Organizing debate, debating organization

Mohamed Chenene, Olga Kavvada, Jean Danielou, Phillipe Calvez

Institut pour la Maîtrise des Risques
Sûreté de Fonctionnement - Management - Cindyniques

Organizing debate, debating organization

Controversy Analysis using a 2-step Approach for Argument Extraction and Classification

Analyse de controverses grâce à une approche en 2 étapes pour l'extraction et la classification d'arguments

Mohamed CHENENE
Computer Science and Artificial Intelligence Lab
ENGIE Lab CRIGEN
France
mohamed.chenene@external.engie.com

Olga KAVVADA
Computer Science and Artificial Intelligence Lab
ENGIE Lab CRIGEN
France
olga.kavvada@engie.com

Rim HANTACH
Computer Science and Artificial Intelligence Lab
ENGIE Lab CRIGEN
France
rim.hantach@external.engie.com

Jean DANIELOU
Centre de Sociologie de l'Innovation
CNRS, UMR 9217,
Mines ParisTech, PSL University
France
jean.danielou@mines-paristech.fr

Phillipe CALVEZ
Computer Science and Artificial Intelligence Lab
ENGIE Lab CRIGEN
France
philippe.calvez1@engie.com

***Abstract*—**Controversy analysis is a broad topic where the opinions of different stakeholders are analyzed to identify the various arguments that are stated and classify the positions taken on the subject. Gathering this data can be very time-consuming to do manually and usually is error-prone and not exhaustive. Automated text classification can enhance this process and make it possible to analyze controversial topics by analyzing lists of relevant documents in a short timeframe. In this paper, we propose a 2-step approach to optimize the extraction and classification of arguments in textual data from controversial topics. First, we extract the most relevant paragraphs for the controversy with a retrieval model and then we use an argument mining model to find and classify the relevant arguments. With this method, we are able to successfully characterize long documents and understand the various opinions that are recorded.

***Résumé*—**L'analyse de controverses est un vaste sujet dans lequel les opinions des différentes parties prenantes sont analysées afin d'identifier les arguments et les positions prises sur le sujet. La collecte manuelle de ces données peut être longue, source d'erreurs et non exhaustive. La classification automatique de textes peut améliorer ce processus et permettre d'analyser des sujets controversés dans un délai court. Dans cet article, nous proposons une approche en 2 étapes pour optimiser l'extraction et la classification d'arguments dans des données issues de sujets controversés. Tout d'abord, nous faisons l'extraction des paragraphes les plus pertinents pour la controverse à l'aide d'un modèle de recherche. Puis nous utilisons un modèle d'extraction d'arguments pour trouver et classifier les arguments pertinents. Grâce à cette méthode, nous sommes en mesure de caractériser avec succès de longs documents et de comprendre les différentes opinions enregistrées.

***Keywords*—**Argument Mining, Information retrieval, Controversy analysis

I. Introduction

Nowadays, companies collect a huge amount of unstructured data (PDF, web pages, etc) and thus have new resources to utilize to make insightful decisions. Controversy analysis is a research area that requires reviewing and understanding and extracting arguments from long documents. Dealing manually with these documents can be exhausting, time-consuming and error-prone. Utilizing large amounts of data to develop automated text classification can be truly beneficial to extract different points of view in a controversial topic that would assist with the task of analyzing a controversy.

Argument mining (AM) is a promising field of research in Natural Language Processing (NLP) [1]. In this field, the focus is on the analysis and extraction of arguments in texts. There are many sub-tasks related to AM. The one that we will look at in this paper is stance classification, which consists of classifying arguments (support, attack, neutral) given a certain topic. Thanks to the recent advancements in NLP and Deep Learning, with models like BERT [2], we can obtain results with a high precision on this complex task. The advancements in the field of NLP have of course been aided by the enrichment of available tagged data sources that can be used to develop complex models to solve the tasks at hand.

Information retrieval (IR) is a well-known field of research in NLP in which deep learning models enhanced our ability to extract pertinent results. It has been applied in different

areas such as the document extraction on question-answering systems. We can apply the methods developed in this field in order to extract text that could potentially contain arguments before trying to classify the arguments. This filtering step allows the creation of a smaller list of candidates per document for the argument extraction. In this manner, we can reduce the number of errors by avoiding irrelevant passages because the subset contains significantly less unrelated paragraphs.

