Constituency Tree Representation for Argument Unit Recognition

Samuel Guilluy
Univ Rennes 1 - IRMAR
Rennes, France
samuel.guilluy@univ-rennes1.fr

Florian Mehats
Ravel Technologies on leave from Univ Rennes
Rennes, France
florian.mehats@univ-rennes1.fr

Billal Chouli
Headmind AI & Blockchain
Paris, France
bchouli812@headmind.com

Abstract

The conventional method of extracting arguments from sentences solely relies on word proximity, disregarding the syntactic structure of the sentence. This approach often leads to inaccuracies, especially when identifying argumentative span boundaries. In this research, we investigate the benefits of utilizing a constituency tree representation of sentences to predict Argument Discourse Units (ADUs) at the token level. We first evaluate the effectiveness of utilizing the constituency tree representation for capturing the structural attributes of arguments within sentences. We demonstrate empirically that the constituency structure surpasses simple linear dependencies among neighboring words in terms of effectiveness. Our approach involves leveraging graph neural networks in conjunction with the constituency tree, adapting it specifically for argument unit recognition. Through extensive evaluation, our model outperforms existing approaches in recognizing argument units at the token level. Furthermore, we employ explainability methods to assess the suitability of our model architecture, providing insights into its performance.

1 Introduction

Argument identification within documents serves as the initial step in studying rhetorical speech processes, student essays, or political debates. The objective is to accurately identify Argument Discourse Units (ADUs), defined as minimal analysis units, within sentences, and predict their stance and relation to each other.

Previous works on token-level argument analysis (Trautmann, 2020) have employed language models such as Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019), in conjunction with probabilistic models like Conditional Random Field (CRF) (Lafferty et al., 2001). This combination enhances overall prediction coherence with constrained fine-tuning.

The study of arguments and discourse has been approached from a grammatical perspective, including frameworks such as Rhetorical Structure Theory (Mann and Thompson, 1987), one of the conclusion from the annotation guideline (Stede and Taboada) is the use of syntax to better identify the Elementary Discourse Units (EDU). Building grammatical parsers is a complex task that has received extensive research attention. The results achieved thus far are promising and can serve as a foundation for various applications.

In this research, we investigate the benefits of incorporating grammatical structure into a BERT-CRF model for argument unit recognition, with a specific focus on the constituency tree representation of sentences (as illustrated in Figure 1). This representation consists of a tree where interior nodes represents the grammatical structure of the sentence, along with leaf nodes (nodes without children) corresponding to the words within the sentence.

The primary objectives of this paper are:

- Evaluate the potential benefits of using the constituency tree for argument unit recognition and develop rules to modify the constituency tree into a structure better suited for identifying argument structure (Section 3).
- Assess the effectiveness of Graph Neural Network (GNN) models combined with a CRF layer in leveraging the syntactic information encoded in the constituency tree representation (Sections 4, 5, and 6).

2 Related Works

Argumentation Theory The precise definition of the concept of argument is an important step when creating dataset annotation rules. The identification of argument is strongly related to the discourse structure of the text and the identification

Proceedings of the 10th Workshop on Argument Mining, pages 35–44
December 7, 2023.

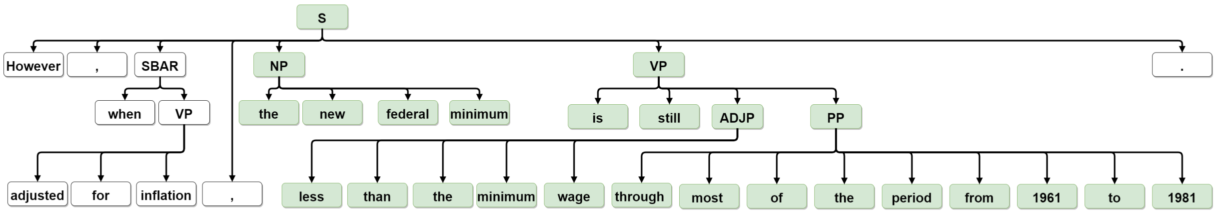

Figure 1: Constituency tree representation of the sentence, according to the Universal POS tags categories (where we limit the depth of the tree to 3): *"However, when adjusted for inflation, the new federal minimum is still less than the minimum wage through most of the period from 1961 to 1981."* from the AURC dataset. The green nodes represents words or spans with "PRO" label and the grey nodes represents words and spans with "NON" label.

