External Knowledge-Driven Argument Mining: Leveraging Attention-Enhanced Multi-Network Models

Anonymous ACL submission

Abstract

Argument mining involves the identification of the argument relations (AR) between pair of propositions. Often, the AR between propositions depends on contextual factors, although this context is not always overtly stated within the propositions. While humans possess the ability to infer the requisite background context for linking propositions, machines encounter challenges in doing so when such contextual cues are not explicitly provided. Establishing connections between propositions relies on maintaining a coherent flow of interconnected ideas, often centered around discussed entities, topics, or themes. This study aims to leverage external resources, including WordNet, ConceptNet, and Wikipedia, by aligning the entities within propositions with these external resources to identify the interconnected flow of ideas by analysing the chain of semantic relations linking these entities. To effectively extract relevant information from these chains, we propose integrating attention mechanisms into Multi-Network architectures. This integration allows the model to focus not only on the information presented in the propositions but also on the chain of entities, thereby enhancing the AR identification process. We evaluate various configurations of the architecture on two ontological resources (WordNet, ConceptNet) and Wikipedia, achieving a new state of the art performance with F-measure scores of 0.85, 0.84, and 0.71 respectively on three diverse datasets.

1 Introduction

Argument mining involves identifying the argumentative structure within a text, which includes segmenting arguments into elementary discourse units (EDUs), distinguishing argumentative units from non-argumentative ones, classifying propositions (such as premise and claim), labeling argument relation (AR) between propositions, and identifying argument schemes (Persing and Ng, 2016; Stab and Gurevych, 2017; Lawrence and Reed, 2020). In this study, we focus on classifying the AR between propositions into three categories: Inference (when one proposition supports the other), Conflict (when one proposition attacks the other), and None (when there is no AR between the propositions).

The AR existing between propositions is dependent on context (Potash et al., 2017; Habernal et al., 2017; Choi and Lee, 2018; Rinott et al., 2015) though such context is not always explicitly expressed at discourse level. Consider the propositions extracted from the 2016 presidential election debate corpus (Visser et al., 2019) presented in Table 1. To identify the AR between (1) and (2), one is expected to know that "NAFTA agreement" is related to "USA", whereas the relation between (4) and (5) is based on the world knowledge allowing one to infer that "building electric grid" is an "economic activity". Such contexts are trivial for human experts and yet present a challenge for a computer.

Establishing the connections between propositions relies on maintaining a coherent flow of interconnected ideas. This cohesion is often centered around discussed entities, topics, or themes, a concept commonly referred to as **Local coherence** (Foltz et al., 1998; Marcu, 2000). Local coherence plays a crucial role in facilitating the smooth transitions of ideas between propositions by recognising the inherent regularities in entity distribution. Similarly, other entity-based theories of discourse (Givón, 1987; Prince, 1981) and **Centering Theory** (Grosz et al., 1995) propose that these regularities contribute to the coherence of discourse by guiding the organisation of ideas around salient entities. Following similar framework, aspect based argument mining techniques employ such fine-grained elements of propositions to identifying argument structure (Misra et al., 2017; Dragoni et al., 2018; Gemechu and Reed, 2019; Trautmann, 2020).

No	Propositions
1	[USA]$_C$ [is in deep trouble]$_{OC}$
2	[NAFTA agreement]$_C$ [is defective]$_{OC}$
3	[We]$_C$ [can have]$_{OC}$ [clean energy]$_A$
4	[We]$_C$ [can build]$_{OC}$ a new modern [electric grid]$_A$
5	[This]$_C$ [is a lot of]$_{OC}$ new [economic activity]$_A$

Table 1: Examples from 2016 presidential election debate corpus (Visser et al., 2019) to illustrate the relation between the functional components of propositions. C represents the theme of the sentence, A represents the aspects specialising the theme, while the opinion on C is represented by OC.

However, it's important to note that the interconnections between these entities are not always explicit; often, inferred based on contextual cues and background knowledge. For example, the AR between propositions (3) and (4) is straight-forward since the relation between the entities mentioned in the respective propositions can be directly obtained from ontology (Miller, 1995; Speer et al., 2017) ("Electric grid; grid" is directly related to "power; electrical power" in WordNet (Miller, 1995)) or via the similarity between their embedding (Pilehvar et al., 2013; Le and Mikolov, 2014; Reimers and Gurevych, 2019). However, identifying the AR between (4) and (5) is demanding since the relation between their entities is not trivial in any of the existing knowledge resources. Actually, the path linking "electric grid" to "economic activity" is missing in WordNet (Miller, 1995) or ConceptNet (Speer et al., 2017) or DBpedia[1]. "Electric grid" is not directly related to "economic activity", yet both concepts are indirectly related via the chain of concepts interlinked using a set of relation types: "economic activity" *involves* "innovation" *which constitutes* developing "clean energy" *transmitted by* "electric grid". The objective of this study is to identify and leverage such chains of entity transitions from external resources, to capture implicit referential information between propositions (Asher and Lascarides, 2003). This information is then used to predict AR between propositions (premise and conclusion) into the three classes: inference, conflict, and none.

Pre-trained language models (LMs) have revolutionised Natural Language Processing (NLP), shifting the paradigm from traditional feature and architecture engineering to more data-driven approaches. They encode various forms of knowledge implicitly in their parameters, encompassing world knowledge in the form of relational, commonsense, and linguistic knowledge (Petroni et al., 2019; Goldberg, 2019; Safavi and Koutra, 2021; AlKhamissi et al., 2022). While such innate access to knowledge has propelled LMs to achieve remarkable results across numerous downstream tasks in NLP (Devlin et al., 2018; Wang et al., 2018a; Xu et al., 2019; Zhang et al., 2019; Reimers et al., 2019; Joshi et al., 2020), their ability to encode background knowledge required to identify AR is questionable (Kassner and Schütze, 2019).