In this paper, we will try to resolve the following problem: how can we extract and classify arguments from heterogeneous texts discussing a specific controversial topic with high precision? Most papers related to AM use a sentence-level approach in which they subdivide documents into sentences and apply deep learning models to find arguments. In our case, we must overcome two issues. The first one is the length of the documents. The articles in our set of interest contain documents with many pages, some of which with more than 100 pages most of which contains irrelevant information for the controversy analysis. Applying the popular approach will lead to many false arguments and our model will not be pertinent enough. The second important issue is the fact that our analysis is focused on one controversial topic around the sustainability of using biomass as an energy source. In fact, there is a debate at the European level concerning the sustainability of the biomass energy mainly because people cannot agree on how the carbon footprint is computed. We can find scientific papers, blogs that deal with this subject. As the topic is fairly contained, there is not a great amount of data sources that would enable us to develop a strong algorithm that would be able to understand the topic and correctly extract the right arguments. Applying a generic AM model to such a specific topic would result in poor performance.

In order to tackle this problem, we propose an approach which will be more resilient to long documents. The approach will be divided in two steps:

- For each document, we will select the most relevant paragraphs by using IR techniques. As we are not interested in extracting all the possible arguments in a document, the idea is to work on a subset of paragraphs which is more likely to contain arguments. This way we decrease the number of wrong arguments extracted compared to the sentence-level approach.
- For each paragraph selected, we will infer the stance of each sentence and retrieve arguments by using BERT models trained on an AM dataset.

In section 2 of this paper, we will introduce the state of the art related to both argument mining and information retrieval. In section 3, we will present in detail our methodology to tackle the problem of argument extraction and the models used for our comparative study. Then, we will present our experimental results on the topic of the sustainability of biomass energy in the section 4. Finally, section 5 concludes our work and introduces eventual further works.

II. Related work

This section introduces the related work in argument mining and information retrieval as well as the different model architectures that we used in our work.

Compared to sentiment analysis, the study of arguments is more complex. To determine the polarity (positive or negative) of a sentence, one can create a machine learning model using a sentiment lexicon and this can lead to improved results. This method, however, will not be sufficient for argument detection because even if the study of the word used is important, the structure of the argument is another important feature. An argument can take different forms that can be separated into some main blocks. Often an argument is composed of one or multiples premises followed by a claim/conclusion. A premise is the starting point of an argument: it is a reason to support the claim. The claim is the main argument. It supports or attacks a statement. The overall position of the argument towards a statement is called the stance (pro or against). In Fig. 1, we show some examples of arguments that could appear in the debate related to the biomass.

In the early work of [3], they present a first approach to detect arguments in legal documents partly based on the study of their structure. Today, AM is well-defined and we can find many sub-tasks related to this field. As presented in the survey of [1], we can find tasks such as argument detection, classification of arguments into “premise” or “claim” classes and classification into support or attack arguments. The last task has been redefined in the work of [4] where they focus on automatic claim detection. They name a statement that support or attack a given controversial topic as Context-Dependent Claim (CDC). Similarly, [5] worked on automatic evidence detection where they differentiate evidences between expert evidences and study evidences to extract them.

With the arrival of deep learning and the increase in the amount of unstructured data, many tried to apply AM on large and diverse dataset and create tools for web crawling. In [6] paper, they worked with articles from Wikipedia. To deal with the fact that most sentences were not arguments, they used a query to select potential candidates. They looked at sentences with a specific form and that can characterize the argument structure. In [7], they used the same idea and worked on a dataset of 10 billion sentences from newspapers. Now, we can find many tools for AM such as MARGOT, TARGER and ArgumenText ([8]–[10]) that extract arguments from the web given a topic.

Recent works on topic-dependent stance classification showed that methods using BERT (Bidirectional Encoder Representations from Transformers, [2]) outperformed other models [11]. This is why we decided to work with it in our project.

