RGAT at SemEval-2024 Task 2: Biomedical Natural Language Inference using Graph Attention Network

Abir Chakraborty
Microsoft
Abir.Chakraborty@microsoft.com

Abstract

In this work, we (team RGAT) describe our approaches for the SemEval 2024 Task 2: Safe Biomedical Natural Language Inference for Clinical Trials (NLI4CT). The objective of this task is multi-evidence natural language inference based on different sections of clinical trial reports. We have explored various approaches, (a) dependency tree of the input query as additional features in a Graph Attention Network (GAT) along with the token and parts-of-speech features, (b) sequence-to-sequence approach using various models and synthetic data and finally, (c) in-context learning using large language models (LLMs) like GPT-4. Amongst these three approaches the best result is obtained from the LLM with 0.76 F1-score (the highest being 0.78), 0.86 in faithfulness and 0.74 in consistence.

1 Introduction

Clinical trials are advanced treatments and tests to evaluate new ways of treating life-threatening diseases where interventions include new drugs, cells and other biological products, advanced surgical or radiological procedures and devices. As the trial progresses the observations are documented systematically in a Clinical Trial report that includes the subject selection criteria ('Eligibility'), treatments ('Interventions') and results at group level including adverse effects. These reports constitute a rich source of past endeavours to learn from and help in formulating new treatment plans. However, the sheer volume of CT reports[1] makes it impossible to conduct extensive manual evaluation. Thus, it is necessary to have an automated pipeline that can enquire a CT report for specific hypothesis and provides high accuracy and reliability at the same time.

[1]As of Jan 17, 2024, ClinicalTrials.gov lists 480,795 CT studies

Natural language inference or NLI (Devlin et al., 2019) is one of the standard NLP tasks where a hypothesis is qualified as true (entailment) or false (contradiction) or even undetermined (neutral) given a premise. This task is adopted for reasoning over CT reports by Jullien et al. (2023) where two new tasks are created based on NLI4CT dataset, (1) NLI over CT reports and (2) extracting the evidence/mention from CT reports to support the inference label. The Semeval 2024 Task 2 NLI4CT is also based on the same NLI4CT dataset (identical for training) with modifications in the test split (more details in the Data section). The inferencing is challenging as it requires multi-hop reasoning, i.e., dependency and aggregation are required over different pieces of the document.

Other than the complexity associated with multi-hop reasoning, the domain and the associated word-distribution also creates significant challenge due to the presence of aliases, acronyms and biomedical terminologies (Lee et al., 2019a; Shickel et al., 2018; Jin et al., 2019). This results in significant drop in model performance as is evident in the NLI results last year (Jullien et al., 2023) where it was found that majority of the submitted solutions failed to outperform the baseline solution with a significant margin. The challenge is also evident in the overall performance of models on general NLI datasets (e.g., Stanford NLI or SNLI) where the best model results in 93.1% F1-score (Wang et al., 2021).

When it comes to different modeling approaches, many of the top-performing models for the SNLI dataset are ensemble in nature. While initial individual models are based on RNN, most of the latest ones are based on the Transformer architecture and pretrained language models like RoBERTa or T5. Similar trend can also be seen in Jullien et al. (2023) where the best model is an ensemble and both DeBERTa and Flan-T5 made their way to the top. Interestingly, LLMs like GPT3.5 could not

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 116–122

make a significant boost in the performance.

In our approach, we explored three different modeling paradigms, namely, (1) custom Graph Attention Network (GAT) based discriminative model with novel features based on the dependency tree of the input query, (2) generative models based on T5 and Flan-T5 but enriched with synthetic data used for both pre-training and fine-tuning, and (3) LLM like GPT-4 applied with and without few-shot examples. It is not surprising that the best performance was obtained by GPT-4 stressing on the importance of generic knowledge (that is embedded in these LLMs) rather than fine-tuning, especially when the dataset is not large enough.

