GATology for Linguistics: Syntactic Dependencies and Complementarity

Anonymous ACL submission

Abstract

Graph Attention Network (GAT) is a novel graph neural network that can process and represent types of different linguistic information using a graph structure. Although GAT and syntactic knowledge can primarily be used in downstream tasks and help in performance improvement, there is still a lack of discussion on what syntactic knowledge GAT is good at learning compared to other neural networks. Therefore, we investigate the robustness of GAT for syntactic dependency prediction in three different languages in terms of attention heads and the number of model layers. We can obtain optimal results when the number of attention heads increases and the number of layers is 2. We also use paired t-test and F1-score to test the prediction of GAT and the pre-trained model BERT fine-tuned by the Machine Translation (MT) task for syntactic dependencies. We analyze their differences in syntactic dependencies, which can lead to syntactic complementarity in their predictions and the possibility of them working together on downstream tasks. We find that GAT is competitive in syntactic dependency prediction, producing good syntactic complementarity with BERT fine-tuned to MT in most cases, while BERT specifically fine-tuned to the dependency prediction task produces better results than GAT.

1 Introduction

The attention mechanism, which most state-of-the-art models use, can effectively capture potential links between input texts, as demonstrated by the Transformer model (Vaswani et al., 2017) in Machine Translation (MT). The graph convolution network can be an extensible, supervised learning network for graph-structured data, which extends the choice of convolutional architectures through spectral and spatial graph convolution. (Veličković et al., 2017) propose the Graph Attention Network (GAT) inspired by the attention mechanism. The shared edge mechanism makes GAT independent of the structure of the global graph, and the attention mechanism also empowers it to compute the importance of different neighbors on the graph, which is easily used in transductive and inductive learning. Syntactic dependency in natural language processing is the mainstream way of analyzing sentence structure, using syntactic tree structures to represent the dependency relationships between words in a sentence. However, the representation of syntactic dependencies has been mainly represented by models such as LSTM or GRU (Zhang et al., 2019a; Hao et al., 2019; Liu et al., 2021). The cumbersome representation process and the consumption of computational resources have limited the application of syntactic knowledge in downstream tasks. GAT simplifies and streamlines the representation of syntactic relationships, allowing separate linear information and linguistic knowledge in sentences to be linked via graphs and applied to various downstream tasks.

Combining the representation of GAT with the widely utilized pre-trained model BERT (Devlin et al., 2019) makes it possible to achieve performance breakthroughs in the downstream tasks (Huang et al., 2020; Li et al., 2022). However, it is unclear why syntactic knowledge incorporated and represented by GAT can work effectively with BERT. Increasing the interpretability of GAT in terms of syntactic knowledge can contribute to better natural language processing, both for downstream tasks which require syntactic knowledge and for the combination of pre-trained models, including but not limited to BERT. Therefore, in this work, we investigate the predictions of GAT on syntactic knowledge. We select syntactic dependencies of three different languages as prediction targets to test how the number of attention heads and layers of GAT is robust to syntactic dependencies. Second, we add a pre-trained model BERT which is fine-tuned for the MT task. The differences between GAT and BERT in syntactic dependencies are compared by paired t-test and F1-score to analyze their syntactic complementarity. Our main contributions are as follows:

- We investigate which configurations of attention heads and model layers work best for GAT for syntactic dependency learning in three languages. We find that increasing the number of attention heads can help GAT to be optimal in syntactic dependency prediction, and the best prediction results are obtained for all languages when the number of model layers is 2, which is not common knowledge that the deeper, the better.

- We evaluate the predictions of GAT and the pre-trained model BERT for typical syntactic dependencies, interpret the discrepancies in their predictions as syntactic complementarity, and discuss the possibility of their syntactic cooperation in MT tasks. We find that GAT not only outperforms BERT fine-tuned for MT tasks, such as "***amod***" for Chinese, "***advmod***" for German, and "***cop***" for Russian but is also competitive for syntactic knowledge learning in most cases. The discrepancies between GAT and BERT in gaining syntactic knowledge suggest the potential of syntactic complementarity.

2 Related Work

In natural language processing, graphs can represent linguistic knowledge, which carries explicit semantic and syntactic information. GAT is a graph network that constructs a graph over a spatial domain using an attention mechanism, which generates new characteristics for each node by aggregating information from nearby nodes and distinguishing the importance of neighbors. As it can be applied to inductive and transductive learning (Salehi and Davulcu, 2019; Busbridge et al., 2019), it has garnered considerable attention. Since GAT can supplement linguistic knowledge in different downstream tasks (Lyu et al., 2021; Huang and Carley, 2019), and its fusion with the pre-trained model BERT in downstream tasks is possible and has attracted the majority of the focus in the study. (Huang et al., 2020) inject syntactic cognitive knowledge into the model using GAT's representation of syntactic knowledge and BERT's pre-trained knowledge, which results in better interaction between context and aspectual words. In the span-level emotion cause analysis task, (Li et al., 2021) use the graph attention network to collect structural information about contexts while using BERT to obtain representations of emotions and contexts. Graph features and word embeddings are used to obtain semantic and syntactic information to classify the comparative preference between two given entities (Ma et al., 2020). However, most of the work focuses only on the representation and application of linguistic knowledge of GAT in downstream tasks and still lacks to investigate its learning of syntactic dependencies in the model structure. What is the contribution of model layers and attention heads to syntactic dependency learning? (Brody et al., 2021) proposes a more expressive dynamic attention, but lacks tests of linguistic knowledge. While integrating GAT and BERT in downstream tasks can bring performance gains, it is not yet clear how they contribute to each other in terms of syntactic dependencies. Most of the work has focused on the discussion and exploration of the linguistic knowledge of BERT (Clark et al., 2019; Papadimitriou et al., 2021), but the learning of the linguistic knowledge of GAT is still unclear. The application of GAT to MT tasks remains largely unexplored. Although some works try to use syntactic knowledge for MT tasks (Peng et al., 2021; McDonald and Chiang, 2021), they do not discuss the possibilities of GAT. (Dai et al., 2022) points out that BERT acts as an MT engine for the encoder to produce low-quality translations when translating sentences with partially syntactic structures, although BERT knows syntactic knowledge. The syntactic knowledge that GAT needs to learn comes mainly from parser or the gold corpus, and it does not need to focus on additional knowledge, as opposed to BERT, which needs to analyze more in the tasks. Suppose GAT can learn syntactic knowledge and perform more competitively than BERT fine-tuned for MT tasks. In that case, one conjecture is that if effectivity representation of syntactic knowledge in GAT can be used to improve translation quality with BERT, it may lead to a breakthrough in MT tasks and more interpretability of linguistic knowledge.