In the field of Information retrieval, we will focus on the task of passage ranking. Given a query and a list of documents, we want to rank the documents in terms of similarity with the query. One notable tool is BM25 [12]. It is a statistical method based on term frequency and inverted document frequency

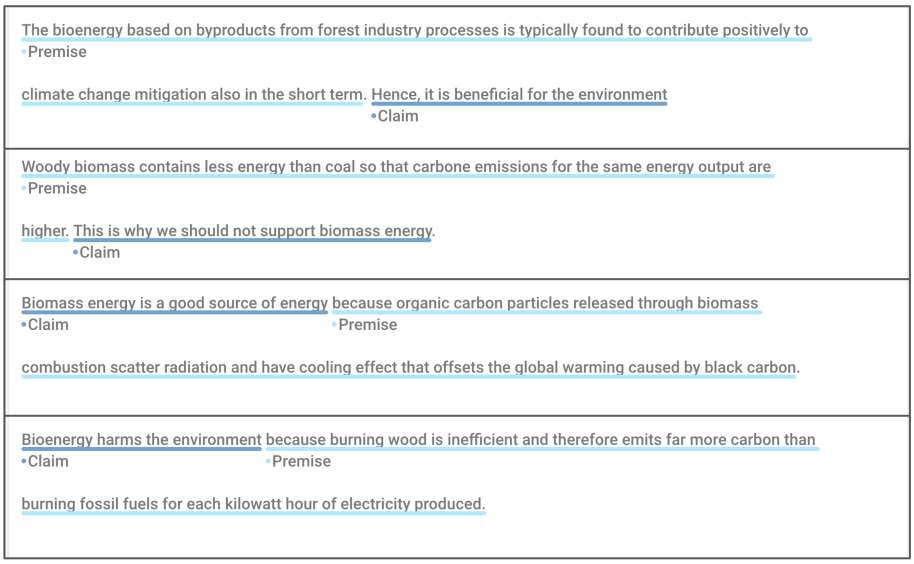

Figure 1: Examples of arguments in the biomass controversy debate

(TF-IDF). It is an efficient method that is easy to implement but it is limited as it only looks at the query terms and similar terms are not considered. For example, in the biomass energy controversy, if we use as query the term the word "biomass" to select documents we will skip documents that speak of "bioenergy" even if they are similar terms. This issue can be overcome by creating exhaustive lists of terms that could be found in relevant documents.

Another approach would be to use word embedding. Word2Vec [13] is a word embedding algorithm that converts a word into a dense vector that encapsulates the meaning of the word. However, we cannot apply this method at the document level because it is difficult to aggregate the meaning of each word in the document. A sentence embedding approach is proposed in [14] which is based on the BERT model. The Sentence-BERT evaluation is made on semantic textual similarity tasks where we can see that it outperforms methods using word embedding. We can find many sentence embedding models trained for specific tasks. In our work, we wanted to retrieve passages similar to a query. Hence, we looked at the MSMARCO model [15] which is trained on a dataset containing more than 1 million pairs of questions and answers. There is also the dense passage retrieval (DPR) [16] model that is specifically trained on this task of passage ranking using a query. The model is composed of two BERT models. One is used for the embedding of the question, the other for the embedding of the passage. DPR outputs a score associated to the pertinence of the passage. These three approaches (BM25, Sentence-BERT and DPR) are going to be evaluated using the biomass energy dataset.

III. Methodology

In this section, we present our approach to extract and classify arguments from articles. First, we will use IR techniques to extract the most relevant paragraphs to decrease our false positive rate and focus on the paragraphs that are more related to the topic at hand. Second, we will apply a deep learning model to do both extraction and classification of arguments. In Fig. 2, we present the different parts of our methodology.

A. Paragraph retrieval

For each document, we want to retrieve the most relevant paragraphs for the extraction of arguments related to our topic (biomass energy). Sentence-BERT, which is one of the models tested to convert paragraphs into vectors, has a limit of characters on the input, thus we decided to delete stop words for the paragraph retrieval task. In very long paragraphs we can lose information thus having shorter paragraphs without stop words is important.