of Elementary Discourse Units. As introduced in the Rhetorical Structure Theory (RST) by Mann and Thompson (1987), the Elementary Discourse Units (EDU) refers to a minimal unit of meaning within a larger discourse or conversation. It represents a self-contained piece of information that contributes to the overall structure and coherence of a discourse. A non-elementary Discourse Unit (DU) is called a complex discourse unit (CDU). The structure of a document is the set of linked DUs. As presented by Jo et al. (2019), Argument Discourse Unit (ADU) are units of meaning that contribute to the development and presentation of an argumentative structure. ADUs typically contain propositions, claims, evidence, or reasoning that support or challenge a particular standpoint or claim.

In practical applications, while certain studies rely on the annotator's judgment to determine the boundary of an ADU, many studies prefer to utilize a set of syntactic rules as a foundation (Stede and Taboada). This approach is favored because employing syntactic structure for annotating a sizable corpus at the token level is comparatively easier (Carlson and Marcu).

Tree Structure Representation in Natural Language Processing (NLP) Substantial evidence (Crain and Nakayama, 1987) supports the hypothesis that semantic interpretation of sentences by humans involves a tree-structured, hierarchical computation, where smaller constituents recursively combine into larger constituents, until we reach the full sentence.

In NLP, pioneer work (Gildea and Palmer, 2002) presenting the benefits of using constituency tree representation has failed to scale into production. According to the authors, this is caused by the lack of a reliable model to generate constituency tree representation of the sentences. However, recent promising results (Zhang, 2020) were made in consistency and dependency parsing.

Other papers have recently studied the use of tree structure to incorporate syntactic information to their models. Marcheggiani and Titov (2017) uses Graph Convolutional Networks (GCNs) based on the dependency tree structure of the sentence for semantic role labeling. Beck et al. (2018) uses GNNs for generation tasks from abstract meaning representation. Recently, Murty et al. (2022) demonstrate that for some tasks, transformers models become more "tree-like" over the course of training and in some cases unsupervisedly recovering the same trees as supervised parsers. Showing the importance of constituency tree in the learning process of the Transformers models.

Segmentation of argumentative units in texts has been explored in Ajjour et al.. The research indicates that both structural and semantic features are pivotal for segmenting argument units across various domains. However, within specific domains, semantic features stand out as the most effective for identifying the boundaries of these units.

3 Evaluation of the node similarity

In the subsequent section, we evaluate the effectiveness of utilizing the constituency tree representation for capturing the structural attributes of arguments within sentences. In Subsection 3.2, we introduce three metrics to assess the proximity of nodes in the tree concerning their argumentative label. Additionally, in Subsection 3.3, we propose modifications to the tree to enhance its suitability for argument recognition.

3.1 Experimental Setup

Our experimental setup involves the utilization of four argument datasets: ARG2020 (Alhindi and Ghosh, 2021), AURC (Trautmann et al., 2020), CDCP (Park and Cardie, 2018), and UKP (Stab and Gurevych, 2014) (detailed in subsection 5.1). These datasets share a common characteristic, as they are all annotated at the token level, meaning that each word in the sentences is assigned a label. To represent the sentences in the datasets as constituency trees, we employ the Berkeley Neural Parser (BENEPAR) (Kitaev and Klein, 2018), which is introduced in subsection 5.2. As a brief reminder, in the context of a constituency tree, a node without any children is referred to as a "leaf," while an "Interior Node (IN)" is a node that has child nodes.

As the labels for the interior nodes (IN) of the constituency tree were not initially provided, we made the decision to annotate these interior nodes following the same labeling rules utilized in the AURC (presented in 5.1) for sentence labeling. Our approach prioritizes the "no argument" label as less significant compared to the others, selecting the more predominant label among the remaining options. This strategy enables us to effectively learn the representation of IN nodes while ensuring consistency with the sentence-level labeling annotations.

