greater emphasis needs to be put on getting data on more categories like rebuttal or counterclaim to improve results overall.

Fig. 1: Macro f-1 scores of all models after training

Trained Models and their F-1 scores		
Model Name	F-1 score	Accuracy
BERT (baseline model)	0.225	0.331
BERT	0.395	0.736
GPT-2	0.362	0.765
Longformer	0.535	0.826

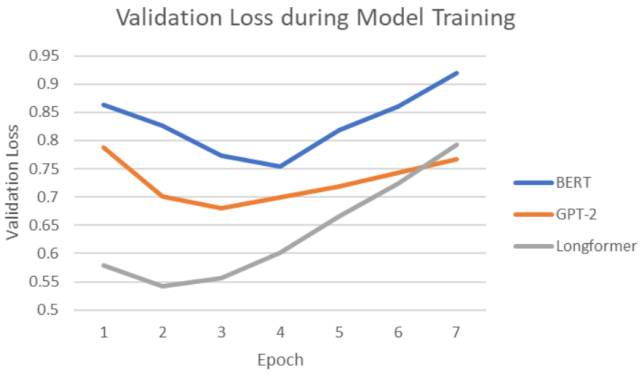

Fig. 2: Validation loss during model training

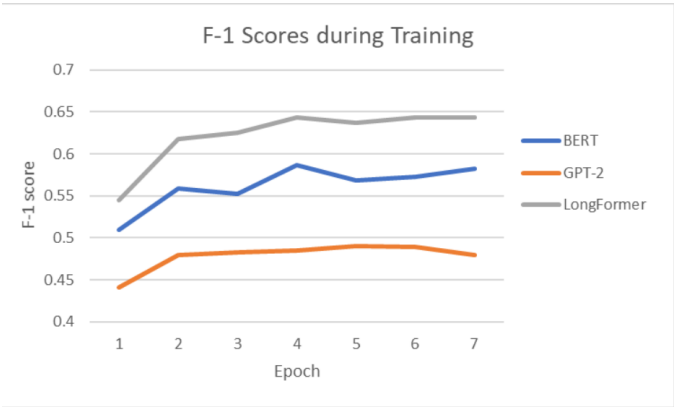

Fig. 3: F-1 scores during model training

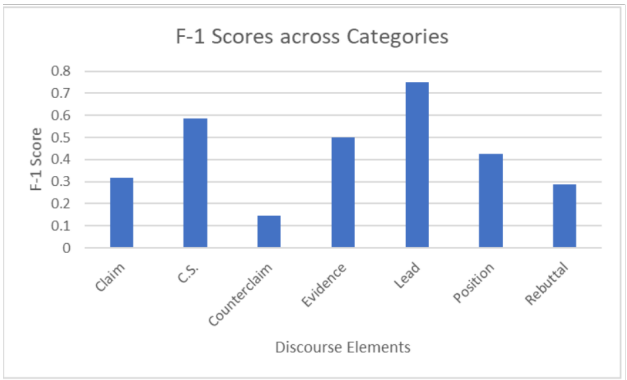

Fig. 4: Average F-1 scores across all models (except baseline) for each discourse element

References

[1] SourceExpert, "43 reasons why writing is important for students in 2022," Apr 2019. [Online]. Available: https://srcxp.com/why-writing-is-important-for-students/

[2] G. S. University, "Feedback prize - evaluating student writing," Dec 2021. [Online]. Available: https://www.kaggle.com/c/feedback-prize-2021

[3] Trey, "5 reasons to use grammarly," Oct 2019. [Online]. Available: https://www.apoven.com/grammarly-benefits/

[4] May 2021. [Online]. Available: https://www.the-learning-agency-lab.com/the-feedback-prize/

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[6] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," *arXiv preprint arXiv:2004.05150*, 2020.

[7] J. Burstein, D. Marcu, S. Andreyev, and M. Chodorow, "Towards

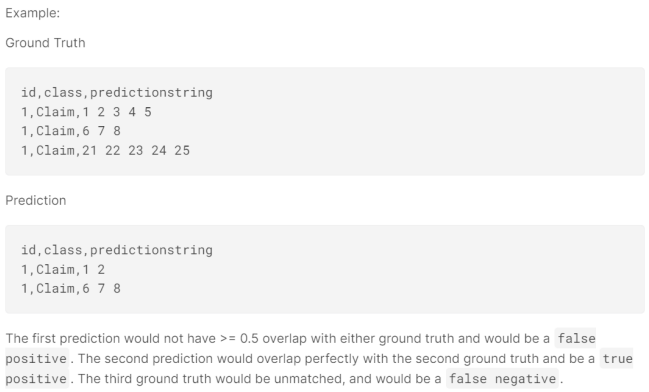

Fig. 5: How TP/TN/FN are calculated [19]