1) Creating the query: To get relevant paragraphs, we must create the most fitting query that would describe well the topic of interest. It can be either natural questions, examples of sentences that we are searching or a list of keywords. We proposed two types of queries:

- *Expert queries*: By discussing with an expert of the biomass controversy, we manually created 9 questions that address the topic. Here are some examples: "Does extracting biomass destroy biodiversity?", "Does carbon sink and stock decrease?", "Does certification protect forest?", etc...
- *Argument queries*: We selected a list of arguments that were labelled and used them as queries. We gathered 410 arguments. These arguments come from the training split of the Biomass corpus. The details are in the section Dataset.

2) Rank and retrieve: For the paragraphs ranking and retrieval, we want to associate a score to a paragraph which will characterize the relevance of that paragraph. Then, we rank each paragraph by their score and select the top-k paragraphs (Fig. 3). For the hyperparameters k we chose 5 because

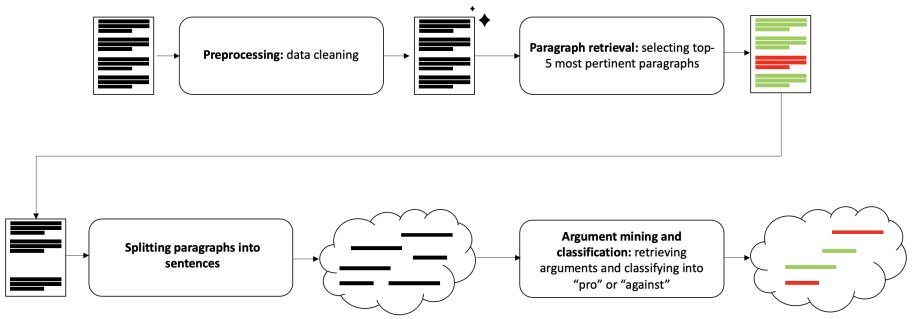

Figure 2: Methodology for the extraction of arguments

most documents contain more than 5 paragraphs ($> 90\%$) and the number of arguments returned with the pipeline was satisfactory. We did not make a quantitative study to fine-tune this parameter.

In our work, we have made a comparative study of different models (BM25, Sentence-BERT, DPR) for the ranking and retrieval task. Each method uses a different score computation. In the next part, we will present how the score is computed (we do not present DPR scoring method because it is similar to that of Sentence-BERT).

3) BM25: It is a statistical method that directly looks at terms frequencies in documents.

Given a list of query terms $Q = \{q_1, .., q_n\}$ and a list of paragraphs $P = \{P_1, .., P_N\}$, we can compute a relevance score that says how much a document is close to the query. We changed the format of our queries for this method. We aggregated all the queries into a single one and we split it by words (q_i is a word from the queries whereas Q_i in Sentence-BERT is either a question from the expert queries or an argument from the argument queries). For a document P_i, the score is computed as follows:

$$score(P_i, Q) = \sum_{j=1}^{n} idf(q_j) \times \frac{tf(q_j, P_i)(k_1 + 1)}{tf(q_j, P_i) + k_1(1 - b + b\frac{|P_i|}{avgdl})} \quad (1)$$

where $idf(q_j) = log(\frac{N}{|\{P_i \,:\, q_j \in P_i\}|})$ is the inverse document frequency, $tf(q_j, P_i)$ corresponds to the term frequency of the term q_j in the paragraph P_i , $|P_i|$ is the length of a paragraph, $avgdl = \frac{1}{N}\sum_{i=1}^{N} |P_i|$, k_1 and b are hyperparameters [1]. Since we are using only one query, we directly have the score of the paragraph.

4) Sentence-BERT: This method is based on the embedding of the text. Given a paragraph, Sentence-BERT will convert the passage into a dense vector of size 768 that encapsulates the meaning of the passage. The same approach is used to convert the queries into vectors.

Given a list of queries (expert or argument queries) $Q = \{Q_1, .., Q_n\}$ and a list of paragraphs (from a document) $P = \{P_1, .., P_N\}$ embedded with a Sentence-BERT model, we want to compute a relevance score that describes how much a paragraph is similar to the query. To compute the similarity between a paragraph P_i and a query Q_j, we used the cosine similarity:

$$score(P_i, Q_j) = \frac{P_i \cdot Q_j}{\|P_i\| \times \|Q_j\|} \quad (2)$$

The calculated score is between -1 and 1. The higher the score the closer the paragraph and the query are. Since we need to associate a unique score to each paragraph, we took the maximum score of a paragraph on the list of queries: $score(P_i, Q) = \underset{j \in [|1,n|]}{max}\ score(P_i, Q_j)$.