Leveraging knowledge from external resources has been shown to improve the performance in argument mining (Kobbe et al., 2019; Botschen et al., 2018; Fromm et al., 2019; Plenz et al., 2023) and related tasks, such as enhancing semantic plausibility (Wang et al., 2018b), identifying inferences (Chen et al., 2017), and determining entailment (Glockner et al., 2018). However, these studies uniformly rely on structured knowledge bases alone, without addressing the challenge of integrating external information into model architectures. Additionally, some of these works employ conventional feature engineering for encoding external knowledge (Kobbe et al., 2019), while others utilise ranking algorithms to assign weights to paths (Plenz et al., 2023).

In this paper, we propose traversing Wikipedia, WordNet, and ConceptNet to identify the chain of paths connecting Wikipedia pages, WordNet Synsets, or ConceptNet Synsets associated with the entity mentions in the pair of propositions. These paths are then utilised to identify ARs. For instance, to link (4) and (5), we identify the paths linking "economic activity" to "electric grid" via the chain of hyperlinks interconnecting Wikipedia pages of: "economic activity" $\rightarrow$ "innovation" $\rightarrow$ "clean energy" $\rightarrow$ "electric grid". AR between propositions is identified by attending to such paths in addition to the explicit information provided in the propositions. We introduce attention mechanisms into the Multi-Network architecture (Triplet and Siamese) to attend to the external knowledge relevant to the AR between the proposition pairs.

We experimented with several baseline models to compare the background knowledge embedded in the recent LMs in both zero-shot and fine-tuning configurations. In the zero-shot setups, we prompted generative LMs to produce paths between components and use them to predict ARs and compare against methodologies leveraging external knowledge bases. Furthermore, we fine-

[1] https://wiki.dbpedia.org/

tuned LMs both in the standard sequence classification and Multi-Network setup for comparisons. The multi-network models surpassed the baselines, while the models incorporating external resources outperformed those without them, achieving a new state-of-the-art performance.

2 Methodology

2.1 Data

We use three corpora. The first is AAEC (Stab and Gurevych, 2017) which has a total of 402 arguments. Propositions under each argument are labelled as premise, claim or major claim. The corpus has 147,271 tokens, 6,089 propositions and 3832 ARs.

The second corpus is the Argumentative Micro Text (AMT) (Peldszus and Stede, 2013) which is a collection of 112 short texts collected from human subjects in German and translated into English. It is annotated following the argumentation structure outlined by (Peldszus and Stede, 2013). The structure consists of a central claim, and supporting propositions. It has a total of 8007 tokens, 576 propositions and 272 ARs.

We have also used part of the US 2016 presidential election debate corpus (US2016G1tv) (Visser et al., 2019) which is annotated based on the AIF standard (Chesnevar et al., 2006) using the OVA+ annotation tool (Janier et al., 2014)[2] and stored in the AIFdb database (Lawrence et al., 2015). The corpus has a total of 15805 tokens, 1473 propositions and 505 ARs.

2.2 External Knowledge Alignment and Paths Extraction

We use the information explicitly provided in a given argument as local context and the knowledge obtained from external resources as external context. We adopt the argument representation proposed by Gemechu and Reed (2019), which decomposes propositions into four functional components: target concepts (***C***), aspects (***A***), opinion on ***C*** (***OC***) and opinion on ***A*** (***OA***). ***C*** is a set of concepts representing the topic of a proposition and ***A*** is the set of concepts specialising the topic (see Table 1). The external context relevant to identify the AR is often assumed to be generally known. In this work, we aim to obtain such information from two ontological resources, WordNet (Miller, 1995) and ConceptNet (Speer et al., 2017), as well as a semi-structured resource, Wikipedia, as detailed in Sections 2.2.1 to 2.2.2.

[2] http://ova.arg-tech.org

2.2.1 Ontology as External Source

Drawing upon the components of the proposition pairs identified through the technique elucidated in Section 3.2 and evaluated in Section 3.3, we traverse the WordNet (Miller, 1995) and ConceptNet (Speer et al., 2017) Synset hierarchies and align the components of the propositions with the Synsets, to identify the chain of Synsets that connects the functional components of the propositions. This alignment depends on the similarity of embeddings generated by Sentence-transformers (Reimers and Gurevych, 2019) for the components and Synsets. Starting with one of the components, we search the knowledge graphs until the other component is obtained or the level (depth) of the search reaches a threshold $\alpha = 5$. If the search is successful, we concatenate the Synsets and the type of semantic relation between.

2.2.2 Wikipedia as External Source

We also traverse Wikipedia to identify the chain of Wikipedia pages linking the functional components of propositions. Breadth first search is used to identify all the possible Wikipedia paths connecting the components. Two types of information are identified along the paths: (a) Wikipedia page titles, and (b) the relationship types denoting the connections between these page titles and the hyperlinks linking to the subsequent page in along the paths.

Given a pair of components (c_1,c_2) addressed by a pair of propositions (p_1,p_2), we first align the components with the Wikipedia pages. This alignment is achieved by computing the cosine similarity between the embeddings of the Wikipedia page titles and the components, which are generated using Sentence-transformers (Reimers and Gurevych, 2019). Considering Wikipedia as a graph (with pages as nodes and hyperlinks as edges), we begin a breadth-first search from the Wikipedia page of one concept (c_1), continuing until we locate the second concept (c_2) or reach a depth threshold, $\alpha = 5$. During this search, we record sentences containing Wikipedia page titles (hyperlinks) of the current page and hyperlinks leading to the next page along the path. These sentences contribute to the formation of a tuple: $\langle hl_1, hl_2, keywords \rangle$, where the keywords represent the set of words linking the pair of hyperlinks (hl_1, hl_2) within the sentences. We leverage the structured information offered by

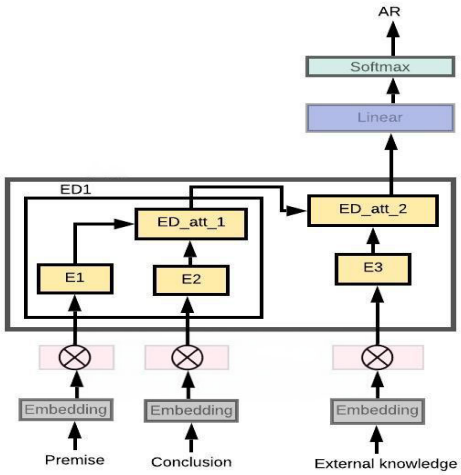

Figure 1: Triplet-networked with attention layers.

the semantic role labeling system implemented in AllenNLP[3], rather than examining every word between the hyperlinks. Subsequently, the hyperlinks are replaced with the constants "hl_1", "hl_2" based on their order within the preceding sentence. Finally, a frequency count is used to rank the tuples, and the top "***m***" are considered.

