Analysis of Dependency Pruning on the effectiveness of Transformer models

Kaveh Eskandari
Tehran Institute for Advanced Studies
kaveeskandari96@gmail.com

Mahdi Zakizadeh
Tehran Institute for Advanced Studies
mahdizakizadeh.me@gmail.com

Abstract

We propose a new method for decreasing the size of input sentences in Word-In-Context[1] tasks using what we call Dependency Tree Pruning. We argue that the tokens with the highest effectiveness on correct label prediction lie closer to the target word in dependency tree of input sentences and thus, by pruning the tree in a way such that only the words with some distance K from the target word are present in the final form of the input sentences, we can retain majority of the relevant data needed to make correct predictions.

In order to test our findings, we first fine-tune various models with different pruning depths using XLM-Roberta transformer model and compare their performance to another model fine-tuned on full sentences. Then, we analyze our finding by first comparing the sentence size of pruned sentences versus the original sentence and directly comparing the performance of models trained on pruned sentences against a model trained on full sentences. Then we solidify our findings by doing an attribution analysis to show that the most relevant words indeed lie close to the target word.

We finally argue that our proposed method is a reliable technique for reducing the Word-In-Context input size without damaging the information contained in the sentences to make meaningful inferences. And by putting more emphasis on more important tokens, in some cases it can even boost the performance.

1 Introduction

Advances in word display in NLP systems has been one of the areas that has improved the performance of many NLP systems in various tasks. Contextualized models such as BERT are among the most successful word representation methods. In this method, the word representation is made based not only on the word itself but also on its context. These models can recognize different senses of a word. One of the challenges of working with contextualized models is determining how well these models can understand the different senses of the word. For measuring that, various metrics and tasks have been proposed, including word sense disambiguation, in which the model must match the sense of a word in its specific context. More recently, Word-in-Context (WiC) and its multilingual extension, XLWiC[2], have been proposed, in which the goal is to identify whether the target word sense is the same or different in two different sentences. The research aims to show that the words needed to solve the WiC problem are usually close to the target word.To achieve this goal, we propose a method called Dependency Tree pruning which given the target word of the WIC task, only retains the words that are close to the target word with some factor K. Then we compare the models fine-tuned using the data achieved using this method against another model trained on full sentences. We find that the performance degradation is either minimal or there is a slight performance improvement in some cases.

2 Implementation

For implementation purposes, we have used Stanza[3] in order to generate the Dependency Tree of a given sentence. Figure 1 showcases an example sentence that is ran through the Stanza Dependency Tree generator. We argue that for a

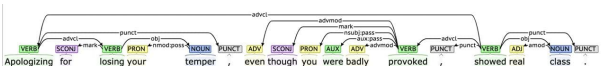

Figure 1: Dependency Tree Stanza example

Model	English	French	German	Italian	Farsi
Normal Fine-Tuning	0.6394	0.565	**0.582**	0.5858	0.66
K=2 Fine-Tuning	0.6471	0.555	0.557	0.591	0.661
K=3 Fine-Tuning	**0.661**	**0.568**	0.561	**0.608**	**0.663**

Table 1: XL-WIC Results

given sentence in Word-In-Context task, the tokens that are most important for the model to make decisions on, are actually the ones that are close to the target word. We measure this closeness by counting the number of edges between the target token and another token in the sentence. For instance, if we consider the token 'class' as our target word, then the token 'showed' is close to it by the degree of 1, and the token 'provoked' is close to it by the degree of 2 and so on. Based on this logic, we further hypothesize that by only keeping a small portion of each sentence(up to a distance K), we won't see any significant degradation in model performance and by making the model focus on more important tokens, we might actually observe improvements in model performance.

In order to put this hypothesis to test, we put every sentence-tuple in a dataset $D = [i^0, i^1]$ through a process that we call reduction. In order to perform reduction, first we run every tuple through the Stanza Dependency Tree generator in order to have a tree representation for every example in the dataset. Then, starting from the target token which is now a node in the tree, we perform Depth-First-Search up to a depth K which is the distance that we are covering starting from the target token. By doing this process iteratively over the entire dataset, we achieve a new dataset that is identical to the original one; however, there is a difference such that the examples in the dataset only contain tokens that are close to the target token by a degree of K.

Having constructed a new dataset based on the aforementioned hypothesis, we now fine-tune a Transformer model using the new dataset and compare its result to another Transformer model fine-tuned on full sentences.