[1] https://en.wikipedia.org/wiki/Okapi_BM25

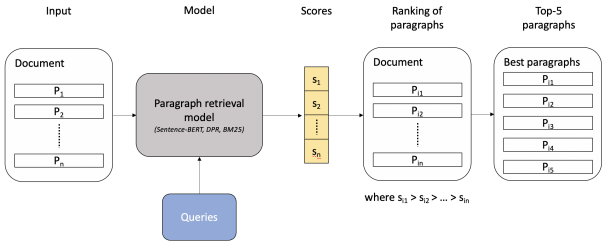

Figure 3: Paragraph retrieval process

B. Argument retrieval and classification

For this task, we trained a model that would perform for both retrieving and classifying arguments tasks. The architecture of the model is shown in Fig. 4.

- The first block of our model is a **BERT** model. For the comparative study we used small versions of BERT: ALBERT, DistilBERT, DistilRoBERTa ([17]–[19]) because we have a small dataset. The model takes as input a sentence and the topic associated to the sentence (e.g. biomass energy) as it is done in [11]. Then it is passed through a tokenizer that will convert it into a list of tokens (A token is a sequence of character that has a meaning, which can be a single character, a subword or a complete word). Then, we use a BERT model that will process each token and output a vector of size 768. For the analysis of the sentence, we will look at the special token "[CLS]". This token is used to represent the whole sentence and it is widely used for classification tasks.

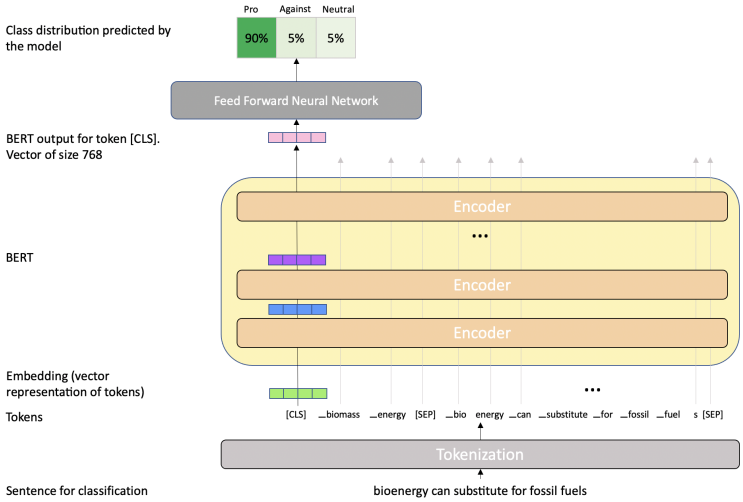

Figure 4: BERT architecture for sentence classification

- The second block is a **feed-forward neural network** that takes the output of the "[CLS]" token, performs the classification task and outputs a vector of size 3. This vector represents the probability distribution of the sentence in the 3 classes "pro","against" and "neutral". The label of a sentence is then assigned as the class which corresponds to the highest probability.

Each paragraph selected during the paragraph retrieval phase is split into sentences. Then, each sentence passes through the classification model. We select then the sentences labelled "pro" or "against". We optimized this process by adding a filtering of the arguments using a threshold. It allows us to extract the sentences that the model is more confident about.

IV. Evaluation

A. Dataset

In this section, we present the datasets that we used in our study to evaluate our proposed approach.

1) UKP Sentential Argument Mining Corpus [20]: It is a rich dataset for topic-dependent stance classification. It contains 25,492 sentences from 8 controversial topics (death penalty, cloning, abortion, gun control, school uniforms, marijuana legalization, minimum wage, nuclear energy) labelled for stance classification (neutral, pro, against).

2) Biomass Corpus: For our analysis, we worked specifically on the biomass energy controversy. The dataset is composed of articles coming from academic papers and organizations' websites. This dataset contains 76 documents of various lengths that all revolve around the biomass controversy. The number of paragraphs per document fluctuates between 1 and 876 with a median around 25.