3.2 Label proximity computation

One of the main advantages of adding a constituency tree to argument identification methods is the greater proximity of words that belong to the same grammatical class. In this section, we aim to validate the intuition that words belonging to the same grammatical class have more often the same label than words that are only neighbours in the sense of the linear representation of the sentence.

We have established three metrics to evaluate the suitability of employing the constituency tree representation for argument unit recognition. The three proportions computed, summarized in Table 1 and illustrated in Figure 2, are the following.

- **Leaf-Leaf similarity metric**: This refers to the ratio of nodes in a linear chain sentence (Table 1 column 3) representation that have both the same label and are adjacent to each other. In the cases where a constituency tree representation is available (Table 1 columns 4, 5, 6), we further narrow down this set of nodes to those that not only share the same label but also have the same parent node (illustrated in color red of Figure 2).
- **Leaf-IN similarity metric**: Only when a constituency tree representation is available (Table 1 columns 4, 5, 6), this indicates the proportion of leaf nodes that share the same label as their corresponding parent node (illustrated in color blue of Figure 2).
- **IN-IN similarity metric**: Only when a constituency tree representation is available (Table 1 columns 4, 5, 6), this measures the ratio of interior nodes that are connected by an edge and have the same label (illustrated in color orange of Figure 2).

The Leaf-Leaf metric tends to favor deeper trees, as deeper trees contain neighboring nodes that belong to finer grammatical categories and the same Argumentative Discourse Unit (ADU). However, an excessively deep tree is not desirable as it reduces the proximity between parent and child nodes. Metrics 2 and 3 are used to address this bias.

Indeed, regarding the Leaf-Leaf metric, we observe a stronger proximity between neighboring words within the same grammatical class compared to neighboring words when the grammatical structure is not considered. Additionally, the constituency tree with a maximum depth of 4 exhibits greater node similarity than the tree with a maximum depth of 2 or 3. As for the other two metrics, when the tree becomes too deep, the distance between words of the same grammatical class may become longer than that between words of different grammatical classes. This leads us to impose a limit on the maximum allowed tree depth. Setting a depth cap at 4 may not necessarily be the best choice, as the constituency tree with a maximum depth of 3 demonstrates better results concerning grammatical class similarity. In conclusion, these findings prompt us to experiment with a maximum depth of 3 for our models.

3.3 Fine grained stats

In this section, we explore the possibility of transforming the constituency tree to better align it with grammatical structures, with the aim of reducing tree complexity while maintaining its ability to segment into Argumentative Discourse Units (ADUs). To achieve this, we consider the grammatical class of nodes and identify those that exhibit higher coherence with the ADU segmentation. In practical terms, this involves examining the grammatical labels of linked nodes to determine whether parent and child nodes share the same label or differ in nature.

Metrics	Dataset	No tree	Depth 2	Depth 3	Depth 4
With Constituency tree					
Leaf-Leaf similarity	ARG2020	95.4 %	97.9%	98.2 %	**98.6 %**
	AURC	97.1 %	98.3%	98.4%	**98.6 %**
	CDCP	97.8 %	99.6 %	99.7 %	**99.7 %**
	UKP	91.9 %	97.9%	97.9%	**98.3 %**
Leaf-IN similarity	ARG2020	//	91.8%	**92.9 %**	59.2%
	AURC	//	90.2%	**91.3**%	84.2%
	CDCP	//	98.1%	**98.7 %**	68.9%
	UKP	//	89.2 %	**89.7 %**	49.8%
IN-IN similarity	ARG2020	//	91.7%	**95.1%**	88.3%
	AURC	//	88.5%	**93.3**%	92.9%
	CDCP	//	96.4%	**97.8%**	90.4%
	UKP	//	85.1 %	**92.1%**	84.3%
With reduced Constituency tree					
Leaf-Leaf similarity	ARG2020	//	//	98.2 %	**98.6**%
	AURC	//	//	98.4%	**98.6**%
	CDCP	//	//	**99.8**%	99.7%
	UKP	//	//	97.8%	**98.3**%
Leaf-IN similarity	ARG2020	//	//	**93 %**	59.2%
	AURC	//	//	**91.4%**	84.2%
	CDCP	//	//	**98.8%**	68.7%
	UKP	//	//	**90%**	50%
IN-IN similarity	ARG2020	//	//	**95 %**	88.3%
	AURC	//	//	**93.4%**	92.9%
	CDCP	//	//	**97.8%**	90.1%
	UKP	//	//	**92.1%**	84.4%