2.3 Model

We frame the task as sequence alignment (the alignment between propositions and external knowledge paths) and propose to use Attention-Based Multi-Network for learning the alignment (Section 2.3.1). Additionally, the details of the models that utilise LMs alone as sources of background knowledge is presented in Section 2.3.2.

2.3.1 Attention-Based Multi-Network

We investigate two attention based multi-network configurations, namely Siamese and Triplet (Schroff et al., 2015) networks. Initially, we utilise the Siamese network, where one sub-network encodes the propositions while the other encodes the external information. Furthermore, we examine the Triplet network, which encodes the individual propositions and external resources separately.

Siamese Architecture with Attention. In this setup, given the Siamese network comprising two sub-networks, denoted as **E1** and **E2**, an attention mechanism is applied to the outputs of these sub-networks to predict AR. **S1** processes the concatenation of a premise and conclusion, while **S2** handles the concatenation of the information from external resources. The attention mechanism, denoted as **ED-att-1**, attends to the external resources relevant to the AR between the premise and conclusion. Accordingly, the output of (**E1**) serves as the query, while the output of (**E2**) is used as keys and values, enabling to query the external information relevant to the premise and conclusion. It employs multi-head attention, outlined as follows:

$$\mathbf{e}_i = \text{softmax}(\frac{1}{\sqrt{d_k}} \sum_{j=1}^{h} \mathbf{e}_i^j) \quad (1)$$

where h represents the number of attention heads. Each head j computes scaled dot-product attention using query $\mathbf{Q}^j$, key $\mathbf{K}^j$, and value $\mathbf{V}^j$ matrices, which are linear transformations of the input hidden state $\mathbf{h}_i$:

$$\mathbf{Q}^j = \mathbf{W}_Q^j \cdot \mathbf{h}_i,$$
$$\mathbf{K}^j = \mathbf{W}_K^j \cdot \mathbf{h}_i,$$
$$\mathbf{V}^j = \mathbf{W}_V^j \cdot \mathbf{h}_i,$$

where $\mathbf{W}_Q^j$, $\mathbf{W}_K^j$, and $\mathbf{W}_V^j$ are learnable weight matrices specific to each attention head. The final attention weight $\mathbf{e}_i$ is obtained by summing over all attention heads and applying the softmax function. The resulting attention weights are then combined with the output of **E1** and passed through a fully connected classification layer, for AR classification. This fusion enables the model to integrate the original representations of the premise and conclusion with the extracted external information (detailed model parameters are provided in Section A.3 of the supplementary material).

Triplet Network Architecture with Attention. In contrast to the Siamese Architecture, the Triplet Network Architecture consists of three sub-networks: **E1**, **E2**, and **E3**. Sub-networks **E1** and **E2** encode the premise and conclusion, respectively, while **E3** encodes the external knowledge connecting them. Two cross-attention layers are introduced (**ED-att-1** and **ED-att-2**). **ED-att-1** is utilised to focus on the relation between the premise and conclusion, where the output of **E1** serves as queries and the output of **E2** is used as keys and values. On the other hand, **ED-att-2** attends to the external knowledge relevant to the AR between the premise and conclusion. Specifically, the output of **ED-att-1** acts as the query, while the output of **E3** is used as keys and values. For the classification layer, a similar fusion strategy to the Siamese architecture is employed, but in

[3]

this case, the fusion is based on the output of the two attention layers. The rationale behind this approach is that **ED-att-1** encodes the relation between the premise and conclusion, while **ED-att-2** encodes the relevant external resource, enabling the model to effectively leverage both the relationship between the premise and conclusion and the relevant external knowledge for AR classification (see Figure 1). Detailed model parameters and configuration can be found in the supplementary material provided in Section A.3.

2.3.2 Leveraging LMs as Background Knowledge Sources

To explore the effectiveness of LMs in leveraging background knowledge for AR identification, we examine two setups: zero-shot and fully fine-tuning.

In the zero-shot setup, we leverage GPT-4[4], a generative LM, to generate paths between the components of propositions. These paths are then used as external knowledge for AR classification. Additionally, we prompted GPT-4 to predict ARs without explicitly providing the paths. For the fine-tuning setup, we fine-tune BERT (Devlin et al., 2018) using a vanilla sequence classification setup, where the concatenated premise and conclusion are presented as an input sequence. Additionally, we fine-tune it within Siamese architectures, both with and without attention layers, for comparison, with each sub-network encoding the premise and conclusion separately.

3 Experiments

3.1 Experimental Setup

We employ Adam optimisation (Kingma and Ba, 2014) to minimise the cost function, using a learning rate of 2×10^{-5} and categorical cross-entropy loss. The dataset is randomly split into 80% for training and 20% for testing. The models are trained for ten epochs using a batch size of 16. The optimal model is determined based on the lowest validation loss observed throughout the epochs. For each proposition pair in the test set with known AR categories, predictions are made and evaluated against the gold standard labels. We calculate precision (P), recall (R), and F-measure (F) and report macro-averaged precision, recall, and F-measure.