We manually extracted and labelled arguments from those documents. We extracted random sentences to get 50% of neutral sentences (sentences without arguments) because we want our model to be resilient to false positive predictions (Table I).

For the training and the evaluation for the different tasks we divided the dataset into a train dataset composed of 53 documents (70%) and a test dataset composed of 23 documents (30%).

3) Preprocessing: For each document in the Biomass Corpus, we had to extract the text from PDF or HTML files and split it into paragraphs by using regular expressions. For research papers, we deleted references by creating a clustering algorithms that detects references as outliers.

For both datasets, we applied data cleaning processes. We lowered the text and removed emails, weblinks, parentheses, brackets, punctuation (except dots to delimit the sentences).

4) Training and optimization: The model is trained on UKP Sentential Argument Mining Corpus dataset. It is trained on 3 epochs, the training batch size is 16, the learning rate (= 7.98e-6) is fine-tuned using a hyperparameter search tool called Optuna [21]. For the loss function, we use a weighted cross entropy to deal with the fact that the classes are imbalanced (56% "neutral", 24% "against" and 20% "pro").

Then, the model is fine-tuned on our small dataset for the biomass controversy (epochs = 3, training batch size = 16, learning rate = 2.10e-5). Even if the first set do not contain sentences related to our topic, the model learns features that can help for the stance classification task. Hence, the model will perform better on our dataset.

B. Evaluation measure

We evaluated separately models for the paragraph retrieval task and the stance classification task. Here we present the different metrics we used to evaluate the models.

Table I: Classes distribution

	Sentences	Neutral	Pro	Against
Biomass dataset	1291	704	132	455

1) Paragraph retrieval: For the evaluation of this task, we used the Normalized Discounted Cumulative Gain (nDCG) metric [22]. This metric is used to evaluate the pertinence of web search. It can be used in our case to evaluate the ranking of the paragraphs of our models. To do so, we labelled 17 documents from the test set of the Biomass Corpus (total of 372 paragraphs). For each document with n paragraphs $D = \{P_1, ..., P_n\}$, we gave a relevance score $rel_i \in \{0, 1, 2\}$ for each paragraph. The higher the score the more relevant the paragraph is for the controversy analysis.

Let's note $p_1, ..., p_n$ the ranking predicted by our model and $t_1, ..., t_n$ the true ranking of the paragraphs based on their relevance score. Since we extract the top-k elements in a document, we will compute the score for the k first paragraphs. For each document, we define the predicted DCG (pDCG) and the ideal DCG (iDCG) as:

$$pDCG = \sum_{i=1}^{k} \frac{rel_{p_i}}{log_2(i+1)} \tag{3}$$

$$iDCG = \sum_{i=1}^{k} \frac{rel_{t_i}}{log_2(i+1)} \tag{4}$$

Thus, the nDCG is:

$$nDCG = \frac{pDCG}{iDCG} \tag{5}$$

The score for the dataset is obtained by taking the mean nDCG of all the documents.

2) Stance classification: For this task we focused on the precision of the "pro" and "against" classes. To take this into account, we used the F_β metric:

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{\beta^2 \cdot precision + recall} \tag{6}$$

The β coefficient determines the weight of the recall compared to the precision. In our case, we chose $\beta = 0.5$ because we consider the recall less important than the precision. This is because we want to identify correctly the overall position of the document. We need to detect the stance of a subset of arguments as usually there is consistency in documents on whether the authors are in support or opposed to the biomass controversy. We take the mean F_β score for "pro" and "against" classes for the evaluation of our model.

C. Results

1) Paragraph retrieval: We compared the different models with the two types of queries using nDCG on the Biomass dataset. The results can be seen in Table III.

The results of the experiment shows that the BM25 model performs better for this task. It can be explained by the fact that arguments developed in the biomass controversy use a specific lexicon and BM25 is very sensitive to the vocabulary. Moreover, the fact that we created long query lists helped the model in its research. Another advantage of BM25 is its inference time. Compared to the DL methods, the BM25 is faster even with the argument queries that contain 410 arguments.