Table 1: Assessment of three measures to evaluate the suitability of the constituency tree representation. The first section of the table examines the evaluation of the constituency tree with varying the maximum depths allowed for the tree, while the second section focuses on the assessment of the tree reduction method from Subsection 3.3.

Table 2 provides a snapshot of the statistics observed in the datasets for the most common grammatical classes present in the constituency trees. We observe significant differences across the studied datasets. The argumentation structure in the AURC and CDCP datasets align more closely with the syntactic structure, compared to the UKP and ARG2020 datasets. This is consistent with the fact that the CDCP and AURC datasets are both online feedback datasets and UKP and ARG2020 are both student essays datasets.

When a particular constituency class consistently shares the same grammatical label as its children, it indicates coherence with the grammatical structure. In such cases, our reduction method involves simplifying the tree structure by grouping all its children and removing the intermediate interior node. In practice, we establish a threshold. If the ratio of identical labels exceeds this threshold as outlined in Table 2, we simplify the structure at this level. This adjustment reduces tree complexity while preserving the fact that words sharing the same parent node are more likely to have the same grammatical label.

For a tangible illustration based on Table 2, consider the AURC dataset where the tag "NML", representing nominal modifiers, has a ratio of 97%. This indicates that 97% of the time, its child elements bear the same labels. Given that a nominal modifier is a noun that adjusts another noun (effectively functioning as an adjective) it makes sense for them to share the same labels. Therefore, simplifying the structure to retain only the parent node "NML" and treating all the leaf nodes below it as its direct children appears to be an effective strategy.

The latter part of Table 1 illustrates the updated proximity statistics after the tree transformation. We observe that the three metrics are preserved

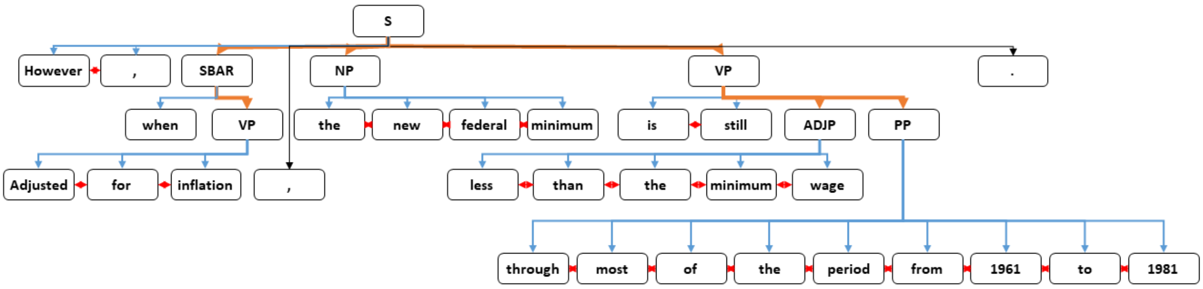

Figure 2: Visualization of the label proximity metrics on the constituency tree representation of the sentence: *"However, when adjusted for inflation, the new federal minimum is still less than the minimum wage through most of the period from 1961 to 1981."* from the AURC dataset. The blue arrows represent the edges analyzed for the Leaf-IN metric, the orange arrows for the IN-IN metric, and the red arrows for the Leaf-Leaf metric.

across all four datasets. We can thus reduce the complexity of the constituency tree in order the accelerate the training process of the models while hoping to preserve its capacity. We will evaluate this assumption in Section 5 and 6.

4 Presentation of our Model

In this section, we present a detailed overview of the architecture and components of our proposed model for argument unit recognition.