3 Methodology

3.1 Syntactic Learning through Attention Heads and Layers

We use GAT (Brody et al., 2021) as our experimental model. The model is more powerful and robust through dynamic attention compared with the standard GAT (Veličković et al., 2017). The node features given to a GAT layer are $X = [x_1, x_2, x_3, \dots x_i, x_{i+1}], x_i \in \mathbb{R}^F$, where x_{i+1} is the total number of nodes, F is the hidden state of each node given. The Equation (1) summarises the attention mechanism of the GAT.

$$h_i^{out} = \mathop{\|}_{k=1}^{K} \sigma \left(\sum_{j \in N_i} \alpha_{ij}^k W^k x_j \right) \tag{1}$$

$$\alpha_{ij}^k = \frac{exp(a^T f(W^k[x_i \parallel x_j]))}{\sum_{v \in N_i} exp(a^T f(W^k[x_i \parallel x_v]))} \tag{2}$$

1-hop neighbors $j \in N_i$ for node i, $\mathop{\|}_{k=1}^{K}$ means the K multi-head attention outputs are concatenated in this term, σ is a sigmoid function, h_i^{out} is the output hidden state of the node i. In Equation (2), α_{ij}^k is an attention coefficient between node i and j with the attention head k, W^k is linear transformation matrix, a is the context vector during training, and $f(\cdot)$ is LeakyReLU nonlinearity function (Maas et al., 2013). For simplicity, the feature propagation in GAT can be written as $H_{l+1} = GAT(H_l, A; \Theta_l)$, where H_{l+1} is the stacked hidden states of all input nodes at layer l, $A \in \mathbb{R}^{n \times n}$ is the graph adjacency matrix in GAT. Θ_l is the model parameters at that layer.

We treat each word in a sentence as a graph node, and the edges between the nodes are derived from the golden syntactic dependencies in the Parallel Universal Dependencies (PUD) corpus, and the GAT needs to learn and predict the types of syntactic dependencies of the edges between the nodes. Although syntactic dependencies in linguistics are unidirectional from parent to child, we think of the edges in the graph created by GAT as being of two different kinds, from parent to child and from child to parent, respectively. This is due to the fact that, despite being connected, neighboring nodes have different significance depending on whether the current node is acting as a parent or child, and GAT must take into account and learn the significance of neighboring nodes in order to ascertain the syntactic dependencies that must be predicted at the time. Since PUD is a corpus containing golden linguistic knowledge, such as golden lexical information, syntactic dependencies, and other linguistic morphological knowledge, we do not rely on any linguistic parser to generate and extract syntactic dependencies. We select Chinese (Zh), German (De), and Russian (Ru) as the three languages and their syntactic dependencies for the tests in order to reduce the problems related to single-language experiments. The PUD corpus for each language has 1000 sentences that are always arranged in the same order (UD Chinese PUD[1], UD Russian PUD[2], UD German PUD[3]). Because of syntactic dependencies' restrictions, a sentence's sequential input takes on a topological structure generally referred to as a syntactic tree, providing information on the structure of a graph.

We increase the number of attention heads and model layers of GAT, add part-of-speech information to as the additional syntactic knowledge of node features, and evaluate its performance in predicting syntactic dependencies of languages under various collocations. Given the classification imbalance of different syntactic dependencies in the PUD corpus, we use F1-score as an evaluation metric to reflect the prediction performance of GAT on syntactic dependencies as much as possible by considering the effects of precision and recall. We report the overall and individual prediction performance of syntactic dependencies.

In experiments, GAT's attention heads are set to 2, 4, 6, and 8, respectively. Moreover, the model depth contains a different number of layers, which are 2, 3, 4, 5, and 6. We record the variation and trend of syntactic dependency learning of GAT for different languages with these parameters paired with each other. All languages include a randomly divided training set, validation set, and test set with the number of sentences of 800, 100, and 100, respectively. Word embeddings = 768, dropout = 0.2, optimizer = Adam, and learning rate = 2e-5.

3.2 Syntactic Dependency Complementarity with Fine-tuned BERT

BERT is often used as a popular pre-trained model for downstream tasks in natural language processing and has achieved significant performance breakthroughs (Reimers and Gurevych, 2019; Zhang et al., 2019b). GAT and BERT use attention mechanisms as essential feature extraction, which makes their combination in downstream tasks has become

[1]https://github.com/UniversalDependencies/UD_Chinese-PUD

[2]https://github.com/UniversalDependencies/UD_Russian-PUD

[3]https://github.com/UniversalDependencies/UD_German-PUD

possible. Most of the tasks are at the application level to discuss how GAT works with BERT in downstream tasks and what is the performance gain for the downstream tasks. However, there is still a lack of investigation at the level of linguistic knowledge as to why GAT can help BERT in downstream tasks in terms of syntactic knowledge and thus improve performance. In MT tasks, it is achievable that syntactic knowledge can improve translation quality, but the work of GAT and BERT in MT tasks is less discussed. (Dai et al., 2022) point to the impact of syntactic dependencies on translation quality in MT engines with BERT. Given the feasibility of the GAT representation of syntactic dependencies, it is possible that a more efficient representation and complementation of syntactic dependencies can improve the poor translation quality caused by the BERT translation engines. We, therefore, explore the interpretability and potential for collaboration between BERT and GAT in downstream tasks by examining the differences between them in terms of syntactic dependencies, which we refer to as syntactic complementarity.

Following (Dai et al., 2022), we select Chinese (Zh), Russian (Ru), and German (De) as the experimental languages and the different BERT-base versions for their corresponding languages (Kuratov and Arkhipov, 2019; Cui et al., 2021; Devlin et al., 2019), BERTs are fine-tuned by the MT task as the comparison objects to study the syntactic dependencies possessed by the fine-tuning in the MT scenario. Although the pre-training strategies of BERTs are different for different languages, the model structure is the same. The United Nations Parallel Corpus (UNPC) (Ziemski et al., 2016) trains the Chinese and Russian MT engines, whereas Europarl (Koehn, 2005) trains the German MT engine. BERTs are used as encoders in MT engines for Zh→En, De→En, and Ru→En translations.

After completing the fine-tuning of the BERTs for MT tasks, we extract the BERTs in the translation systems, a simple fully-connected layer is then added to the last layer of the fine-tuned BERTs, and all parameters are frozen except for the last fully-connected layer to prevent learning new syntactic knowledge from the syntactic dependency data set. However, BERT predicts differently from GAT because it does not know the child word of the present parent word. Since BERT knows syntactic knowledge and syntax tree can be detected (Htut et al., 2019; Manning et al., 2020), we do not add any additional algorithms, for example, to specify all the parent and child words in the sentence. This simulates syntactic knowledge in downstream tasks as closely as possible. The correctness of the prediction of syntactic dependencies can indirectly corroborate the difference between the syntactic tree formed and the golden syntactic tree. Unlike GAT, which always focuses on syntactic knowledge, BERT has syntactic knowledge as part of what it needs to learn in the MT task, and BERT's learning of this syntactic knowledge may not be sufficient. We also add another BERT but updated the PUD corpus's parameters as a reference. Knowing how well the BERT performs is necessary when it focuses on syntactic knowledge, which would be considered the best performance. If GAT can beat it on specific syntactic dependencies, this implies that the syntactic knowledge of GAT is competitive and potential. We want to investigate the complementarity and possibility of syntactic dependencies between the GAT and the BERT fine-tuned by downstream MT tasks and syntactic tasks. We evaluate the complementarity of GAT and BERT in terms of syntactic dependencies in overall and individual terms. First, a paired t-test is used to compare the overall difference between the two models for predicting syntactic dependency and determining whether there is significant variation. Second, considering the diversity and complexity of syntactic dependencies, we also discuss the performance variation of individual syntactic dependencies by F1-score, examining how different models learn different sentence constituents.