[4] https://openai.com/chatgpt

3.2 Model Configurations

Proposition Decomposition: To identify the functional components within propositions (***C***, ***A***, ***OC***, and ***OA***), we implement a sequence labeling strategy following the methodology outlined by Gemechu and Reed (2019). In contrast to Gemechu and Reed's (2019) approach, which relies on a convolutional neural network (CNN), we fine-tune BERT in a token classification setup using a dataset annotated with the BIO sequence labeling scheme.

AR Identification: We conduct a comprehensive evaluation of our approaches across multiple datasets, leveraging the two ontological resources (WordNet, and ConceptNet) and Wikipedia for AR prediction. It includes three configurations of Triplet network architectures, to leverage the external resources: Triplet$\odot$Att$\oplus$wp for Wikipedia, Triplet$\odot$Att$\oplus$wn for WordNet, and Triplet$\odot$Att$\oplus$cn for ConceptNet. Similarly, we evaluate three Siamese network architectures across Wikipedia, WordNet and ConceptNet (Siamese$\odot$Att$\oplus$wp, Siamese$\odot$Att$\oplus$wn and Siamese$\odot$Att$\oplus$cn).

Furthermore, to evaluate the attention layers' impact on external resources, we compare Triplet and Siamese architectures without attention layers across the three external resources, totaling six configurations: Triplet$\odot$Vanl$\oplus$wp, Triplet$\odot$Vanl$\oplus$wn, Triplet$\odot$Vanl$\oplus$cn, Siamese$\odot$Vanl$\oplus$wp, Siamese$\odot$Vanl$\oplus$wn, and Siamese$\odot$Vanl$\oplus$cn. We also evaluate pre-trained LMs under three configurations: standard sequence classification (SeqClass$\odot$Vanl$\oplus$BERT), vanilla Siamese network (Siamese$\odot$Vanl$\oplus$BERT), and Siamese network with attention layers (Siamese$\odot$Att$\oplus$BERT). Finally, we evaluate Triplet architecture on GPT-4 generated paths (Triplet$\odot$Vanl$\oplus$gpt, Triplet$\odot$Att$\oplus$gpt).

3.3 Results and Discussions

Proposition decomposition. BERT for token classification surpasses the top-performing method proposed by Gemechu and Reed (Gemechu and Reed, 2019) by 3%, achieving a Macro F-measure of 0.784. This outcome is expected due to BERT's ability to capture both token-level and cross-token inter-dependencies, which is crucial for tagging multi-token entities. In contrast to the methodology proposed by Gemechu and Reed (2019), which relies on CNN with conventional word embeddings, BERT offers a rich semantic space of the input, re-

Model Configurations	Dataset								
	AAEC			AMT			US2016G1tv		
	P	R	F	P	R	F	P	R	F
LMs as Knowledge Base									
SeqClass⊙Vanl⊕BERT	0.78	0.73	0.75	0.79	0.67	0.72	0.56	0.64	0.60
Siamese⊙Vanl⊕BERT	0.79	0.72	0.73	0.80	0.65	0.72	0.55	0.63	0.59
Siamese⊙Att⊕BERT	0.80	0.73	0.76	0.80	0.68	0.74	0.57	0.64	0.60
No Attention + External Resources	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Triplet⊙Vanl⊕gpt	0.76	0.84	0.80	0.72	0.79	0.75	0.56	0.74	0.64
Siamese⊙Vanl⊕wn	0.84	0.79	0.81	0.82	0.75	0.78	0.62	0.70	0.66
Siamese⊙Vanl⊕cn	0.83	0.77	0.80	0.83	0.71	0.76	0.62	0.70	0.66
Siamese⊙Vanl⊕wp	0.83	0.81	0.82	0.84	0.77	0.80	0.63	0.71	0.67
Triplet⊙Vanl⊕wn	0.83	0.80	0.81	**0.85**	0.75	0.80	0.62	0.70	0.66
Triplet⊙Vanl⊕cn	0.83	0.81	0.82	0.84	0.77	0.80	0.63	0.71	0.67
Triplet⊙Vanl⊕wp	0.84	0.80	0.82	0.83	0.76	0.80	0.65	0.70	0.67
Attention + External Resources	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Triplet⊙Att⊕gpt	0.77	**0.84**	0.80	0.71	**0.85**	0.77	0.56	0.72	0.63
Siamese⊙Att⊕wn	0.84	0.81	0.82	0.83	0.80	0.81	0.62	0.73	0.67
Siamese⊙Att⊕cn	0.84	0.81	0.82	0.81	0.82	0.81	0.65	0.73	0.69
Siamese⊙Att⊕wp	0.85	0.82	0.83	0.82	0.83	0.82	0.66	0.73	0.70
Triplet⊙Att⊕wn	0.85	0.83	0.84	0.83	0.84	0.83	0.64	0.72	0.68
Triplet⊙Att⊕cn	0.84	0.82	0.83	0.82	0.84	0.83	0.65	0.73	0.69
Triplet⊙Att⊕wp	**0.87**	0.83	**0.85**	0.83	**0.85**	**0.84**	**0.67**	0.76	**0.71**

Table 2: Performance of our models on the three external resources across the three datasets.

sulting in enhanced performance.

AR identification. The evaluation of various model configurations revealed clear trends in performance. Models incorporating external resources consistently outperformed those lacking such integration, underscoring the importance of leveraging additional knowledge sources for AR identification. This integration led to a notable enhancement, surpassing the baseline by over 5% in F-measure. Likewise, models equipped with attention mechanisms consistently surpassed those without, demonstrating an average increase in F-measure of over 4% across diverse configurations. Notably, Triplet Network architecture with attention mechanism leveraging Wikipedia as an external knowledge source, attained an average F-measure of 80% across all three datasets. This represents a new state-of-the-art performance in AR identification, showcasing the effectiveness of integrating external knowledge into the model. Both Multi-network architectures (Siamese and Triplet Networks) surpassed standard sequence classification approaches, aligning with the findings of Reimers and Gurevych (2019) in sentence embedding achieving state-of-the-art performance on various tasks, including text similarity. Note that they employ pooling mechanisms, unlike the attention mechanism utilised in our approach, for learning the similarity between a pair of text. It is also interesting to observe that zero-shot methods exhibited the worst performance, indicating their limited effectiveness in AR identification tasks. In the following sections, we delve into the results to unpack the influence of model architecture and external knowledge on the performance of the models.