4.1 Baseline: Linear chain approach

The reference model, to which our model will be compared, has been introduced by Trautmann et al. (2020). It is composed of two modules. In the first module, the sentence is tokenized following the BERT tokenizer and the BERT model is fine-tuned for token classification, where the output of the last layer matches the number of classes of the dataset. In the second module, a linear chain Conditional Random Field (Lafferty et al., 2001) is applied to estimate the probability of each label class. The main intuition of this model is to leverage the BERT LLM "semantic knowledge" and then to improve the results by incorporating a linear chain dependency structure for the syntactic part. This takes advantage of neighbours dependency relations between words. The good results of this model lead us to use it as a competitive benchmark for our approach based on constituency trees as input representations of sentences.

4.2 Our model: Graph Neural Network approach

A major difficulty in choosing a graph neural layer architecture is that each sentence has a different tree representation. Hence, the model needs to be agnostic to the lack of completeness of the tree structures from the dataset. The message passing design enables to share the model weights among the network nodes, thus the results do not depend on the upfront global tree structure access. The Graph Attention Layer (GAT) (Veličković et al., 2018) allows to combine the attention mechanism with the graph structure in a message passing design, preserving the syntactic structural information of the sentence. In order to improve the model stability, adding multi-head attention layers is beneficial to the training step. The different heads are then aggregated in order to provide the next hidden states of the neural network. To leverage the dependency structure of the sentence, we integrate a multi-layer GAT (Graph Attention Network) model between the BERT module and the linear chain CRF. For the CRF, we use the implementation from (Gardner et al., 2017), which was present in the baseline model. The idea behind this architecture is the following. The GAT model outputs the probability of each label for each node in the graph. When subsequently employing a linear chain CRF, we retain only the leaf nodes to represent the sentence in a traditional linear chain format. As illustrated in Figure 3, first, the BERT language model outputs the sentence hidden representation. Next, the information is spread to the graph neighbours at each iteration. In that way, we expect to reach a better consistency between neighbour nodes when we train on a restricted dataset.

5 Experimental Setup

In the next sections, we present a comprehensive evaluation of our proposed model for argument unit recognition using constituency tree representations and GNNs with a CRF layer. We compare the performance of our model against existing approaches and analyze its effectiveness in capturing syntactic

Dataset	Parent node type	Number of same labels	Number of different labels	ratio
ARG2020	VP	25710	13911	65 %
	NP	18414	9767	65 %
	S	28169	14325	66%
	PP	9744	4097	68 %
	SBAR	8545	5099	63 %
	ADJP	1293	753	63 %
	NML	182	101	64%
	ADVP	344	211	62 %
AURC	VP	30342	3274	90 %
	NP	29792	2279	93 %
	S	28202	9181	75 %
	PP	12878	1173	92 %
	SBAR	9474	2517	79 %
	ADJP	2316	269	90 %
	NML	638	22	97 %
	ADVP	411	58	88 %
CDCP	VP	4112	1623	72 %
	NP	2608	840	76 %
	S	3960	1371	74 %
	PP	1253	386	76 %
	SBAR	1607	725	69 %
	ADJP	202	49	80 %
	NML	48	13	79 %
	ADVP	81	27	75 %
UKP	VP	25266	15789	61 %
	NP	17147	12534	57 %
	S	29179	23857	55 %
	PP	8952	7265	55 %
	SBAR	8210	6498	56 %
	ADJP	1400	1276	52 %
	NML	106	94	53 %
	ADVP	519	440	54 %

Table 2: Extracts from the metrics of Evaluation of fine grained stats. We present only the parent nodes that appear most frequently in the training dataset.

information from the constituency tree.

First, we describe the experimental setup and datasets used for evaluation

5.1 Data Source

ARG2020 (Alhindi and Ghosh, 2021) is an argument mining corpus annotated with argumentative structure composed of "claims" and "premises". It is composed of 145 English argumentative essays selected through the Writing Mentor Educational App. It is based on middle school students writing. The claims is defined as a potentially arguable statement that indicates a person is arguing for or arguing against something. The premises is defined as the reasons given by either for supporting or attacking the claims.