We select the number of attention heads and layers with the highest overall prediction scores as the experimental parameters for the GAT. All languages have two layers in the GAT, although Zh has six attention heads and Ru and De each have four. The strategy of GAT for predicting syntactic dependencies is the same as in the previous experiment, and the PUD corpus is the data set for BERT and GAT. We add K-fold cross-validation and ensure that the training and test sets are the same for both models, and the F1-score is still used as the evaluation metric for this experiment to maximize the consistency of the two models on the prediction task. The number of the training and test set of the PUD corpus is 850 and 150. The word embeddings for GAT and BERT are 768, the other settings are kept the same as in Experiment 3.1.

4 Results

4.1 Syntactic Predictions with Attention and Layers

As shown in Table 1, we observe that a specific number of attention heads gives the optimal prediction performance of GAT, and arbitrarily increasing the number of attention heads may also lead to a decrease in prediction. In models like Transformer and BERT, it has been demonstrated that increasing the number of attention heads can improve the model's capacity to extract and represent features. Increasing the number of attention heads in GAT by a certain amount can result in improved profits. However, this does not imply that further increases are helpful to the model. For instance, the optimal performance for Ru and De is reached with two layers and four attention heads. In Zh, however, six or eight attention heads yield better outcomes than that of 2 with two layers. We believe this is associated with the input structure of the model. Each word in a sentence can contribute to feature extraction when sequential input models such as Transformer are used, increasing attention heads can collect and learn potential links between words in various sub-spaces, leading to improved representations. The sequence input is transformed into a graph-based topology in GAT. We believe that unlike with sequential input, where it is necessary to allocate attention to discuss the potential contributions of each word, the observed range of each word in the sentence is already restricted and instructive due to the structure of syntactic dependencies. Thus the increase in the number of attention heads is far less straightforward than the gain of, for example, the Transformer model. Its multiple heads of attention may also suffer from redundancy, which impairs the learning of syntactic dependencies.

We notice that the GAT prediction for syntactic dependencies is acceptable with a reasonable number of attention heads and layers. However, experiments also reveal that increasing the number of layers of the neural network causes the overall prediction to be significantly impaired, and GAT loses learning and prediction of some syntactic dependencies, as shown in Table 2*. As the number of layers increases, GAT fails to learn some syntactic dependencies, as evidenced by the F1-score dropping entirely to 0. This phenomenon appears in all three languages in the experiment. We record the number of syntactic dependencies with an F1-score of 0 under the different number of attention heads in each layer for the three languages. As shown in the Figure 1, they are concentrated in the deep layers, and the increase in attention heads does not alleviate this phenomenon. This is different from the intuition that the deeper the model depth, the better the performance. The increase in layers does not bring more significant performance, which may be because the increase in the number of layers of the graph network causes the nodes to lose their properties or may absorb some irrelevant information leading to degradation of the model performance. Also, we observe that GAT produces a consistent learning performance for specific syntactic dependencies when presented with different languages, they are "***advmod***", "***case***", "***cc***", "***mark***", "***nsubj***", "***punct***". When increasing the number of attention heads and the depth of the model layers, they can maintain relatively high prediction scores, and the predicted outcome of an F1-score of 0 does not occur. The model can acquire some common underlying linguistic knowledge in a deeper layer across multiple languages, which means that GAT is more sensitive to such syntactic knowledge and capturing the same syntactic knowledge across languages is possible for deep graph neural networks.

	2 Heads	**4 Heads**	**6 Heads**	**8 Heads**
Zh				
2 Layers	0.63	0.62	**0.64**	**0.64**
3 Layers	0.64	0.61	0.62	0.63
4 Layers	0.56	0.58	0.64	0.49
5 Layers	0.49	0.50	0.51	0.50
6 Layers	0.37	0.40	0.33	0.33
Ru				
2 Layers	0.58	**0.61**	0.47	0.56
3 Layers	0.45	0.55	0.54	0.53
4 Layers	0.44	0.47	0.56	0.57
5 Layers	0.42	0.52	0.46	0.49
6 Layers	0.41	0.36	0.31	0.33
De				
2 Layers	0.64	**0.67**	0.64	0.56
3 Layers	0.60	0.56	0.56	0.57
4 Layers	0.56	0.50	0.53	0.53
5 Layers	0.58	0.61	0.50	0.47
6 Layers	0.48	0.49	0.48	0.42

Table 1: Overall GAT predictions of syntactic relationships for three languages with different numbers of attention heads and layers. The increased number of attention heads and layers does not result in a performance advantage.

*The appendix contains all experimental results for the three languages.

					Zh						
Layers	**Heads**	**appos**	**advmod**	**clf**	**case**	**cc**	**dep**	**mark**	**nsubj**	**obj**	**punct**
2	2	0.60	0.90	0.87	0.98	0.99	0.64	0.99	0.64	0.53	0.99
	4	0.55	0.90	0.82	0.99	0.99	0.63	0.99	0.66	0.58	0.99
	6	0.61	0.91	0.89	0.99	0.99	0.66	0.98	0.68	0.61	0.99
	8	0.58	0.90	0.83	0.99	0.99	0.62	0.99	0.67	0.59	0.99
3	2	0.54	0.90	0.88	0.99	0.99	0.64	0.90	0.68	0.63	0.99
	4	0.57	0.91	0.86	0.59	0.99	0.64	0.96	0.66	0.58	0.99
	6	0.61	0.90	0.88	0.59	0.99	0.66	0.96	0.66	0.60	0.99
	8	0.60	0.91	0.90	0.59	0.99	0.66	0.96	0.68	0.63	0.99
4	2	0.55	0.89	0.68	0.97	0.99	0.64	0.95	0.64	0.55	0.99
	4	0.60	0.90	0.66	0.98	0.99	0.65	0.98	0.69	0.62	0.99
	6	0.56	0.91	0.69	0.99	0.99	0.68	0.92	0.67	0.60	0.99
	8	0	0.90	0	0.98	0.80	0.64	0.96	0.62	0.44	0.98
5	2	0.52	0.90	0	0.56	0.99	0	0.93	0.65	0.56	0.99
	4	0.62	0.90	0	0.92	0.75	0	0.88	0.66	0.60	0.99
	6	0.54	0.90	0	0.88	0.99	0	0.91	0.65	0.58	0.99
	8	0	0.89	0	0.97	0.99	0	0.84	0.56	0.52	0.99
6	2	0	0.83	0	0.81	0.99	0	0.82	0.42	0	0.98
	4	0	0.86	0	0.88	0.77	0	0.87	0.50	0	0.98
	6	0	0.84	0	0.83	0.75	0	0.82	0.47	0	0.96
	8	0	0.86	0	0.89	0.73	0	0.84	0.51	0	0.99

Table 2: Part of Chinese syntactic dependencies is shown. As the number of layers increases, GAT gradually loses its prediction ability for some syntactic dependencies in Chinese. Some syntactic dependencies are not significantly affected by the number of layers increased that the F1-score drops to 0.