The organization of the paper is as follows. In the next section we provide a detailed literature survey on the techniques employed for NLI. Next, we present the details of the proposed approaches. Subsequently, the model predictions and comparisons with other baseline methods are discussed. Finally, conclusions are drawn and scope for future works is outlined.

2 Related Work

The existing body of work for the general NLI is quite rich where they are based on the Stanford NLI (SNLI) dataset (550k examples but restricted to a single text genre) (Bowman et al., 2015) and three other NLI datasets present in GLUE (Wang et al., 2018), namely, MNLI, QNLI and WNLI. The MNLI (Multi-Genre Natural Language Inference Corpus) dataset (Williams et al., 2018) is a crowd-sourced NLI dataset gathered from different sources, e.g., government reports (and covers different genres, e.g., fiction, travel). Given a premise-hypothesis pair of sentences, the task is to predict one of the three classes, namely, whether the premise sentence entails the hypothesis (entailment), contradicts the hypothesis (contradiction), or neither (neutral). The QNLI is modified from Stanford Question Answer Dataset (Rajpurkar et al., 2016) where the task is to determine whether the context sentence contains the answer to the question. Similarly, the WNLI dataset is created from the Winograd Schema Challenge (Levesque et al., 2012) where a coreference resolution problem is converted into an entailment problem involving a pronoun and its referent. Another large NLI dataset is multi-genre NLI (MNLI) that has 433k examples covering multiple genres and supporting cross-genre evaluation. Some of the best performances are obtained by RoBERTa (Liu et al., 2019b), XLNet (Yang et al., 2020), Multi-Task Deep Neural Network (MT-DNN) (Liu et al., 2019a) and generative pre-training (GPT) approach (Radford et al., 2018).

There are few NLI datasets in the biomedical domain, namely, MedNLI (Romanov and Shivade, 2018) and BioNLI (Bastan et al., 2022). MedNLI has 14k example pairs created by clinicians on 4,683 premises with three categories, entailment, contradiction and neutral. BioNLI, on the other hand, goes beyond sentence-level inference and includes large context as premises that requires handling complex texts as well as domain knowledge. Bastan et al. also includes negative examples as adversarial hypothesis using nine strategies which is a speciality of this dataset.

There are three biomedical domain specific models that are typically used on these datasets. Starting with the available weights of BERT (pretrained on general domain corpora), BioBERT (Lee et al., 2019b) is trained on PubMed abstracts and PMC full-text articles and shown to outperform BERT on NER, relation extraction and Q&A, all in the biomedical domain. PubMedBERT (Gu et al., 2021) is a BERT model created from scratch (rather than starting with general domain corpora) on large biomedical domain dataset like PubMed and achieved impressive performance for tasks like NER and Q&A. BioLinkBERT (Yasunaga et al., 2022) further exploited links between PubMed documents to create a richer context that is used to build a language model (LM). This model has obtained SOTA performance on biomedical datasets such as BLURB (Gu et al., 2021) and BioASQ (Nentidis et al., 2020). Another model that achieved SOTA performance on MedNLI is SciFive (Phan et al., 2021) which is based on T5 paradigm.

There are not many studies on the application of Graph Neural Network for NLI. Inspired by KIM (Chen et al., 2018) where external knowledge is infused for NLI task, Song et al. (2020) developed a joint training model where Graph Attention Network (GAT) is used to represent the sub-graph associated with entities that are involved in the hypothesis. Another closely related GAT application is from Chen et al. (2021) applied for fact verification on Wikipedia articles. Typical applications of GAT in the NLP domain are for question answering, semantic parsing, information extraction and Named Entity Recognition (Wu et al., 2022;

Chakraborty, 2023).

3 Task Description & Data

The dataset for Multi-evidence NLI for Clinical Trial (NLI4CT) is based on a collection of breast-cancer CT reports[2] containing statements, explanations and labels annotated by domain expert annotators (Jullien et al., 2024). Each CT report has four sections: (a) Eligibility criteria (a set of conditions for patients to be included in the trial cohort), (b) Intervention (information regarding the details of treatments administered), (c) Results (what is the outcome of these treatments) and (d) Adverse events (if anything was observed during the period of the trial). The annotated statements (hypothesis) are claims extracted from one of the four sections (with an average length of 19.5 tokens) and may even compare more than one report. Each statement is qualified as either 'Contradiction' or 'Entailment'.