Argument Unit Recognition and Classification (AURC) (Trautmann et al., 2020) is a corpus for argument mining that includes annotations for argumentative structure information, capturing the polarity of arguments on a given topic. The corpus consists of 8000 sentences, evenly distributed across 8 topics. The authors distinguished between PRO (supporting), CON (opposing) arguments, and NON (non-argumentative) words for each topic, in order to construct sentence-level labels. Their labeling rule is as follows: if only NON words occur, the sentence is labeled as NON. If both NON and

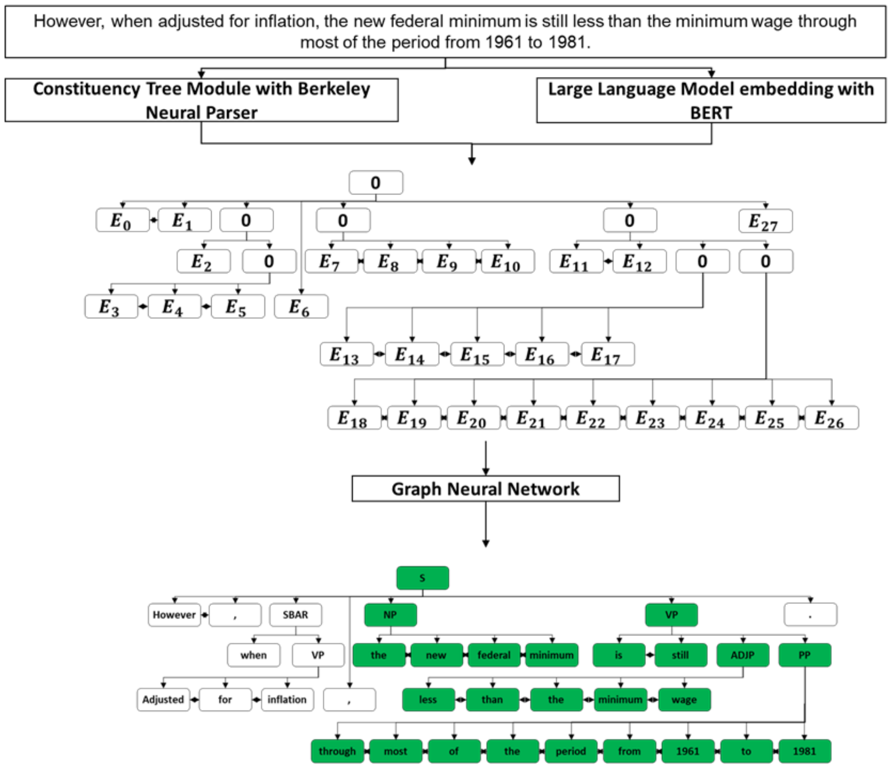

Figure 3: Illustration of the model architecture on an example of a sentence. We present the three distinct modules composing the model with their associated input/output. The colored node after the Graph Neural Network refers to the nodes where the label predicted in "PRO", this phrase is taken from the AURC dataset. The E_x refers to the embeddings from the BERT model, the interior nodes are initialized with the 0 vector.

only PRO (or only CON) words occur, the label PRO (or CON) is assigned. If both PRO and CON words occur, the label that appears more frequently is assigned. In (Trautmann et al., 2020), the authors distinguish between "in-domain" settings, where the domain of the arguments is present in both the training and test datasets, and "cross-domain" settings, where the domains in the test dataset are not found in the training dataset. In our experiments, we focus solely on the "in-domain" scenario.

The Cornell eRulemaking Corpus (CDCP) (Park and Cardie, 2018) is a corpus for argument mining that includes annotations for argumentative structure information, specifically capturing the evaluability of arguments. The corpus comprises 731 user comments on the Consumer Debt Collection Practices rule issued by the Consumer Financial Protection Bureau. The resulting dataset contains a total of 4931 elementary unit annotations and 1221 support relation annotations.

Argument Annotated Essays corpus (UKP) consists of a collection of persuasive essays gathered by (Stab and Gurevych, 2014). This essay corpus is equipped with annotations of argument components at the clause level, as well as argumentative relations. Specifically, it includes annotations for major claims, claims, and premises, which are interconnected through argumentative support and attack relations. The corpus was annotated by three raters, achieving an inter-annotator agreement of alpha = 0.72 for argument components and alpha = 0.81 for argumentative relations. In total, the corpus consists of 90 essays containing 1673 sentences.