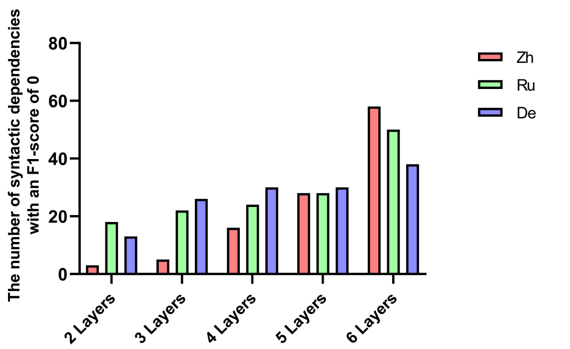

Figure 1: The number of F1-score dropped to 0 made by the GAT in different layers with a different number of attention heads. Although each layer has 2, 4, 6, and 8 attention heads, increasing the number of layers invariably results in more failures for syntactic knowledge learning.

4.2 Complementarity of Syntactic Dependencies with BERT

As shown in Table 3, the paired t-test shows that the p-values for all three languages are less than the significance level when the outliers of the syntactic dependencies are removed. The null hypothesis (H_0) that the two models would be equally effective in predicting syntactic dependencies is rejected, and the significant differences in syntactic dependencies between the GAT and the BERT fine-tuned by machine translation (MT-B) are statistically significant.

Investigating the learning of each syntactic dependency from an F1-score perspective as shown in Table 4, we find that GAT dominates the prediction of the vast majority of syntactic dependencies, with only a small proportion losing out to MT-B. We argue that although BERT is fine-tuned by the MT task, its learning of syntactic dependencies is inadequate in this case. BERT is likely to produce similar results under fine-tuning in other downstream tasks, since many works have shown that incorporating syntactic dependency through GAT with BERT in downstream tasks can improve performance. The complementation of syntactic dependencies by GAT can hardly have a substantial impact on downstream tasks if the syntactic knowledge of BERT does not decrease to varying degrees after fine-tuning. From the study of (Dai et al., 2022): when detection of dependencies deteriorates, MT quality drops. Association between quality and the relations of ”***appos***”, ”***case***”, ”***flat***”, ”***flat:name***”, and ”***obl***” for all languages, ”***dep***”, ”***advcl***”, and ”***mark***” for Chinese, ”***parataxis***” and ”***nummod***” for Russian, ”***compound***” and ”***advcl***” for German. Also, relation of ”***root***” as the sentence’s main predicate [**] is the root node and is used to express the sentence’s main substance. Despite GAT and BERT make predictions in different ways, and it cannot be linked to the decrease

[**]One of the orphaned dependents gets promoted to the root position if the main predicate is absent.

Languages	Observations	Sample size	Significance level	Mean	STDev	T-value	P-value
Zh	MT-B GAT	29	0.05	0.6 0.7	0.2 0.3	4.39	¡0.001
Ru	MT-B GAT	23		0.7 0.7	0.2 0.3	3.555	0.001
De	MT-B GAT	26		0.6 0.7	0.2 0.3	3.682	0.001

Table 3: Paired t-tests are used to compare the findings of GAT and BERT fine-tuned by machine translation on syntactic dependency prediction. There is a significant difference in the prediction results between the two models.

	Zh				Ru				De			
	#	MT-B	GAT	UD-B	#	MT-B	GAT	UD-B	#	MT-B	GAT	UD-B
acl	19	0	0	*0*	256	**0.514**	0.392	*0.854*	20	0	0	*0*
acl:relcl	448	0.478	**0.913**	0.836	160	**0.444**	0.105	*0.960*	271	**0.654**	0.605	*0.912*
advcl	516	0.274	**0.376**	*0.728*	197	0.320	**0.334**	*0.842*	220	0.410	**0.495**	*0.832*
advmod	1332	0.660	**0.909**	*0.946*	914	0.830	**0.902**	*0.964*	1103	0.618	**0.984**	0.958
amod	420	0.388	**0.919**	0.874	1791	0.880	**0.979**	*0.982*	1089	0.656	**0.935**	*0.976*
appos	248	**0.522**	0.423	*0.740*	121	0.420	**0.436**	*0.570*	265	0.344	**0.561**	*0.786*
aux	686	0.746	**0.875**	*0.966*	42	**0.884**	0.836	*0.932*	365	0.820	**0.862**	*0.972*
aux:pass	79	**0.842**	0	*0.970*	128	0.962	**0.988**	0.968	230	0.832	**0.934**	*0.965*
case	1319	0.756	**0.963**	0.928	2121	0.924	**0.983**	0.981	2053	0.844	**0.994**	0.986
case:loc	351	0.664	**0.779**	*0.954*	-	-	-	-	-	-	-	-
cc	283	0.842	**0.990**	0.938	599	0.950	**0.969**	*0.988*	724	0.822	**0.981**	0.972
ccomp	403	0.174	**0.277**	*0.656*	132	0.506	**0.536**	*0.752*	169	**0.336**	0.196	*0.704*
clf	357	**0.804**	0.737	*0.980*	-	-	-	-	-	-	-	-
compound	1777	0.604	**0.881**	*0.886*	9	0	0	0	251	0.488	**0.496**	*0.850*
conj	383	0.484	**0.976**	0.842	965	0.728	**0.862**	*0.920*	842	0.584	**0.673**	*0.912*
cop	251	0.552	**0.962**	0.842	87	0.762	**0.983**	0.830	274	**0.786**	0.755	*0.954*
dep	397	0.276	**0.556**	*0.742*	-	-	-	-	-	-	-	-
det	338	0.710	**0.963**	0.956	476	0.866	**0.997**	0.974	2771	0.906	**0.996**	0.980
expl	-	-	-	-	7	0	0	*0.890*	90	**0.760**	0.319	*0.982*
fixed	-	-	-	-	222	**0.586**	0.277	*0.846*	7	0	0	*0*
flat	91	0.674	**0.867**	*0.965*	61	0.174	**0.483**	*0.538*	14	0.050	**0.271**	*0.344*
flat:foreign	-	-	-	-	97	0.320	**0.903**	0.892	-	-	-	-
flat:name	142	0.778	**0.897**	*0.936*	222	**0.890**	0.588	*0.986*	164	0.502	**0.844**	0.762
iobj	15	0	0	*0.134*	190	**0.508**	0	*0.730*	95	**0.430**	0	*0.874*
mark	291	0.536	**0.980**	0.905	287	0.776	**0.867**	0.854	459	0.822	**0.992**	0.980
mark:adv	22	**0.990**	0.400	*0.970*	-	-	-	-	-	-	-	-
mark:prt	338	**0.414**	0.237	*0.838*	-	-	-	-	-	-	-	-
mark:relcl	626	**0.862**	0.756	*0.944*	-	-	-	-	-	-	-	-
nmod	702	0.36	**0.919**	0.826	1934	0.696	**0.870**	*0.920*	1102	0.580	**0.749**	*0.888*
nsubj	1776	0.608	**0.612**	*0.906*	1362	**0.726**	0.666	*0.936*	1481	0.672	**0.678**	*0.950*
nsubj:pass	70	**0.138**	0	*0.766*	186	**0.272**	0	*0.904*	207	**0.440**	0	*0.974*
nummod	809	0.844	**0.993**	0.988	181	0.530	**0.690**	*0.732*	226	0.758	**0.808**	*0.926*
obj	1526	0.482	**0.558**	*0.858*	749	**0.550**	0.518	*0.928*	895	**0.592**	0.485	*0.960*
obl	578	0.232	**0.846**	0.738	1465	0.670	**0.911**	*0.914*	1344	0.604	**0.801**	*0.918*
obl:agent	22	**0.714**	0	*0.888*	12	0	0	*0.520*	-	-	-	-
obl:patient	39	0	0	*0.986*	-	-	-	-	-	-	-	-
obl:tmod	214	**0.504**	0.104	*0.816*	-	-	-	-	10	**0.618**	0.216	*0.832*
parataxis	-	-	-	-	195	**0.520**	0.200	*0.706*	68	0	0	*0.524*
punct	2902	0.748	**0.990**	0.990	2977	0.958	**0.990**	0.990	2770	0.928	**0.999**	0.981
root	1000	0.486	**0.968**	0.894	1000	0.880	**0.994**	0.982	1000	0.704	**0.932**	*0.982*
xcomp	476	0.278	**0.437**	*0.804*	331	0.580	**0.634**	*0.880*	190	**0.464**	0.291	*0.820*