There are 1700 examples in the training set and 200 in the development/validation set with exactly 50:50 split of the two classes. The test set has 5500 examples with unknown label distribution. A typical example looks like the following:

1. **Hypothesis**: 'All the primary trial participants do not receive any oral capecitabine, oral lapatinib ditosylate or cixutumumab IV, in contrast all the secondary trial subjects receive these.'
2. **Primary context**: 'Patients with early stage, ER positive primary breast cancer undergo FLT PET scan at baseline and 1-6 weeks after the start of standard endocrine treatment. The surgery follows 1-7 days after the second FLT PET scan.'
3. **Secondary context**: 'Patients receive oral capecitabine twice daily on days 1-14 and oral lapatinib ditosylate once daily on days 1-21. Courses repeat every 21 days in the absence of disease progression or unacceptable toxicity'
4. **Label**: 'Contradiction'

where the secondary context provides the justification of the label.

4 Methodology

We have explored three different modeling strategies for the prediction of the inference label. They are (1) custom discriminative model with GAT applied to create features from the dependency tree of the hypothesis statement, (2) sequence-to-sequence generative models based on T5 and Flan-T5 but enriched with synthetic data used in both pre-training and fine-tuning and (3) LLM based solution with and without Few-shot examples.

[2]extracted from https://clinicaltrials.gov/ct2/home

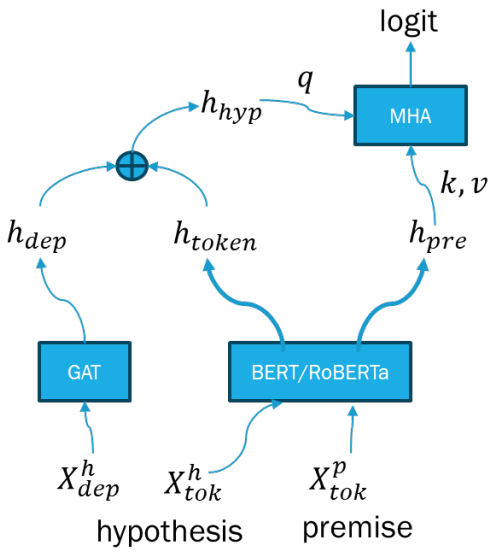

Figure 1: The architecture of the custom model using GAT and Multi-head attention (MHA).

4.1 Discriminative Model

The architecture of our custom discriminative model is shown in Fig 1. We use the tokens of both the hypothesis and the premises to generate a representation using either a standard BERT or RoBERTa model (referred as h_{tokens} for the hypothesis and h_{pre} for the premise. Following the RGAT approach of Wang et al. (2020) (originally meant for aspect polarity detection) we utilize the dependency structure of the input hypothesis (X^h_{dep}) that captures the grammatical relations by connecting the words with the corresponding dependency type. However, we do not reorient the dependency tree since there is no aspect word in our application. Using GAT based processing of the hypothesis dependency tree we generate additional features h_{dep}. Details of the GAT based processing are provided in Appendix A. We concatenate both the features of the hypothesis (h_{dep} and h_{tokens}) and pass through a linear layer to create the final hypothesis feature, h_{hyp}. For the premise, there is only the token based feature, h_{pre}, which is used as a key and value in a standard multi-head attention (MHA) with h_{hyp} as the query vector. This process

is repeated multiple times (maximum 3) with the output of the previous MHA layer. Finally, we take the first vector of the MHA output (corresponding to [CLS]) and pass it through a linear layer to generate the logits. The model is trained for binary cross-entropy loss.