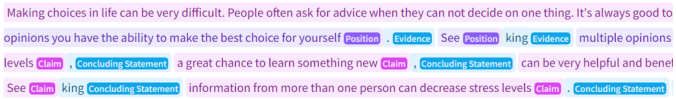

Fig. 6: Sample of model output

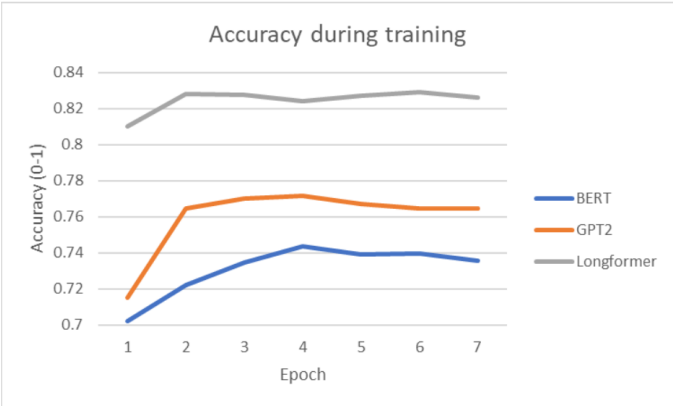

Fig. 7: Accuracy during model training

automatic classification of discourse elements in essays," in *Proceedings of the 39th annual meeting of the Association for Computational Linguistics*, 2001, pp. 98–105.

[8] thedrcat, "Feedback prize huggingface baseline: Inference," Dec 2021. [Online]. Available: https://www.kaggle.com/thedrcat/feedback-prize-huggingface-baseline-inference

[9] M. Taboada and W. C. Mann, "Rhetorical structure theory: Looking back and moving ahead," *Discourse studies*, vol. 8, no. 3, pp. 423–459, 2006.

[10] J. Peller, "Feedback- baseline sentence classifier [0.226]," *Kaggle*, Dec 2021. [Online]. Available: https://www.kaggle.com/code/julian3833/feedback-baseline-sentence-classifier-0-226/notebook

[11] A. Habiby, "Roberta qna model," *Kaggle*, Jan 2022. [Online]. Available: https://www.kaggle.com/code/aliasgherman/roberta-qna-model-maxlen-448-stride-192

[12] C. Deotte, C. Lee, and U. Bamba, "2nd place solution - [cv741 public727 private740]," *Kaggle*, Mar 2022. [Online]. Available: https://www.kaggle.com/code/cdeotte/2nd-place-solution-cv741-public727-private740

[13] A. Habiby, "Randomforest only (gradientboostnow)," *Kaggle*, Jan 2022. [Online]. Available: https://www.kaggle.com/code/aliasgherman/randomforest-only-gradientboostnow-0-25-lb

[14] Lonnie, "Name entity recognition with keras," *Kaggle*, Dec 2021. [Online]. Available: https://www.kaggle.com/code/lonnieqin/name-entity-recognition-with-keras

[15] raghavendrakotala, "Fine-tunned on roberta-base as ner problem [0.533]," *Kaggle*, Dec 2021. [Online]. Available: https://www.kaggle.com/code/raghavendrakotala/fine-tunned-on-roberta-base-as-ner-problem-0-533

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[17] Huggingface, "Models," Apr 2022. [Online]. Available: https://huggingface.co/models

[18] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[19] [Online]. Available: https://www.kaggle.com/c/feedback-prize-2021/overview/evaluation

[20] [Online]. Available: https://huggingface.co/bert-base-uncased

[21] [Online]. Available: https://huggingface.co/allenai/longformer-base-4096

[22] [Online]. Available: https://huggingface.co/gpt2

[23] [Online]. Available: https://www.javatpoint.com/overfitting-in-machine-learning

[24] Sep 2017. [Online]. Available: https://elitedatascience.com/overfitting-in-machine-learning

[25] F. Malik, "The problem of overfitting and how to resolve it - fintechexplained - medium," *FinTechExplained*, Aug 2019. [Online]. Available: https://medium.com/fintechexplained/the-problem-of-overfitting-and-how-to-resolve-it-1eb9456b1dfd