Table 4: Prediction scores of machine translation fine-tuned BERT (MT-B) and GAT and BERT fine-tuned for PUD corpus (UD-B) in syntactic dependencies. GAT is more competitive than MT-B in predicting syntactic dependencies, shown in bold format, and some syntactic dependencies can surpass UD-B, shown in the non-italic format in the column of UD-B.

in MT quality (because it is present in every sentence), the fact that GAT and BERT fine-tuned for the PUD corpus (UD-B) are better in detecting it means that BERT fine-tuned for the MT task lack the ability to detect. GAT has better predictive performance in most cases for all languages, it is possible that translation quality can be further improved if these mentioned syntactic dependencies that affect translation quality are targeted to be supplemented by GAT. If all syntactic knowledge can be incorporated into a translation system through GAT, clearer sentence structure may lead to more fluent translation results. Based on the prediction of syntactic dependencies, we believe that GAT and MT-B in MT tasks are complementary in terms of syntactic dependencies and are highly competitive in predicting at least most syntactic dependencies.

UD-B performs best on the F1-score, but it does not substantially outperform the GAT with only two layers. In most cases, their prediction scores are close to each other. Given that BERT is pre-trained with a large amount of data and is more complicated than GAT regarding the number of attention heads and the model structure, the prediction results are not surprising. However, the GAT still outperform UD-B for some relations, such as "***amod***", "***conj***" for Chinese, "***advmod***", "***flat:name***" for German, and "***cop***" for Russian. We record the common relations that outperformed UD-B in prediction in all three languages: "***case***", "***mark***", "***det***", and "***cc***". This means that GAT can learn the four mentioned syntactic dependencies efficiently and can successfully predict and outperform BERT without pre-training. Although the prediction results are different for all relations, at least we can assume that GAT is more learned for these four syntactic dependencies and can have potential syntactic complementarity with BERT.

The majority of syntactic dependencies number fewer than 500 indicating that the training sample cost of GAT is not expensive, and the same number of training samples can outperform MT-B in the majority of syntactic dependencies and UD-B in a few cases. How to learn linguistic knowledge from a limited number of training samples can be a challenge for both BERT and GAT. Pre-training and more robust model structures allow BERT to effectively alleviate this problem when faced with learning from small samples. However, GAT may be unable to learn them. Examples are "***acl***" for Zh and De, "***aux:pass***" for Chinese, and "***obl:agent***" for Zh and Ru. Not only that, the learning of specific syntactic dependencies is difficult for GAT. "***iobj***" and "***nsubj:pass***" in the three languages cannot be predicted by GAT. These two relations are consistent in linguistic knowledge classification, with core arguments as their functional categories and nominals as their structural categories. GAT may lack sufficient learning of the syntactic subjects of indirect objects and passive clauses. In most cases, the lightweight and inexpensive GAT shows acceptable performance in syntactic knowledge learning relative to BERT fine-tuned for the MT task, and it is possible to complement BERT's deficiencies in syntactic dependencies in the MT task. Furthermore, GAT can outperform BERT fine-tuned for syntactic dependencies on specific dependencies, the same pre-trained GAT may lead to a superior representation of linguistic knowledge in the future.

5 Conclusions

This work investigates the effect of the number of attention-head and model layers in GAT on syntactic dependency learning and whether there is syntactic complementarity with the pre-trained model BERT. We find that appropriately increasing the number of attention-head in GAT does allow for better model optimization, despite the possible redundancy of these attention heads. However, contrary to our previous knowledge, the increase in the number of model layers produces an F1-score of 0 for predicting syntactic dependencies. The reason for this is unclear, but according to experimental results, GAT with a layer of 2 is the most friendly for syntactic-dependent learning. Moreover, paired t-tests and F1-score suggest that GAT is capable of syntactic complementarities at different levels than BERT fine-tuned by MT and syntactic tasks. UD-BERT specifically trained for the UD prediction task is overall better than GAT, especially for the rare syntactic categories, as it benefits from seeing many more examples of them at the pre-training stage, while GAT only learns from the explicit trees. Still, GAT is competitive for syntactic dependency learning and can be incorporated into downstream tasks, and these syntactic complementarities between BERT and GAT may have the potential for the fusion of pre-trained models and graph neural networks. Future work includes further investigating the possibility of using GAT's representation of syntactic dependencies to improve the translation quality of translation engines with BERT.

References

Shaked Brody, Uri Alon, and Eran Yahav. 2021. How attentive are graph attention networks? *arXiv preprint arXiv:2105.14491*.

Dan Busbridge, Dane Sherburn, Pietro Cavallo, and Nils Y Hammerla. 2019. Relational graph attention networks. *arXiv preprint arXiv:1904.05811*.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does BERT look at? an analysis of BERT's attention. *arXiv preprint arXiv:1906.04341*.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for Chinese BERT. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3504–3514.

Yuqian Dai, Marc de Kamps, and Serge Sharoff. 2022. BERTology for machine translation: What BERT knows about linguistic difficulties for translation. In *Proceedings of the Language Resources and Evaluation Conference*, pages 6674–6690, Marseille, France. European Language Resources Association.

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.

Jie Hao, Xing Wang, Shuming Shi, Jinfeng Zhang, and Zhaopeng Tu. 2019. Towards better modeling hierarchical structure for self-attention with ordered neurons. *arXiv preprint arXiv:1909.01562*.

Phu Mon Htut, Jason Phang, Shikha Bordia, and Samuel R Bowman. 2019. Do attention heads in BERT track syntactic dependencies? *arXiv preprint arXiv:1911.12246*.

Binxuan Huang and Kathleen M Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention networks. *arXiv preprint arXiv:1909.02606*.

Lianzhe Huang, Xin Sun, Sujian Li, Linhao Zhang, and Houfeng Wang. 2020. Syntax-aware graph attention network for aspect-level sentiment classification. In *Proceedings of the 28th international conference on computational linguistics*, pages 799–810.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Yuri Kuratov and Mikhail Arkhipov. 2019. Adaptation of deep bidirectional multilingual transformers for Russian language. *arXiv preprint arXiv:1905.07213*.