Model Architecture Influence. As indicated in Table 2, the incorporation of attention layers demonstrated clear benefits in configurations utilising external knowledge. It is interesting to observe that without attention, multi-network configurations fell short, being outperformed by the vanilla sequence classification approach. However, multi-network configurations with attention surpassed the vanilla sequence classification setup. This makes sense since the architecture can encourage the model to focus on learning the patterns aligning the relevant parts of the premise with the conclusion (Vaswani et al., 2017), in contrast to the standard sequence classification setup leading models to adopt shallow heuristics (Naik et al., 2018; Poliak et al., 2018; Nie et al., 2019) overlooking the AR relation patterns learned from the relations between the two sequences. Notably Naik et al. (2018) demonstrated that NLI models associates specific words like "not" and "nobody" with a contradiction label, while Poliak et al. (2018) show that

Propositions	Components, Chain of Hyperlinks
P1 - Trump tax cut is the biggest since Ronald Reagan; **P2** - It will create tremendous numbers of new jobs	**Chain of Hyperlinks for the components, Tax (C), Jobs (A):** - Job → Working hour system → Income tax → Tax - Job → Labor economics → Economic policy → Tax - Job → Unemployed → Tariff → Tax
P1 - Clinton is going to approve one of the biggest tax cut in history; **P2** - Clinton is going to drive business out	**Chain of Hyperlinks for the components, Business (C), Tax cut (A):** - Business → Adam Smith → Economic theory → Tax cut - Business → Adam Smith → Neoliberalism → Tax cut - Business → Corporate tax → Effect of taxes and subsidies on price → Tax cut

Table 3: Examples showing the connection between propositions, components, and the chain of hyperlinks linking them.

NLI models can tend to predict entailment based solely on hypotheses, exploiting dataset-specific artifacts and neglecting the context aligning the premise with the conclusion.

Moreover, attention-based approaches have proved to be better in attending to the external resources consistently outperforming their counterparts without attention layers across all configurations. The attention-based Triplet architecture surpassed the Siamese architecture, achieving an average performance increase of 1.4% in leveraging external knowledge. This improvement can be attributed to its design, which enables each subnetwork to focus on specific features while concurrently learning two levels of alignments: the alignment between the premise and conclusion, and the alignment between the external resource and the premise-conclusion pair.

Models trained on GPT-4 generated paths outperformed the models without external knowledge. This is inline with other similar works exploiting LM generated commonsense knowledge, (Bansal et al., 2022), yet it is still providing low precision as compared to the other sources. The high recall can be linked to the models tendency to find any reason to create paths between entities. And these paths involves chain of thoughts unintended in the original argument, which might be resulted from hallucination as already pointed out by other works relaying on LM to generate commonsense knowledge (Levy et al., 2022). Our analysis relies on manually inspecting a sample of the paths generated by GPT-4. It was observed that for certain false positive predictions, the generated paths did not align with the reasoning presented in the original argument.

External knowledge influence. Our experiment revealed that models augmented with external resources, consistently outperformed the vanilla LM-only configurations. For example, the Siamese architecture leveraging Wikipedia achieved an average F-measure of 76%, while its counterpart without the external resource attained 73%. This underscores, despite their prowess, LMs alone may not fully exploit the depth and specificity of knowledge required for certain applications, notably structured reasoning (Kassner and Schütze, 2019). This finding aligns with previous research demonstrating that while LMs tend to encode relevant world knowledge, they struggle to capture the depth of reasoning necessary for tasks such as AR identification. For example, Polu et al. (2022) revealed their limitations in chaining multiple steps of complex logical reasoning, while Merrill et al. (2021) demonstrated they fail to comprehend the semantics behind commonsense reasoning tasks. This limitation is particularly critical in the context of AR identification, where understanding the chain of reasoning that can be encoded via the indirect relationships between entities is essential.

Wikipedia-based models outperformed both the baseline and ontology-based models across all three datasets. For example, the the attention based Triplet network on Wikipedia (Triplet$\odot$Att$\oplus$wp) achieved an F-measure of 0.85, 0.84, and 0.71 for identifying AR on AAEC, AMT, and US2016G1tv, respectively. Upon manual inspection of hyperlink chains connecting the components of propositions, it was observed that Wikipedia exhibits better results due to the presence of paths linking concepts that are missing in ontological resources. This disparity is attributed to Wikipedia's rich network of hyperlinks connecting pages using arbitrary relations, unlike ontological resources that only connect Synsets based on predefined sets of semantic relations. It is noteworthy that models augmented with external resources can produce false positives by generating pathways of ideas not intended in the original argument, leading to interpretations divergent from the original context.

Approaches	Data sets								
	AAEC			AMT			US2016G1tv		
	P	R	F	P	R	F	P	R	F
Potash et al. (2016)	n/a		0.77	n/a		0.74	n/a		
Kobbe et al. (2019)	n/a		0.59	n/a		0.67			
Peldszus and Stede (2016)	n/a					0.76			
Eger et al. (2017)			0.51^{F1}	n/a					
GPT-4	0.61	0.45	0.52	0.58	0.42	0.44	0.54	0.40	0.46
Gemechu and Reed (2019)	0.81	0.74	0.77	**0.88**	0.66	0.75	0.63	0.61	0.62
Triplet⊙Att⊕wp	**0.87**	**0.83**	**0.85**	0.83	**0.85**	**0.84**	**0.67**	**0.76**	**0.71**

Table 4: The performance of Potash et al. work (2016), Eger et al. technique (2017), Peldszus and Stede's (2016) technique, Kobbe et al. work (Kobbe et al., 2019), OpenAI's GPT-4 (OpenAI, 2023), Gemechu and Reed's work (Gemechu and Reed, 2019) and Triplet⊙Att⊕wp in identifying AR on the datasets.