4.2 Generative Model

In the 2023 SemEval challenge (Jullien et al., 2023), it was found that generative models outperformed discriminative models on the entailment task. We also explore different T5 models (small and base T5 and base SciFive) for the current entailment task with the exception that we have also generated synthetic data for pre-training as well as fine-tuning.

4.2.1 Generation of Synthetic Data

For generating synthetic data for T5 pre-training we follow (1) the standard T5 random span masking[3] for both the hypothesis and premise sentences and (2) ask GPT-4 to identify spans and mask them subsequently. The first approach works better for the quality of the data and we use this approach for generating the final pre-training data. We have used noise density = 0.4 and average noise span length of 2 and generate 73,457 pre-training examples.

For generating additional fine-tuning data, we use GPT-4 (with temperature = 0.7) with three additional tasks, namely, (a) Question answering on the premise text, and (b) additional inference data from the same set of premises and (c) create a contradictory hypothesis from the original hypothesis. For the first task, examples look like

1. Question: 'How many weeks after the start of standard endocrine treatment is the second FLT PET scan conducted?', Answer: '1-6 weeks'

2. Question: 'On which days is oral capecitabine given in Arm A?', Answer: 'days 1-14'

Additional NLI examples are

1. Hypothesis:No adverse events were reported in the clinical trial., Label: Entailment

2. Hypothesis: The clinical trial report had 765 adverse events in one section and 88 in another section., Label: Contradiction

In this process we generate 11k Q&A pairs and 45k NLI pairs and 1700 contradictory NLI examples from the original 1700 training examples.

4.3 Large Language Model

It was also observed in 2023 SemEval challenge (Jullien et al., 2023) that increase in model size also improves the performance. We further validate this hypothesis by applying GPT-4 to the NLI task with and without few-shot examples.

4.4 Implementation Details

For the discriminative model we use the bi-affine parser (Dozat and Manning, 2016) from AllenNLP for dependency parsing. For all experiments, the embedding dimension for the dependency relation is same as the hidden dimension of the BERT/RoBERTa model. We use 3 MHA layers with 8 heads and 2 GAT layers with 6 heads and all the dropouts are fixed at 0.3. The model has a total of 110 million parameters for BERT-base and 351 million parameters for BERT-large. The last hidden state of the pre-trained BERT[4] is used for the initial token representations which is subsequently fine-tuned. All models are trained for 50 epochs using Adam optimizer (Kingma and Ba, 2014) (with the default parameters), a learning rate of 5×10^{-5} and a batch size of 8.

We have pretrained both small and base T5 models for subsequent NLI task. Pretraining is done for 20 epochs with a batch size of 16 and learning rate of 5×10^{-5} with Adam optimizer. From the 73,457 span masked examples, we use 66111 for training and 7346 for validation that is used to keep track of the validation loss and saving the model.

5 Results

In this section, first we describe the performance of the custom discriminative model followed by the performance of the fine-tuned T5 model and finally the results from GPT-4. Although we compute precision, recall and F1-score for all our experiments we report only F1-score here. It is to be noted that we did not evaluate our model on the test dataset for all our experiments and submitted test results only for the best validation performance. Thus, for most of our experiments we report only the validation F1-score and also mention the test F1-score wherever available. Table 1 summarizes the results from the custom discriminative model. There are

[3]https://github.com/google-research/text-to-text-transfer-transformer

[4]https://github.com/huggingface/transformers

Model Type	Base Model	Model Parameters	Dev-F1	Test-F1
Cross-attention	BERT-base	110 M	0.64	
Combined pooler	BERT-base	110 M	0.65	
Cross-attention + GAT	BERT-base	110 M	**0.67**	0.49
Cross-attention + GAT	BERT-large	351 M	**0.67**	0.50

Table 1: Performance of the custom discriminative model on the validation and test dataset