2) Stance classification: For this task we compared different BERT models on the test dataset of UKP Corpus. We compared in our study the small variants of BERT model in order to avoid overfitting when fine-tuning the models on the biomass dataset. The comparative results are shown in Table II.

To increase our precision, we use a threshold to select only sentences that included arguments that were classified with high confidence by the model. For example, a sentence that has as output a probability distribution of 90% "pro", 5% "against", 5% "neutral" is more likely to be truly labelled as "pro" than a sentence with a probability distribution of 40% "pro", 30% "against", 30% "neutral". In our case, we chose a threshold of 80% to get a good balance between precision and recall. It means that a sentence labelled "pro" or "against" should have a probability score superior to 80% otherwise it will be labelled "neutral". We made a comparative study with different threshold values to optimize our F_β score.

As we can see in Table II, our models give high precision on the classification task. By using the threshold, we increased the precision for "pro" and "against" by at least 12% with ALBERT. The ALBERT model gives the best results for the F_β metric. We fine-tuned this model on the Biomass Corpus.

The results of our best model trained on the Biomass Corpus are presented in Table V. Thanks to the transfer learning, the model gives excellent results. In our study, we focused on the precision for the "pro" and "against" class as well as the recall of "neutral" class because we do not want to get non-argument. The high recall for "neutral" class can be explained by our filtering method with a threshold that keep sentences with a high confidence and consider the other as non-arguments. We present in Table IV some examples of arguments extracted with our model.

We need to mitigate our results with the fact that we worked on a small dataset for the evaluation and we present here the score on the validation dataset because we could not create a test dataset. Moreover, the number of sentences pro is really low.

V. Conclusion

In this paper, we proposed an approach to tackle the task of argument mining and classification in long documents. This method is based on the addition of paragraph retrieval step. This is an optimization to the traditional approach and allows the model to focus on the most important paragraphs of a document. This applies better to long documents which enables the identification of the core subject of interest and avoids paragraphs that are not pertinent. Thus, we can obtain better results in the classification stage.

We were able to create a paragraph retrieval model that keep most of the useful information found in a long document. For the argument mining and classification task, we were able to train a smaller model based on our dataset of interest with a high precision. Future works should focus on the

Table II: Results of each model on the UKP Corpus for the stance classification task

Model	F_β	P_{pro}	R_{pro}	$P_{against}$	$R_{against}$
outer-att [23]	0.3182	0.3651	0.1042	0.4696	0.2381
bilstm$_{BERT}$ [11]	0.3371	0.3431	0.1060	0.4397	0.4275
BERT-base [11]	0.5188	0.5048	0.4698	0.5313	**0.5795**
BERT-large [11]	0.5611	0.5535	**0.5051**	0.5843	0.5594
DistilBERT	0.5293	0.6851	0.2709	0.7254	0.2595
DistilRoBERTa	0.6253	0.6682	0.4392	**0.7320**	0.4385
ALBERT	**0.6351**	**0.6975**	0.4801	0.7042	0.4449

optimization of the paragraph retrieval task. In particular, we should work on the creation of better expert queries. The choice of words is crucial and can change radically the performance of the models. We should also create a bigger dataset for the evaluation of the full pipeline. We could also try other methods and create a model that extract arguments directly from paragraphs by giving the start and the end of the argument similarly to Question-Answering models.

REFERENCES

[1] Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. Neural end-to-end learning for computational argumentation mining. *CoRR*, abs/1704.06104, 2017.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[3] Raquel Mochales Palau and Marie-Francine Moens. Argumentation mining: the detection, classification and structure of arguments in text. In *The 12th International Conference on Artificial Intelligence and Law, Proceedings of the Conference, June 8-12, 2009, Barcelona, Spain*, pages 98–107. ACM, 2009.

[4] Ran Levy, Yonatan Bilu, Daniel Hershcovich, Ehud Aharoni, and Noam Slonim. Context dependent claim detection. In Jan Hajic and Junichi Tsujii, editors, *COLING 2014, 25th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, August 23-29, 2014, Dublin, Ireland*, pages 1489–1500. ACL, 2014.

[5] Ruty Rinott, Lena Dankin, Carlos Alzate Perez, Mitesh M. Khapra, Ehud Aharoni, and Noam Slonim. Show me your evidence - an automatic method for context dependent evidence detection. In Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 440–450. The Association for Computational Linguistics, 2015.