3.4 Comparison Systems

We have compared our approach to related works (Eger et al., 2017; Peldszus and Stede, 2016; Gemechu and Reed, 2019; Potash et al., 2016; Kobbe et al., 2019; Plenz et al., 2023; OpenAI, 2023).

Eger et al. (2017) propose a neural approach for argument mining, treating the task as dependency parsing and token-based sequence tagging. They also investigate a multi-task setup to exploit the dependency between component identification and AR prediction, achieving an F1-score of 0.51 for AR identification on AAEC dataset. Peldszus and Stede (2016) aim to map RST trees to argumentation structures (Taboada and Mann, 2006) using sub-graph matching and an evidence graph model. They evaluate several features of their system on AMT and achieve an overall F-measure of 0.76 in identifying ARs. Similarly, Potash et al. (2016) train an encoder-decoder (Sutskever et al., 2014) with attention mechanism (Bahdanau et al., 2014) to identify AR. Gemechu and Reed (2019) decompose propositions into fine-grained components and use classifiers to predict AR based on the relations between these components. We also prompt GPT-4 in AR identification. Please note that direct comparisons with some of these works may not be feasible due to variations in task setup and complexities. For instance the work of Eger et al. (2017) involves argument segmentation in addition to AR identification as an end-to-end task. In our case, the goal is to identify AR based on the correct segments. However, as can be seen from Table 4, our techniques outperform the comparison systems, including OpenAI's GPT-4 (OpenAI, 2023) for identifying AR across the datasets.

Most related to our work are Kobbe et al.'s (2019) and Plenz et al.'s (2023) studies, as they also leverage paths connecting propositions through nodes and directed edges. However, their methodologies diverge significantly from ours. Firstly, they rely on structured knowledge bases with predefined relation types, limiting their adaptability to resources like Wikipedia, which lack explicit relation types. Additionally, Kobbe et al. utilise traditional feature encoding techniques, while Plenz et al., similar to our approach, integrate path embeddings for richer semantic features. However, Plenz et al. align entire propositions with knowledge base triplets, posing challenges with heterogeneous concepts, unlike our focused alignment based on triplet and proposition functional components similarity. Moreover, they do not address the challenge of encoding external information into model architectures. Instead, they employ Dijkstra's algorithm for weighting paths, and the effectiveness of their approach on AR identification is not evaluated.

4 Conclusion

Our investigation into argument mining using external resources has revealed significant insights regarding the influence of model architecture and external knowledge on performance. Our exploration of various model configurations underscored the significance of external resources and multi-network architecture with attention mechanisms in AR identification. Models augmented with external resources, such as Wikipedia, consistently outperformed those relying solely on language models. This emphasises the necessity of leveraging supplementary knowledge sources to enrich argument understanding, where language models alone may fall short. Structured knowledge from LM enhances performance, though the external sources remain a reliable option. Furthermore, multi-network architectures featuring attention mechanisms, notably the attention-based Triplet, demonstrated superiority across all configurations.

Limitations

While our work presents promising advancements, it is essential to acknowledge several limitations that should be considered. The sections mentioned above discuss some of the key limitations, including evaluation domain specificity, dependency on external knowledge quality, and interpretability and explainability.

Evaluation Domain Specificity. Robust evaluation involving cross-domain evaluation, where models are trained on one domain and evaluated on a new domain, is essential for uncovering the robustness of the proposed approaches. While our evaluation has primarily focused on specific domains or datasets, cross-domain evaluation can provide insights into the generalizability and adaptability of the models across diverse domains and real-world applications. By assessing model performance in different domains, researchers can better understand the strengths and limitations of the proposed approaches and identify areas for improvement. Additionally, cross-domain evaluation can help mitigate the risk of overfitting to specific datasets or domains, leading to more robust and reliable models.

Dependency on External Knowledge Quality. More work is required in aligning the concepts with external resources, particularly in disambiguating the senses of the Synsets and Wikipedia page titles. Our current approach relies on simple similarity measures between the embeddings of glosses of the resources and the components, which may lead to missing alignments and empty paths between certain concepts. Improving the alignment procedure to account for semantic ambiguity and variability in external resources is crucial for enhancing the effectiveness of the proposed approach. Additionally, sophisticated techniques are needed to identify the semantic relation types existing between Wikipedia hyperlinks. Unlike ontologies, Wikipedia does not encode explicit semantic relation types between hyperlinks. Therefore, developing methods to infer and leverage implicit semantic relations from Wikipedia articles can improve the quality and relevance of external knowledge integration in argument relation identification tasks.

Interpretability and Explainability. The explanations provided regarding the performance of the architectures and external resources are primarily based on the analysis of empirical results. While empirical analysis is valuable for understanding model behavior, additional techniques beyond the results themselves can provide deeper insights into model performance. Exploring techniques such as model visualization, attention mechanisms analysis, and interpretability methods like LIME (Local Interpretable Model-Agnostic Explanations) or SHAP (SHapley Additive exPlanations) can help uncover the underlying reasons behind model decisions and configurations. By complementing empirical analysis with interpretability techniques, researchers can gain a more comprehensive understanding of model behavior and identify potential avenues for model improvement.

References

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, and Marjan Ghazvininejad. 2022. A review on language models as knowledge bases. *arXiv preprint arXiv:2204.06031*.

Nicholas Asher and Alex Lascarides. 2003. *Logics of conversation*. Cambridge University Press.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Rachit Bansal, Milan Aggarwal, Sumit Bhatia, Jivat Neet Kaur, and Balaji Krishnamurthy. 2022. Cose-co: Text conditioned generative commonsense contextualizer. *arXiv preprint arXiv:2206.05706*.