Model Type	Model	Additional Data	Dev-F1	Test-F1
random initial weight	small T5 (60.5 M)	None	0.55	
random initial weight	small T5 (60.5 M)	synthetic NLI data-I	0.51	
random initial weight	small T5 (60.5 M)	synthetic NLI data-II	0.53	
pretrained with CTR data	small Flan-T5 (76 M)	None	0.58	
pretrained	base T5 (223 M)	None	0.64	
pretrained	base T5 (223 M)	Synthetic Q&A data	0.43	
pretrained	base T5 (223 M)	Synthetic NLI-I data	0.55	
pretrained	base T5 (223 M)	Synthetic NLI-II data	0.54	
pretrained	Flan-T5 base (247 M)	None	0.66	0.608
pretrained	Flan-T5 base (247 M)	Synthetic NLI-I	-	0.535
-	GPT-4 (`0613`)	Zero-shot	-	**0.761**

Table 2: Performance of different generative models including GPT-4.

four flavors of this model, one with BERT-large and three with BERT-base. Within BERT-base, we have one with cross-attention, one without ('combined-pooler' that only concatenates the two BERT outputs) and the third one with cross-attention and GAT. It can be seen that the presence of GAT improves the validation F1 score over the other variants. However, the performance does not improve with the larger BERT model. Surprisingly, the corresponding test F1-score shows significant degradation implying substantial difference in the test data distribution (tokens, nature of problem or label) from that of the validation dataset. The small number of validation dataset also contributes to this mismatch.

Table 2 captures the details of different experiments with generative models like, T5, Flan-T5 and GPT-4. The size of the generative model (small vs. base) has strong contribution to the performance as confirmed earlier (Jullien et al., 2023). However, the addition of synthetic data does not improve (rather degrade) the F1-score which is evident for both the small and base version of T5. This challenges the traditional belief of improvement due to multi-task learning and indicates potential conflicts in the synthetic data due to either a mismatch in the nature of the problem (e.g., Q&A) or accuracy of the synthetic data (since they are not manually verified). The best result is obtained by a base Flan-T5 model trained without any synthetic dataset that results in a test F1-score of 0.61. Finally, using GPT-4 (version `0613`, maximum context length of 8192) without any Few-shot examples results in the best test F1-score of 0.76.

6 Conclusion

In this work we have explored both discriminative and generative models for NLI applied to CT reports. While our custom discriminative model outperforms generative models like T5-base and Flan-T5-base the same is not true when evaluated on the test dataset indicating the limitation of the small validation dataset and significant change in data distribution. Since the training dataset is small (1700) we also explore enriching the same with synthetic data created by LLMs like GPT-4 for additional task (e.g., Q&A) and the same NLI task. However, the addition of these synthetic data substantially degrades the performance rather than improving pointing to a deeper analysis of the role of synthetic data for NLI task. The only exception is in the pretraining synthetic data created for small Flan-T5 model that boosted the final performance. The best result is obtained by GPT-4 without using Few-shot examples and we suspect both the addition of examples and modification of the prompt can further improve the performance.

References

Mohaddeseh Bastan, Mihai Surdeanu, and Niranjan Balasubramanian. 2022. BioNLI: Generating a biomedical NLI dataset using lexico-semantic constraints for adversarial examples. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5093–5104, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Abir Chakraborty. 2023. RGAT at SemEval-2023 task 2: Named entity recognition using graph attention network. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 163–170, Toronto, Canada. Association for Computational Linguistics.

Chonghao Chen, Jianming Zheng, and Honghui Chen. 2021. Knowledge-enhanced graph attention network for fact verification. *Mathematics*, 9(16).

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Diana Inkpen, and Si Wei. 2018. Neural natural language inference models enhanced with external knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2406–2417, Melbourne, Australia. Association for Computational Linguistics.

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.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23.

Qiao Jin, Jinling Liu, and Xinghua Lu. 2019. Deep contextualized biomedical abbreviation expansion. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 88–96, Florence, Italy. Association for Computational Linguistics.

Maël Jullien, Marco Valentino, and André Freitas. 2024. SemEval-2024 task 2: Safe biomedical natural language inference for clinical trials. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*. Association for Computational Linguistics.