Gang Li, Chengpeng Zheng, Min Li, and Haosen Wang. 2022. Automatic requirements classification based on graph attention network. *IEEE Access*, 10:30080–30090.

Xiangju Li, Wei Gao, Shi Feng, Daling Wang, and Shafiq R. Joty. 2021. Span-level emotion cause analysis by BERT-based graph attention network. *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*.

Rui Liu, Berrak Sisman, and Haizhou Li. 2021. Graphspeech: Syntax-aware graph attention network for neural speech synthesis. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6059–6063. IEEE.

Bo Lyu, Lu Chen, Su Zhu, and Kai Yu. 2021. Let: Linguistic knowledge enhanced graph transformer for chinese short text matching. In *AAAI*.

Nianzu Ma, S. Mazumder, Hao Wang, and Bing Liu. 2020. Entity-aware dependency-based deep graph attention network for comparative preference classification. In *ACL*.

Andrew L Maas, Awni Y Hannun, Andrew Y Ng, et al. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3. Citeseer.

Christopher D Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Colin McDonald and David Chiang. 2021. Syntax-based attention masking for neural machine translation. In *NAACL*.

Isabel Papadimitriou, Ethan A Chi, Richard Futrell, and Kyle Mahowald. 2021. Deep subjecthood: Higher-order grammatical features in multilingual BERT. *arXiv preprint arXiv:2101.11043*.

Ru Peng, Nankai Lin, Yi Fang, Shengyi Jiang, and Junbo Jake Zhao. 2021. Boosting neural machine translation with dependency-scaled self-attention network. *ArXiv*, abs/2111.11707.

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

Amin Salehi and Hasan Davulcu. 2019. Graph attention auto-encoders. *arXiv preprint arXiv:1905.10715*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Jian Zhang, Xu Wang, Hongyu Zhang, Hailong Sun, Kaixuan Wang, and Xudong Liu. 2019a. A novel neural source code representation based on abstract syntax tree. In *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, pages 783–794. IEEE.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019b. Bertscore: Evaluating text generation with BERT. *arXiv preprint arXiv:1904.09675*.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia. European Language Resources Association (ELRA).

6 Appendices

6.1 Syntactic Predictions with Attention and Layers

We investigate syntactic dependency learning in GAT for Chinese (Zh), Russian (Ru), and German (De) for different numbers of attention heads (A) and layers (L) as shown in Table 5 to Table 9. As some syntactic dependencies in the PUD corpus are uncommon with only a small number of samples, they do not reasonably reflect the learning performance of the model, so we remove them in the experiments. Due to the diversity of language knowledge, the categories of syntactic dependencies may vary between languages.

					Zh					
L-A	acl:relcl	advcl	advmod	amod	appos	aux	case	case:loc	cc	ccomp
2–2	0.82	0	0.90	0.80	0.60	0.90	0.98	0.95	0.99	0
2–4	0.83	0	0.90	0.81	0.55	0.91	0.99	0.94	0.99	0.40
2–6	0.87	0.14	0.91	0.85	0.61	0.91	0.99	0.91	0.99	0.53
2–8	0.84	0.15	0.90	0.80	0.58	0.91	0.99	0.94	0.99	0.30
3–2	0.87	0	0.90	0.84	0.54	0.90	0.99	0.92	0.99	0.66
3–4	0.85	0	0.91	0.83	0.57	0.89	0.59	0.95	0.99	0.38
3–6	0.88	0	0.90	0.87	0.61	0.90	0.59	0.95	0.99	0.66
3–8	0.87	0	0.91	0.85	0.60	0.91	0.59	0.94	0.99	0.64
4–2	0.83	0	0.89	0.80	0.55	0.90	0.97	0.89	0.99	0
4–4	0.87	0	0.90	0.80	0.60	0.90	0.98	0.94	0.99	0
4–6	0.89	0.19	0.91	0.83	0.56	0.90	0.99	0.94	0.99	0.21
4–8	0.83	0	0.90	0.78	0	0.87	0.98	0.95	0.80	0
5–2	0	0.36	0.90	0.74	0.52	0.88	0.56	0.83	0.99	0
5–4	0.91	0.38	0.90	0.76	0.62	0.90	0.92	0	0.75	0
5–6	0.87	0.36	0.90	0.79	0.54	0.87	0.88	0	0.99	0
5–8	0.86	0	0.89	0.80	0	0.86	0.97	0.85	0.99	0
6–2	0.79	0	0.83	0.71	0	0.82	0.81	0	0.99	0
6–4	0.84	0	0.86	0.73	0	0.88	0.88	0	0.77	0
6–6	0	0	0.84	0.59	0	0.86	0.83	0	0.75	0
6–8	0	0	0.86	0	0	0.85	0.89	0	0.73	0
L-A	clf	compound	conj	cop	dep	det	discourse:sp	flat	flat:name	mark
2–2	0.87	0.86	0.99	0.88	0.64	0.97	0.22	0.96	0.88	0.99
2–4	0.82	0.86	0.99	0.95	0.63	0.97	0.22	0.99	0.88	0.99
2–6	0.89	0.87	0.99	0.97	0.66	0.97	0.29	0.96	0.88	0.98
2–8	0.83	0.87	0.99	0.98	0.62	0.97	0.33	0.99	0.88	0.99
3–2	0.88	0.87	0.99	0.94	0.64	0.97	0.22	0.96	0.92	0.90
3–4	0.86	0.85	0.99	0.95	0.64	0.97	0.20	0.96	0.94	0.96
3–6	0.88	0.86	0.99	0.97	0.66	0.97	0	0.96	0.94	0.96
3–8	0.90	0.87	0.99	0.97	0.66	0.97	0.22	0.92	0.97	0.96
4–2	0.68	0.82	0.97	0.91	0.64	0.95	0.18	0.96	0	0.95
4–4	0.66	0.82	0.99	0.97	0.65	0.95	0.22	0.99	0	0.98
4–6	0.69	0.84	0.99	0.97	0.68	0.97	0.29	0.99	0	0.92
4–8	0	0.78	0	0.92	0.64	0.85	0	0.76	0	0.96
5–2	0	0.83	0.99	0.91	0.64	0.84	0.33	0.99	0	0.93
5–4	0	0.81	0	0.97	0	0.84	0.29	0.99	0.80	0.88
5–6	0	0.82	0.99	0.95	0	0.85	0	0.99	0	0.91
5–8	0	0.83	0.86	0.97	0	0.85	0.22	0.81	0.84	0.84
6–2	0	0.83	0.53	0.92	0	0.85	0	0.96	0	0.82
6–4	0	0.76	0	0.94	0	0.83	0	0.73	0	0.87
6–6	0	0.66	0	0.91	0	0.82	0	0.88	0	0.82
6–8	0	0.62	0	0.92	0	0.83	0	0.81	0.72	0.84
L-A	mark:prt	mark:relcl	nmod	nsubj	nummod	obj	obl	obl:tmod	punct	root
2–2	0.68	0.96	0.92	0.64	0.97	0.53	0.79	0.40	0.99	0.98
2–4	0.66	0.97	0.93	0.66	0.98	0.58	0.79	0.42	0.99	0.98
2–6	0.71	0.97	0.92	0.68	0.98	0.61	0.77	0.44	0.99	0.98
2–8	0.70	0.97	0.92	0.67	0.98	0.59	0.80	0.41	0.99	0.98
3–2	0.75	0.98	0.92	0.68	0.98	0.63	0.81	0.42	0.99	0.99
3–4	0.73	0.74	0.73	0.66	0.99	0.58	0.84	0.44	0.99	0.98
3–6	0.69	0.77	0.72	0.66	0.99	0.60	0.79	0.42	0.99	0.98
3–8	0.69	0.79	0.71	0.68	0.99	0.63	0.84	0.53	0.99	0.99
4–2	0	0.97	0.92	0.64	0.97	0.55	0.80	0.34	0.99	0.99
4–4	0	0.96	0.94	0.69	0.99	0.62	0.82	0.37	0.99	0.98
4–6	0.72	0.97	0.92	0.67	0.99	0.60	0.82	0.44	0.99	0.99
4–8	0	0.97	0.90	0.62	0.98	0.44	0.78	0.34	0.98	0.98
5–2	0	0.62	0.72	0.65	0.98	0.56	0	0.36	0.99	0.98
5–4	0	0.97	0.92	0.66	0.86	0.60	0.77	0	0.99	0.99
5–6	0	0.97	0.91	0.65	0.85	0.58	0.73	0.37	0.99	0.98
5–8	0	0.97	0.92	0.56	0.83	0.52	0.73	0	0.99	0.89
6–2	0	0.97	0.89	0.42	0.83	0	0	0	0.98	0
6–4	0	0.97	0.90	0.50	0.86	0	0.64	0	0.98	0.82
6–6	0	0.88	0.68	0.47	0	0	0.66	0	0.96	0.88
6–8	0	0.72	0.80	0.51	0	0	0.66	0	0.99	0.79