Teresa Botschen, Daniil Sorokin, and Iryna Gurevych. 2018. Frame-and entity-based knowledge for common-sense argumentative reasoning. In *Proceedings of the 5th Workshop on Argument Mining*, pages 90–96.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Diana Inkpen, and Si Wei. 2017. Neural natural language inference models enhanced with external knowledge. *arXiv preprint arXiv:1711.04289*.

Carlos Chesnevar, Jarred McGinnis, Sanjay Modgil, Iyad Rahwan, Chris Reed, Guillermo Simari, Matthew South, Gerard Vreeswijk, and Steven Willmott. 2006. Towards an argument interchange format. *The knowledge engineering review*, 21(4):293–316.

HongSeok Choi and Hyunju Lee. 2018. Gist at semeval-2018 task 12: A network transferring inference knowledge to argument reasoning comprehension task. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 773–777.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Mauro Dragoni, Celia da Costa Pereira, Andrea GB Tettamanzi, and Serena Villata. 2018. Combining argumentation and aspect-based opinion mining: the smack system. *AI Communications*, 31(1):75–95.

Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. 2017. Neural end-to-end learning for computational argumentation mining. *arXiv preprint arXiv:1704.06104*.

Peter W Foltz, Walter Kintsch, and Thomas K Landauer. 1998. The measurement of textual coherence with latent semantic analysis. *Discourse processes*, 25(2-3):285–307.

Michael Fromm, Evgeniy Faerman, and Thomas Seidl. 2019. Tacam: Topic and context aware argument mining. *arXiv preprint arXiv:1906.00923*.

Debela Gemechu and Chris Reed. 2019. Decompositional argument mining: A general purpose approach for argument graph construction. In *Proceedings of the 57st Annual Meeting of the Association for Computational Linguistics*, pages 1341–1351.

Talmy Givón. 1987. Beyond foreground and background. *Coherence and grounding in discourse*, 11:175–188.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking nli systems with sentences that require simple lexical inferences. *arXiv preprint arXiv:1805.02266*.

Yoav Goldberg. 2019. Assessing bert's syntactic abilities. *arXiv preprint arXiv:1901.05287*.

Barbara J Grosz, Aravind K Joshi, and Scott Weinstein. 1995. Centering: A framework for modelling the local coherence of discourse.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2017. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. *arXiv preprint arXiv:1708.01425*.

Mathilde Janier, John Lawrence, and Chris Reed. 2014. OVA+: an argument analysis interface. In *Computational Models of Argument: Proceedings of COMMA*, volume 266, page 463.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Nora Kassner and Hinrich Schütze. 2019. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. *arXiv preprint arXiv:1911.03343*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Jonathan Kobbe, Juri Opitz, Maria Becker, Ioana Hulpus, Heiner Stuckenschmidt, and Anette Frank. 2019. Exploiting background knowledge for argumentative relation classification. In *2nd Conference on Language, Data and Knowledge (LDK 2019)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

John Lawrence, Mathilde Janier, and Chris Reed. 2015. Working with open argument corpora. In *European Conference on Argumentation*, pages 367–380.

John Lawrence and Chris Reed. 2020. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196.

Sharon Levy, Emily Allaway, Melanie Subbiah, Lydia Chilton, Desmond Patton, Kathleen McKeown, and William Yang Wang. 2022. Safetext: A benchmark for exploring physical safety in language models. *arXiv preprint arXiv:2210.10045*.

Daniel Marcu. 2000. *The theory and practice of discourse parsing and summarization*. MIT press.

William Merrill, Yoav Goldberg, Roy Schwartz, and Noah A Smith. 2021. Provable limitations of acquiring meaning from ungrounded form: What will future language models understand? *Transactions of the Association for Computational Linguistics*, 9:1047–1060.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Amita Misra, Pranav Anand, Jean E Fox Tree, and Marilyn Walker. 2017. Using summarization to discover argument facets in online ideological dialog. *arXiv preprint arXiv:1709.00662*.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. *arXiv preprint arXiv:1806.00692*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*.

R OpenAI. 2023. Gpt-4 technical report. arxiv 2303.08774. *View in Article*, 2:13.

Andreas Peldszus and Manfred Stede. 2013. Ranking the annotators: An agreement study on argumentation structure. In *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 196–204.

Andreas Peldszus and Manfred Stede. 2016. Rhetorical structure and argumentation structure in monologue text. In *Proceedings of the Third Workshop on Argument Mining*, pages 103–112.

Isaac Persing and Vincent Ng. 2016. End-to-end argumentation mining in student essays. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1384–1394.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. 2013. Align, disambiguate andwalk: A unified approach for measuring semantic similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1341–1351.

Moritz Plenz, Juri Opitz, Philipp Heinisch, Philipp Cimiano, and Anette Frank. 2023. Similarity-weighted construction of contextualized commonsense knowledge graphs for knowledge-intense argumentation tasks. *arXiv preprint arXiv:2305.08495*.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. *arXiv preprint arXiv:1805.01042*.

Stanislas Polu, Jesse Michael Han, Kunhao Zheng, Mantas Baksys, Igor Babuschkin, and Ilya Sutskever. 2022. Formal mathematics statement curriculum learning. *arXiv preprint arXiv:2202.01344*.

Peter Potash, Robin Bhattacharya, and Anna Rumshisky. 2017. Length, interchangeability, and external knowledge: Observations from predicting argument convincingness. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 342–351.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2016. Here's my point: Joint pointer architecture for argument mining. *arXiv preprint arXiv:1612.08994*.

Ellen F Prince. 1981. Toward a taxonomy of given-new information. *Radical pragmatics*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. 2019. Classification and clustering of arguments with contextualized word embeddings. *arXiv preprint arXiv:1906.09821*.

Ruty Rinott, Lena Dankin, Carlos Alzate Perez, Mitesh M Khapra, Ehud Aharoni, and Noam Slonim. 2015. Show me your evidence-an automatic method for context dependent evidence detection. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 440–450.