3 Experiments

We test our proposed model on two datasets. First, we cover XL-WIC dataset in a cross-lingual setting, which provides 5000 training examples in English. We then compare the results achieved using our approach in English development set and French, Italian, Farsi and German test sets against a model fine-tuned on full sentences from the English training set. Second, we run the same process on MCL-WIC[4] to solidify our results achieved from the XL-WIC experiment. MCL-WIC objective task is the same as XL-WIC, with the difference being in the data. As the examples in MCL-WIC are human generated in comparison to being crawled from online sources. MCL-WIC contains 8000 English training examples. We again test the performance of our approach in a cross-lingual setting on English, French, Arabic, Russian and Chinese test sets. All experiments are done with fine-tuning on XLM-Roberta[5] for 15 epochs with the batch size of 64 and learning rate of $1e-5$.

We test our approach with three different settings of $Distance = K = 1, 2, 3$. Meaning that for each experiment, 3 new datasets are created each containing tokens that are separated from the target token by the degree of K.

For $K = 1$, we observed significant performance degradation, with our model not being able to learn any meaningful inference rules beyond randomly choosing a label. Thus we decided to forego further tests using $K = 1$ and have conducted the tests with $K = 2$ and $K = 3$. All experiments are done with fine-tuning for 15 epochs with the batch size of 64 and learning rate of $1e-5$. Table 1 showcases the results achieved in XL-WIC dataset averaged across 5 fine-tuning runs for every model.

We observe that pruning the dependency tree with $K = 3$ achieves minor improvement over the normal Fine-Tuning method on some languages, and $K = 2$ has no meaningful effect on the performance of the model. However, degradation in German might suggest that the maintenance of information against removal using our approach might actually be language dependent.

Table 2 showcases the results achieved using MCL-WIC dataset using the same Hyperparameters and averaging method. In the case of MCL-WIC, we observe that all languages except for Arabic do better with Normal Fine-Tuning compared to our model. However, with 3 new languages present in this dataset, it can be the case that our approach is indeed language dependent.

Model	English	French	Arabic	Russian	Chinese
Normal Fine-Tuning	**0.776**	**0.70**	0.748	**0.686**	**0.729**
K=2 Fine-Tuning	0.755	0.6456	0.69	0.6732	0.6623
K=3 Fine-Tuning	0.7744	0.6704	**0.7636**	0.6825	0.6757

Table 2: MCL-WIC Results

Furthermore, even with significant portion of input sentences gone, we can see that XLM-Roberta can still make meaningful inferences, which suggests that the words that are most important for decision making are indeed close to the target token in the dependency tree. In the next part, we conduct analysis to show this pattern.

4 Analysis

We first take the number of tokens pruned using our approach with $K = 2$ and $K = 3$ and compare it to the number of original tokens in the main dataset for both MCL-WIC and XL-WIC.

For XL-WIC, we showcase the ratio of tokens present in the pruned dataset against the original dataset using English Training and Dev sets and French, Italian, German and Farsi test sets. Table 3 shows the results.

We see that the sentences in the XL-WIC dataset

Tree	English-Train	English-Dev	French	German	Italian	Farsi
K=2	0.628	0.625	0.38	0.436	0.478	0.439
K=3	0.666	0.645	0.69	0.465	0.529	0.514

Table 3: Percentage of Words Present in Pruned XL-WIC against the Original Dataset

are rather long, with $K = 3$ covering at most 66.6% of tokens in the original dataset. Next, we compare the performance of models trained using our approach against the model trained using the entire dataset by dividing the performance of our approaches by the performance of the full model. Table 4 showcases the results. We observe that

Tree	English	French	German	Italian	Farsi
K=2/Full Model	1.01	0.982	0.957	1.008	1
K=3/Full Model	1.04	1.005	0.963	1.037	1

Table 4: Performance of K=2 and K=3 models against the Full Model in XL-WIC

even by drastically decreasing the number of tokens in target sentences, our approach still either outperforms the Full Model by a small margin or scores close to it.

Table 5 and 6 showcase the same analysis on MCL-WIC dataset with English Train and Test sets and French, Arabic and Russian test sets. We

Tree	English-Train	English-Test	French	Arabic	Russian
K=2	0.251	0.257	0.2808	0.2419	0.2737
K=3	0.338	0.3414	0.3728	0.3522	0.3417

Table 5: Percentages of words present in Pruned Models against the Full Model in MCL-WIC

observe that the change is even more drastic in MCL-WIC, with texts being larger, the highest amount of tokens retained are in french with $K = 3$, with only $37.\%$ of tokens remaining. By looking at the performance ratio, we observe that even by removing most of the original sentences using our method, our model performs well against a model trained on full data, with 92.2% being the largest performance degradation in MCL-WIC dataset using the cross-lingual setting.