Table 5: GAT predictions of syntactic dependence in Chinese.

Zh	
L-A	xcomp
2–2	0.48
2–4	0.54
2–6	0.56
2–8	0.58
3–2	0.63
3–4	0.53
3–6	0.65
3–8	0.68
4–2	0.47
4–4	0.44
4–6	0.56
4–8	0.47
5–2	0.41
5–4	0.53
5–6	0.48
5–8	0
6–2	0
6–4	0
6–6	0
6–8	0

Table 6: GAT predictions of syntactic dependence in Chinese.

					Ru					
L-A	acl	acl:relcl	advcl	advmod	amod	appos	aux	aux:pass	case	cc
2–2	0.54	0	0	0.90	0.98	0.32	0.75	0.96	0.99	0.97
2–4	0.52	0	0.71	0.91	0.98	0.55	0.89	0.96	0.99	0.99
2–6	0.64	0.81	0	0.89	0.98	0.24	0	0	0.98	0.96
2–8	0.54	0	0	0.90	0.98	0.50	0.67	0.92	0.98	0.97
3–2	0.57	0	0	0.90	0.98	0.12	0	0	0.98	0.96
3–4	0.63	0	0.56	0.92	0.98	0.45	0	0	0.98	0.96
3–6	0.63	0.84	0	0.90	0.98	0.48	0	0	0.98	0.96
3–8	0.67	0.72	0	0.91	0.98	0.13	0	0	0.99	0.96
4–2	0.51	0	0	0.92	0.97	0	0	0	0.97	0.84
4–4	0.60	0.64	0	0.89	0.97	0	0.67	0	0.99	0.82
4–6	0.73	0.84	0.39	0.90	0.98	0.65	0	0.86	0.99	0.82
4–8	0.65	0	0	0.92	0.99	0.55	0.44	0	0.99	0.96
5–2	0.57	0	0.23	0.91	0.96	0	0	0	0.97	0.85
5–4	0.67	0.78	0.49	0.91	0.97	0	0	0	0.98	0.82
5–6	0.77	0.75	0.17	0.91	0.97	0.44	0	0	0.97	0.81
5–8	0.56	0	0	0.91	0.96	0.54	0	0.86	0.99	0.86
6–2	0	0	0	0.90	0.96	0	0	0.89	0.94	0.83
6–4	0	0.42	0	0.88	0.88	0	0	0	0.95	0.78
6–6	0.30	0	0	0.88	0.91	0	0	0	0.94	0.79
6–8	0	0	0	0.90	0.96	0	0	0	0.96	0.85
L-A	ccomp	conj	cop	csubj	det	fixed	flat	flat:forign	flat:name	mark
2–2	0.70	0.84	0.96	0	0.99	0.43	0.86	0.87	0.58	0.97
2–4	0.67	0.87	0.99	0	0.99	0.57	0.86	0.92	0.56	0.94
2–6	0.54	0.88	0.58	0	0.98	0	0	0.80	0.52	0.96
2–8	0.57	0.87	0.96	0	0.99	0.50	0.86	0.87	0.64	0.90
3–2	0.50	0.88	0.56	0	0.98	0	0	0.74	0.51	0.93
3–4	0.81	0.90	0.67	0	0.99	0.67	0.86	0.87	0.55	0.94
3–6	0.67	0.89	0.67	0	0.99	0.56	0.77	0.83	0.59	0.93
3–8	0.63	0.87	0.65	0	0.99	0.67	0.86	0.92	0.61	0.93
4–2	0.60	0	0.63	0	0.99	0	0	0.69	0.52	0.94
4–4	0.31	0	0.73	0	0.99	0.76	0.77	0.83	0.64	0.94
4–6	0	0	0.96	0.13	0.99	0.84	0.67	0.83	0.69	0.97
4–8	0.72	0.88	0.85	0	0.99	0.80	0.80	0.92	0.68	0.94
5–2	0.63	0	0.56	0	0.99	0	0.55	0.88	0.59	0.93
5–4	0.69	0	0.58	0	0.99	0.71	0.77	0.87	0.59	0.96
5–6	0	0	0.61	0	0.99	0	0.67	0.80	0.62	0.93
5–8	0.49	0	0.96	0	0.99	0.80	0.48	0	0.61	0.96
6–2	0.28	0	0.88	0	0	0	0	0.71	0.58	0.91
6–4	0.48	0	0.63	0	0.94	0	0	0.81	0.43	0.97
6–6	0	0	0.58	0	0.93	0	0	0.74	0.43	0.93
6–8	0.49	0	0.56	0	0.99	0	0	0.83	0.55	0.93
L-A	nmod	nsubj	nummod	nummod:gov	obj	obl	punct	root	xcomp	
2–2	0.90	0.71	0.76	0.33	0.58	0.89	0.99	0.98	0.53	
2–4	0.90	0.67	0.75	0.43	0.56	0.91	0.99	0.98	0.53	
2–6	0.88	0.67	0.76	0	0.48	0.90	0.99	0.98	0	
2–8	0.90	0.69	0.75	0	0.54	0.91	0.99	0.98	0	
3–2	0.88	0.67	0.65	0.31	0.55	0.93	0.99	0.98	0	
3–4	0.89	0.69	0.71	0.43	0.59	0.92	0.99	0.99	0.56	
3–6	0.91	0.67	0.73	0.50	0.52	0.92	0.99	0.98	0	
3–8	0.91	0.70	0.71	0.40	0.60	0.93	0.99	0.99	0	
4–2	0.83	0.70	0.70	0.43	0.57	0.90	0.99	0.94	0.45	
4–4	0.86	0.65	0.71	0.43	0.52	0.91	0.99	0	0	
4–6	0.91	0.72	0.75	0.43	0.59	0.92	0.99	0.98	0	
4–8	0.92	0.71	0.77	0.40	0.63	0.93	0.99	0.98	0.61	
5–2	0.87	0.63	0.78	0.53	0.44	0.90	0.99	0	0	
5–4	0.83	0.71	0.72	0.31	0.56	0.90	0.99	0.97	0.52	
5–6	0.87	0.69	0.72	0.31	0.60	0.89	0.99	0	0.52	
5–8	0.89	0.68	0.79	0.43	0.50	0.91	0.99	0.98	0	
6–2	0.78	0.67	0.68	0	0.41	0.88	0.98	0.96	0	
6–4	0	0.64	0.62	0	0.46	0.75	0.99	0.95	0	
6–6	0	0.53	0.54	0	0.40	0.75	0.98	0	0	
6–8	0.83	0.53	0.63	0	0.40	0.88	0.99	0	0	