Tara Safavi and Danai Koutra. 2021. Relational world knowledge representation in contextual language models: A review. *arXiv preprint arXiv:2104.05837*.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Maite Taboada and William Mann. 2006. Rhetorical structure theory: Looking back and moving ahead. *Discourse studies*, 8(3):423–459.

Dietrich Trautmann. 2020. Aspect-based argument mining. *arXiv preprint arXiv:2011.00633*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Jacky Visser, Barbara Konat, Rory Duthie, Marcin Koszowy, Katarzyna Budzynska, and Chris Reed. 2019. Argumentation in the 2016 us presidential elections: annotated corpora of television debates and social media reaction. *Language Resources and Evaluation*, pages 1–32.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018a. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Su Wang, Greg Durrett, and Katrin Erk. 2018b. Modeling semantic plausibility by injecting world knowledge. *arXiv preprint arXiv:1804.00619*.

Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2019. Bert post-training for review reading comprehension and aspect-based sentiment analysis. *arXiv preprint arXiv:1904.02232*.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2019. Semantics-aware bert for language understanding. *arXiv preprint arXiv:1909.02209*.

A Supplementary Material

We provide additional details regarding the methodology and experimental setups used in our study.

A.1 Alignment of Ontologies and Wikipedia

For aligning ontologies and Wikipedia with the components of propositions, we employ a multi-step procedure. Initially, we align them based on surface-level matching. In cases where surface-level matching is not applicable, we compute the cosine similarity between the components and the Synsets of the ontologies. Similarly, for aligning with Wikipedia, we compute the cosine similarity between the components and the corresponding Wikipedia articles. Additionally, we utilize the similarity between the concepts and the gloss texts of the respective sources for disambiguating senses, especially for concepts involving multiple senses.

A.2 Path Generation Using ChatGPT-4

We employ the chat completion setup of ChatGPT-4 based on `gpt-3.5-turbo-instruct` for generating the path of semantic relations between the components of propositions and argument relation (AR) predictions. The respective instructions provided to ChatGPT-4 for this task are as follows.

A.3 Attention Mechanisms in Multi-Network Architectures

We introduce a Triplet Network architecture, aiming to encode the individual components of propositions as well as the external knowledge paths connecting them. The architecture consists of three sub-networks, each focusing on a different aspect of the input:

- **Sub-Network 1**: Encodes the premise using the BERT model and extracts its representation.
- **Sub-Network 2**: Encodes the conclusion using the same BERT model and obtains its representation.
- **Sub-Network 3**: Encodes the paths between components of propositions using the BERT model.

After encoding, attention mechanisms are employed to capture relevant information between different parts of the input. We utilize two attention layers for this purpose.

1. **First Attention Layer**: This layer attends to the information from the premise and paths encoded in Sub-Networks 1 and 3, respectively. The output of the first attention layer is then passed to the next layer for further processing.
2. **Second Attention Layer**: Building upon the output of the first attention layer, this layer attends to the combined information from the first attention layer and the conclusion encoded in Sub-Network 2. This allows the network to integrate information from all components of the input sequence.

Finally, the outputs of the attention layers are averaged to obtain a unified representation of the input, which is then passed through a linear classification layer to predict the argument relation category.

We experimented with different configurations for the attention layers. Initially, we explored using the final output of the cls token and the mean of the last output of all tokens to compute the attention. Through experimentation, we found that using the last hidden layers of the entire sequence to compute attention provided the best performance.

A.4 Input Setup

The input to the model follows a specific setup. For the sequence classification task, we concatenate the premise to the conclusion using a special token `[SEP]`. In the Siamese architecture, one of the sub-networks takes the concatenation of the premise and conclusion based on the special token `[SEP]`, while the other takes the concatenation of the chain of hyperlinks. Hyperlinks are concatenated using the special token `[SEP]`.

The number and length of the paths between the components of the propositions vary, with some propositions not involving any path at all. For proposition pairs involving a large number of paths exceeding the maximum sequence length, we concatenate the paths until the maximum sequence length is reached. In such cases, we sort the paths based on their weighted frequency, where paths appearing more frequently across the dataset receive higher weights. The concatenation process starts from the most frequent paths until the maximum sequence length is reached.

A.5 Baseline Language Model Configuration

For the baseline language model configuration, we utilised BERT-for-sequence classifica-

tion. Specifically, we experimented with two variants of BERT: `bert-base-uncased` and `bert-large-uncased`. Our experiments revealed that `bert-large-uncased` consistently provided better performance compared to `bert-base-uncased`. This observation guided our choice of language model for the baseline configuration.

A.6 Training Procedure Setup

During the training procedure, we employed various techniques to optimise the learning process and improve model performance. Specifically, we utilised scheduler, gradient clipping, and early stopping techniques.

Scheduler: We incorporated a scheduler to adjust the learning rate dynamically during training. In particular, we employed the Stochastic Gradient Descent (SGD) optimizer with the cosine annealing scheduler technique, also known as the Stochastic Gradient Descent with Warm Restarts (SGDR) technique. This technique gradually decreases the learning rate in a cyclical manner, allowing the model to escape from local minima and converge faster.

Gradient Clipping: To prevent exploding gradients during training, we applied gradient clipping. This technique limits the magnitude of gradients to a predefined threshold, thereby stabilising the training process and preventing divergence.

Early Stopping: We implemented early stopping to prevent overfitting and to determine the optimal number of epochs. Early stopping monitors the validation loss during training and terminates training if the validation loss does not improve for a specified number of consecutive epochs.

Additionally, we conducted hyperparameter tuning to identify the optimal values for parameters such as learning rate, number of epochs. This process involved systematically searching through a predefined parameter space using grid search to identify the combination of hyperparameters that resulted in the best model performance.

In our implementation, we integrated these techniques into our training pipeline using the PyTorch library.