Tree	English	French	Arabic	Russian
K=2\Full Model	0.972	0.922	0.922	0.981
K=3\Full Model	0.997	0.957	1.02	0.994

Table 6: Performance of K=2 and K=3 models against the Full Model in MCL-WIC

We believe that this observation conforms to our initial argument that the words that are closer to the target word in the Dependency Tree carry a crucial role in final inference of XL-Roberta model.

Next, we do a more direct analysis on the effect of words on the final inference of the model. We achieve this by making use of the Erasure[6] method. In order to do this, for each tuple i^0, i^1 in the dataset, we erase the tokens one by one. For instance, $Heisagreatleader$ first becomes $isagreatleader$ and then $heagreatleader$ and so on. And then, for each new sentence created, we predict the outcome using an XLM-Roberta model trained on Full Sentences. For each output, we define a variable $effect = NewOutput - OriginalOutput$ on the probability of the ground truth label of that example using the Original Output against the probability of the ground truth label of the example with a single token erased. A positive $effect$ showcases that the erased token has a positive contribution towards the prediction of the correct label, while a negative effect shows a negative contribution. After calculating $effect$ for each tuple, we then take the top 3 tokens with the highest positive effect and find their distance in the Dependency Tree from the target word. We finally average across all instances to find the dis-

tance with the highest effect concentration. Table 7 showcases the results for English, French, German, Italian and Farsi in XL-WIC dataset. We

English	French	German	Italian	Farsi
2.45	3.33	3.81	2.72	2.37

Table 7: Highest Effect Concetration of XL-WIC languages.

observe that using the Erasure method, the average distance of tokens with the highest effect on predicting the correct label are indeed close to the target word. This further solidifies our argument that dependency pruning can be used to decrease the size of input sentences and put more focus on tokens with high contribution without losing any major performance and even in some cases, gaining some improvements.

5 Conclusion

We showed that Dependency Tree pruning is a reliable method to decrease the size of input sentences in WIC tasks without losing significant relevant information that might damage the ability of the models in inferring the correct outputs. This can be useful when input sentences are very long(longer than Model maximum), in order to decrease their size without losing information.
Furthermore, the method proposed might actually improve the performance of some languages as it forces the model to pay extra attention to the words that are more relevant to the problem.
For future works, one can instead focus on removal of word representations instead of words themselves in order to solve the input abnormalization problem while feeding the input sentences to the model. Pruning using attention scores or gradient scores can also be used to be compared with the current model.
Additionally, the effect of language itself on the efficiency of this model can later be analyzed as it seems like the performance might be language dependent as different structures lead to words being in different depths away from the target word. This can lead to automatic discovery of the correct pruning depth.

6 References

[1] Pilehvar, Mohammad Taher, and Jose Camacho-Collados. “WiC: The Word-in-Context Dataset for Evaluating Context-Sensitive Meaning Representations.” ArXiv:1808.09121 [Cs], Apr. 2019. arXiv.org, http://arxiv.org/abs/1808.09121.
[2] Raganato, Alessandro, et al. “XL-WiC: A Multilingual Benchmark for Evaluating Semantic Contextualization.” ArXiv:2010.06478 [Cs], Oct. 2020. arXiv.org, http://arxiv.org/abs/2010.06478.
[3] Qi, Peng, et al. “Stanza: A Python Natural Language Processing Toolkit for Many Human Languages.” ArXiv:2003.07082 [Cs], Apr. 2020. arXiv.org, http://arxiv.org/abs/2003.07082.
[4] Martelli, Kalach, et al. ”SemEval-2021 Task 2: Multilingual and Cross-lingual Word-in-Context Disambiguation (MCL-WiC)”.
[5] Conneau, Alexis, et al. “Unsupervised Cross-Lingual Representation Learning at Scale.” ArXiv:1911.02116 [Cs], Apr. 2020. arXiv.org, http://arxiv.org/abs/1911.02116.
[6] Li, Jiwei, et al. “Understanding Neural Networks through Representation Erasure.” ArXiv:1612.08220 [Cs], Jan. 2017. arXiv.org, http://arxiv.org/abs/1612.08220.