Table 7: GAT predictions of syntactic dependence in Russian.

				De					
L-A	acl	acl:relel	advcl	advmod	amod	appos	aux	aux:pass	case
2–2	0	0.71	0.83	0.99	0.95	0.39	0.85	0.81	0.99
2–4	0.5	0.75	0.89	0.99	0.95	0.56	0.91	0.81	0.99
2–6	0.5	0.75	0.89	0.99	0.95	0.56	0.91	0.81	0.99
2–8	0	0.41	0	0.99	0.94	0	0.86	0.81	0.99
3–2	0	0.60	0	0.99	0.94	0	0.85	0.81	0.99
3–4	0	0.45	0	0.99	0.94	0	0.85	0.81	0.99
3–6	0	0.41	0	0.98	0.94	0	0.88	0.81	0.99
3–8	0	0.46	0	0.99	0.94	0	0.88	0.81	0.99
4–2	0	0.52	0	0.99	0.95	0	0.81	0	0.99
4–4	0	0.45	0	0.99	0.94	0	0	0	0.99
4–6	0	0.40	0	0.98	0.93	0	0	0.48	0.99
4–8	0	0.45	0	0.98	0.93	0	0	0.52	0.99
5–2	0	0.42	0	0.99	0.92	0	0.86	0.81	0.99
5–4	0	0.68	0	0.99	0.93	0	0.85	0.81	0.99
5–6	0	0.44	0	0.99	0.94	0	0	0	0.99
5–8	0	0.43	0	0.97	0.94	0	0	0	0.99
6–2	0	0	0	0.98	0.9	0.07	0.62	0	0.98
6–4	0	0	0	0.97	0.91	0	0	0.7	0.98
6–6	0	0	0	0.97	0.91	0	0	0	0.98
6–8	0	0.37	0	0.97	0.91	0	0	0	0.98
L-A	cc	ccomp	compound	compound:prt	conj	cop	det	flat:name	mark
2–2	0.99	0.56	0.80	0	0.78	0.93	0.99	0.83	0.97
2–4	0.99	0.60	0.81	0	0.81	0.98	0.99	0.85	0.97
2–6	0.99	0.60	0.81	0	0.81	0.98	0.99	0.85	0.97
2–8	0.99	0	0.72	0	0.80	0.95	0.99	0.81	0.96
3–2	0.99	0.48	0.83	0	0.78	0.93	0.99	0.82	0.95
3–4	0.99	0	0.80	0	0.80	0.95	0.99	0.84	0.86
3–6	0.99	0	0.78	0	0.80	0.95	0.99	0.81	0.91
3–8	0.99	0	0.72	0	0.80	0.95	0.99	0.84	0.91
4–2	0.99	0	0.86	0	0.76	0.93	0.99	0.90	0.93
4–4	0.99	0	0.82	0	0.79	0.57	0.99	0.82	0.84
4–6	0.99	0	0.76	0	0.79	0.90	0.99	0.85	0.93
4–8	0.99	0	0.80	0	0.80	0.88	0.99	0.84	0.85
5–2	0.99	0	0.82	0	0.82	0.95	0.99	0.83	0.92
5–4	0.99	0.52	0.74	0	0.82	0.95	0.99	0.8	0.94
5–6	0.99	0	0.75	0	0.82	0.65	0.99	0.78	0.85
5–8	0.99	0	0	0	0.79	0.57	0.99	0.78	0.86
6–2	0.98	0	0.65	0.67	0.74	0	0.96	0.84	0.82
6–4	0.99	0	0.69	0	0.78	0.70	0.97	0.83	0.84
6–6	0.99	0	0.63	0.69	0.68	0.54	0.98	0.71	0.81
6–8	0.93	0	0.71	0	0	0.55	0.99	0.73	0.87
L-A	nmod	nmod:poss	nsubj	nummod	obj	obl	obl:tmod	punct	root
2–2	0.82	0.85	0.75	0.84	0.63	0.80	0	0.99	0.96
2–4	0.83	0.88	0.72	0.84	0.63	0.83	0	0.99	0.97
2–6	0.83	0.88	0.72	0.84	0.63	0.83	0	0.99	0.97
2–8	0.76	0.86	0.69	0.84	0.56	0.80	0	0.99	0.94
3–2	0.80	0.85	0.78	0.87	0.67	0.84	0	0.99	0.97
3–4	0.80	0.86	0.71	0.84	0.37	0.84	0	0.99	0.92
3–6	0.79	0.85	0.72	0.87	0.56	0.86	0	0.99	0.93
3–8	0.81	0.83	0.74	0.87	0.59	0.84	0	0.99	0.93
4–2	0.81	0.86	0.74	0.84	0.65	0.85	0	0.99	0.95
4–4	0.78	0.85	0.73	0.87	0.51	0.86	0	0.99	0.93
4–6	0.81	0.82	0.77	0.84	0.65	0.85	0	0.99	0.93
4–8	0.78	0.86	0.74	0.87	0.64	0.86	0	0.99	0.95
5–2	0.81	0.83	0.78	0.90	0.62	0.83	0.44	0.99	0.89
5–4	0.82	0.84	0.79	0.90	0.66	0.87	0.44	0.99	0.96
5–6	0.82	0.85	0.72	0.87	0.56	0.82	0	0.99	0.96
5–8	0.76	0.83	0.73	0.80	0.60	0.85	0	0.97	0.89
6–2	0.73	0.81	0.65	0.67	0.23	0.72	0	0.97	0.89
6–4	0.75	0.85	0.65	0.76	0.23	0.87	0	0.97	0.79
6–6	0.81	0.85	0.67	0.81	0.22	0.85	0	0.98	0.90
6–8	0.66	0	0.63	0.81	0	0.86	0	0.98	0.89

Table 8: GAT predictions of syntactic dependence in German.

De	
L-A	xcomp
2–2	0.55
2–4	0.49
2–6	0.49
2–8	0
3–2	0.38
3–4	0
3–6	0
3–8	0
4–2	0.41
4–4	0
4–6	0
4–8	0
5–2	0
5–4	0
5–6	0
5–8	0
6–2	0
6–4	0
6–6	0
6–8	0

Table 9: GAT predictions of syntactic dependence in German.