Maël Jullien, Marco Valentino, Hannah Frost, Paul O'regan, Donal Landers, and André Freitas. 2023. SemEval-2023 task 7: Multi-evidence natural language inference for clinical trial data. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2216–2226, Toronto, Canada. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019a. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019b. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, page 552–561. AAAI Press.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019a. Improving multi-task deep neural networks via knowledge distillation for natural language understanding.

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

Anastasios Nentidis, Konstantinos Bougiatiotis, Anastasia Krithara, and Georgios Paliouras. 2020. *Results of the Seventh Edition of the BioASQ Challenge*, page 553–568. Springer International Publishing.

Long N. Phan, James T. Anibal, Hieu Tran, Shaurya Chanana, Erol Bahadroglu, Alec Peltekian, and Grégoire Altan-Bonnet. 2021. Scifive: a text-to-text transformer model for biomedical literature.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text.

Alexey Romanov and Chaitanya Shivade. 2018. Lessons from natural language inference in the clinical domain. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1586–1596, Brussels, Belgium. Association for Computational Linguistics.

Benjamin Shickel, Patrick James Tighe, Azra Bihorac, and Parisa Rashidi. 2018. Deep ehr: A survey of recent advances in deep learning techniques for electronic health record (ehr) analysis. *IEEE Journal of Biomedical and Health Informatics*, 22(5):1589–1604.

Meina Song, Wen Zhao, and E. HaiHong. 2020. Kganet: a knowledge graph attention network for enhancing natural language inference. *Neural Comput. Appl.*, 32(18):14963–14973.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3229–3238, Online. Association for Computational Linguistics.

Sinong Wang, Han Fang, Madian Khabsa, Hanzi Mao, and Hao Ma. 2021. Entailment as few-shot learner.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Lingfei Wu, Yu Chen, Kai Shen, Xiaojie Guo, Hanning Gao, Shucheng Li, Jian Pei, and Bo Long. 2022. Graph neural networks for natural language processing: A survey.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2020. Xlnet: Generalized autoregressive pretraining for language understanding.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2022. LinkBERT: Pretraining language models with document links. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8003–8016, Dublin, Ireland. Association for Computational Linguistics.

A Graph Attention Network

The dependency tree can be represented by a graph structure where each node is a word and the edges between them are represented by the dependency relation, e.g., nominal subject, adverbial modifier, etc. Following Wang et al. (2020), given a neighborhood of a node $\mathcal{N}_i$, the node embeddings can be iteratively updated using multi-head attention (with K attentional heads) as

$$h_{att_i}^{l+1} = concat_{k=1}^{K} \sum_{j \in \mathcal{N}_i} \alpha_{ij}^{lk} W_k^l h_j^l, \quad (1)$$

$$\alpha_{ij}^{lk} = attention(i, j), \quad (2)$$

where $h_{att_i}^{l+1}$ is the attention head of node-i at layer $l+1$ and α_{ij}^{lk} is the normalized attention coefficient computed by the k-th attention at layer l and W_k^l is an input transformation matrix.

In addition to the attention head of word-i a relational head is also computed for this node as

$$h_{rel_i}^{l+1} = concat_{m=1}^{M} \sum_{j \in \mathcal{N}_i} \beta_{ij}^{lm} W_m^l h_j^l, \quad (3)$$

$$g_{ij}^{lm} = \sigma(relu\left(r_{ij} W_{m1} + b_{m1}\right) W_{m2} + b_{m2}) \quad (4)$$

$$\beta_{ij}^{lm} = exp(g_{ij}^{lm}) / \sum_{j \in \mathcal{N}_i} exp(g_{ij}^{lm}) \quad (5)$$

where r_{ij} denotes the relation embedding between node-i and j and M is the number of relational heads. The final representation of each word (node) is a concatenation of the attention and relational embeddings:

$$x_i^{l+1} = concat(h_{att_i}^{l+1}, h_{rel_i}^{l+1}) \quad (6)$$

$$h_i^{l+1} = relu\left(W_{l+1} x_i^{l+1} + b_{l+1}\right) \quad (7)$$