The models are trained individually on each of the four datasets, conforming to the respective label schemes they offer. For comparison with the baseline, we adhere to the train-test splits presented in the original datasets' experiments when available. In cases where these are not provided, we employ a random sampler to allocate 20% of the sentences for testing and 80% for training. Subsequently, the sentences are segmented into chunks of 64 tokens each.

5.2 Constituency tree construction

One of the main advantages of incorporating a constituency tree into traditional methods is the

	Test Intervals	Best values	Relative parameters importance
Learning rate	10^{-5} to 10^{-3}	$2.8 \cdot 10^{-5}$	30 %
Maximum gradient allowed	10^{-1} to 10^{2}	9.7	49 %
Number of GAT layers	1 to 3	2	2 %
Number of unit per GAT layers	50 to 300	290 and 100	2 %
Number of heads per GAT layers	1 to 3	3 and 3	7 %
Number of linear layers	1 to 3	2	5 %
Number of unit per linear layers	50 to 250	100 and 100	5 %

Table 3: Feature importance of the BERT-GAT-CRF model with Constituency Tree evaluate on the AURC evaluation dataset.

	AURC	CDCP	ARG2020	UKP
BERT	68 %	80 %	75 %	81 %
BERT - CRF	69 %	81 %	75.5 %	81.6 %
BERT - GAT	64 %	75.5 %	75.2 %	79.3 %
BERT - GAT - CRF with Constituency Tree	72.8 %	81.5 %	**76.1 %**	**82.8 %**
BERT - GAT - CRF with Reduced Constituency Tree	**73.2 %**	**83.1 %**	75.9 %	81.4 %

Table 4: F1-score of the different models at token level on the test dataset.

increased proximity of words belonging to the same grammatical class compared to words that are merely adjacent in a linear sentence representation. This can be further illustrated by referring to the constituency tree depicted in Figure 1. In this sentence, the distribution of ADUs aligns with the grammatical structure of the sentence. For instance, although the words "inflation" and "the" are neighboring words in the sentence, they are positioned further apart in the constituency tree structure. This leads to improved identification of boundaries between ADUs.

For our preprocessing step, we employed a neural network model called the Berkeley Neural Parser (BENEPAR) (Kitaev and Klein, 2018), which has been trained on 11 different languages and is available with Spacy and works with GPUs. We utilized the weights provided by the model's development team for our experimentation.

5.3 Hyperparameters Optimization

The BERT-GAT-CRF model has a significantly larger number of hyperparameters compared to the BERT-CRF model. This is primarily attributed to the extensive hyperparameters associated with the GAT, such as the number of layers, units per layer, and number of heads. To determine the optimal hyperparameters for this model, we employed the Optuna library (Akiba et al., 2019). Optuna is a framework specifically designed for efficient hyperparameter optimization. To evaluate the relative importance of different hyperparameters in our model, we conducted experiments on the AURC dataset and presented the results in Table 3. Notably, we observed that the most influential hyperparameters are the learning rate and the maximum gradient value allowed. Empirically, we found that unconstrained gradients led the model to converge to a local optimum, where the label "NO" was assigned to every word. This local optimum emerges due to the dataset's imbalance, which tends to favor the absence of arguments.

6 Models Evaluation

6.1 Results Presentation

While the original paper by Trautmann et al. (2020) introduced metrics such as token level, span level, and sentence level, our focus lies primarily on improving argument border recognition rather than argument stance identification. Consequently, our model excels in token-level performance, showcasing superior results. However, our model achieves comparable outcomes at the sentence and span levels.

The results pertaining to token-level classification are outlined in Table 4. In accordance with the insights from Table 1, we computed our models with a maximum depth of 3. Our best-performing model consists of BERT-GNN-CRF with Reduced Constituency Tree. These outcomes highlight the significant advancements achieved by our proposed model in argument unit recognition. By leveraging the constituency tree representation, integrating GNNs and CRF, and incorporating reduced constituency trees, our model excels in capturing the intricacies of argument structures.