[6] Ran Levy, Shai Gretz, Benjamin Sznajder, Shay Hummel, Ranit Aharonov, and Noam Slonim. Unsupervised corpus-wide claim detection. In Ivan Habernal, Iryna Gurevych, Kevin D. Ashley, Claire Cardie, Nancy Green, Diane J. Litman, Georgios Petasis, Chris Reed, Noam Slonim, and Vern R. Walker, editors, *Proceedings of the 4th Workshop on Argument Mining, ArgMining@EMNLP 2017, Copenhagen, Denmark, September 8, 2017*, pages 79–84. Association for Computational Linguistics, 2017.

[7] Liat Ein-Dor, Eyal Shnarch, Lena Dankin, Alon Halfon, Benjamin Sznajder, Ariel Gera, Carlos Alzate, Martin Gleize, Leshem Choshen, Yufang Hou, Yonatan Bilu, Ranit Aharonov, and Noam Slonim. Corpus wide argument mining - a working solution. *CoRR*, abs/1911.10763, 2019.

[8] Marco Lippi and Paolo Torroni. Margot: A web server for argumentation mining. *Expert Syst. Appl.*, 65:292–303, 2016.

[9] Artem Chernodub, Oleksiy Oliynyk, Philipp Heidenreich, Alexander Bondarenko, Matthias Hagen, Chris Biemann, and Alexander Panchenko. TARGER: Neural argument mining at your fingertips. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 195–200, Florence, Italy, July 2019. Association for Computational Linguistics.

[10] Johannes Daxenberger, Benjamin Schiller, Chris Stahlhut, Erik Kaiser, and Iryna Gurevych. Argumentext: Argument classification and clustering in a generalized search scenario. *Datenbank-Spektrum*, 20(2):115–121, 2020.

[11] Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. Classification and clustering of arguments with contextualized word embeddings. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 567–578. Association for Computational Linguistics, 2019.

[12] Karen Sparck Jones, Steve Walker, and Stephen E. Robertson. A probabilistic model of information retrieval: development and comparative experiments - part 1. *Inf. Process. Manag.*, 36(6):779–808, 2000.

[13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[14] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.

[15] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A human generated machine reading comprehension dataset. In Tarek Richard Besold, Antoine Bordes, Artur S. d'Avila Garcez, and Greg Wayne, editors, *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016.

[16] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. Ms marco: A human generated machine reading comprehension dataset, 2018.

[17] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations, 2020.

[18] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

[19] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019.

[20] Christian Stab, Tristan Miller, Benjamin Schiller, Pranav Rai, and Iryna Gurevych. Cross-topic argument mining from heterogeneous sources. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3664–3674. Association for Computational Linguistics, 2018.

[21] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and

Table III: Results (nDCG score) of each model for the paragraph retrieval task on the Biomass Dataset

Model	Argument queries	Expert queries
BM25	**0.753**	**0.689**
MSMARCO$_{BERT}$	0.350	0.375
DPR	0.533	0.550

Table IV: Examples of arguments extracted with our model on the validation dataset

arguments	position
Labelling forest biomass as renewable has perverse impact on the climate	Against
To mitigate the climate crisis and meet its targets, the EU has utilized biomass; the biomass feedstock of forestry has become the main source for renewable energy in the EU and is a key part of the European Green Deal	Pro
In an essay for the company, Jenkins, a Yale-trained forester, wrote that biomass actually helps grow the forest because it creates a demand for low-value timber	Pro
That leads us to consider whether "paper" bottles, made of wood pulp from processing facilities, are truly any better for the environment than plastic bottles that, ultimately, also come from plants	Against

Table V: Results of ALBERT model on Biomass Corpus for the stance classification task

	Precision	Recall	# of true sentences
Neutral	0.76	0.99	216
Pro	0.92	0.33	40
Against	0.81	0.59	137

Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. *CoRR*, abs/1907.10902, 2019.

[22] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, October 2002.

[23] Christian Stab, Tristan Miller, and Iryna Gurevych. Cross-topic argument mining from heterogeneous sources using attention-based neural networks, 2018.