7 Conclusion

In conclusion, this research study introduces a novel method for identifying the boundary of ADUs using the sentence constituent tree representation. Our model effectively spreads information across the graph and achieves promising results on a small dataset.

Previously identified errors in these datasets include the incorrect recognition of argumentative segment spans and inaccurate classification of stances. In this study, we focus on improving the span detection problem and successfully enhance the method for identifying ADU boundaries.

However, it is worth noting that some argument mining datasets does not strictly adhere to grammatical correctness, as noted in (Trautmann et al., 2020). This limitation arises from sentences where subjects are absent, which hampers the performance of models relying solely on grammatical structure. This issue could be resolved by devising annotation rules that more strictly align with the syntactic structure of sentences. Furthermore, the second type of error, which pertains to position identification, is primarily attributed to the limitations of the BERT model. The dataset only provides sentences with a maximum length of 64, thereby restricting the available context for ADUs and impeding our model's capability. Many arguments require deeper domain knowledge to fully comprehend the underlying issues.

References

Yamen Ajjour, Wei-Fan Chen, Johannes Kiesel, Henning Wachsmuth, and Benno Stein. Unit segmentation of argumentative texts. In *Proceedings of the 4th Workshop on Argument Mining*, pages 118–128. Association for Computational Linguistics.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pages 2623–2631, New York, NY, USA. Association for Computing Machinery.

Tariq Alhindi and Debanjan Ghosh. 2021. "Sharks are not the threat humans are": Argument Component Segmentation in School Student Essays. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 210–222, Online. Association for Computational Linguistics.

Daniel Beck, Gholamreza Haffari, and Trevor Cohn. 2018. Graph-to-Sequence Learning using Gated Graph Neural Networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 273–283, Melbourne, Australia. Association for Computational Linguistics.

Lynn Carlson and Daniel Marcu. Discourse Tagging Reference Manual. page 87.

Stephen Crain and Mineharu Nakayama. 1987. Structure Dependence in Grammar Formation. *Language*, 63(3):522–543. Publisher: Linguistic Society of America.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Daniel Gildea and Martha Palmer. 2002. The Necessity of Parsing for Predicate Argument Recognition. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 239–246, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Yohan Jo, Jacky Visser, Chris Reed, and Eduard Hovy. 2019. A Cascade Model for Proposition Extraction in Argumentation. In *Proceedings of the 6th Workshop on Argument Mining*, pages 11–24, Florence, Italy. Association for Computational Linguistics.

Nikita Kitaev and Dan Klein. 2018. Constituency Parsing with a Self-Attentive Encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2676–2686, Melbourne, Australia. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

William Mann and Sandra Thompson. 1987. *Rhetorical Structure Theory: A Theory of Text Organization*.

Diego Marcheggiani and Ivan Titov. 2017. Encoding Sentences with Graph Convolutional Networks for Semantic Role Labeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1506–1515, Copenhagen, Denmark. Association for Computational Linguistics.

Shikhar Murty, Pratyusha Sharma, Jacob Andreas, and Christopher D. Manning. 2022. Characterizing Intrinsic Compositionality in Transformers with Tree Projections. ArXiv:2211.01288 [cs].

Joonsuk Park and Claire Cardie. 2018. A Corpus of eRulemaking User Comments for Measuring Evaluability of Arguments. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Christian Stab and Iryna Gurevych. 2014. Annotating Argument Components and Relations in Persuasive Essays. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1501–1510, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Manfred Stede and Maite Taboada. Annotation Guidelines for Rhetorical Structure.

Dietrich Trautmann. 2020. Aspect-Based Argument Mining. In *Proceedings of the 7th Workshop on Argument Mining*, pages 41–52, Online. Association for Computational Linguistics.

Dietrich Trautmann, Johannes Daxenberger, Christian Stab, Hinrich Schütze, and Iryna Gurevych. 2020. Fine-Grained Argument Unit Recognition and Classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9048–9056. Number: 05.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks.

Meishan Zhang. 2020. A Survey of Syntactic-Semantic Parsing Based on Constituent and Dependency Structures. *arXiv:2006.11056 [cs]*. ArXiv: 2006.11056.