Less is More: Simplifying Feature Extractors Prevents Overfitting for Neural Discourse Parsing Models

Ming Li  
Texas A&M University  
liming@tamu.edu

Sijing Yu  
Texas A&M University  
sijing@tamu.edu

Ruihong Huang  
Texas A&M University  
huangrh@cse.tamu.edu

Abstract

Complex feature extractors are widely employed for text representation building. However, these complex feature extractors can lead to severe overfitting problems especially when the training datasets are small, which is especially the case for several discourse parsing tasks. Thus, we propose to remove additional feature extractors and only utilize self-attention mechanism to exploit pretrained neural language models in order to mitigate the overfitting problem. Experiments on three common discourse parsing tasks (News Discourse Profiling, Rhetorical Structure Theory based Discourse Parsing and Penn Discourse Treebank based Discourse Parsing) show that powered by recent pretrained language models, our simplified feature extractors obtain better generalizabilities and meanwhile achieve comparable or even better system performance. The simplified feature extractors have fewer learnable parameters and less processing time. Codes will be released and this simple yet effective model can serve as a better baseline for future research.

1 Introduction

Discourse Parsing derives a form of discourse structure that may consist of roles of individual sentences, relations between sentences or relations between a sentence and a larger text unit, and the derived discourse are widely useful for many NLP applications (Yu et al., 2020; Meyer and Popescu-Belis, 2012; Ji et al., 2016). We focus on three discourse parsing tasks in this paper, news discourse profiling (Choubey et al., 2020), RST (Rhetorical Structure Theory) style and PDTB (Penn Discourse Treebank) style discourse parsing tasks (Mann and Thompson, 1988; Prasad et al., 2008).

Like many other recent NLP systems, neural discourse parsing models often employ complex feature extractors on top of neural language models for text representation building. For instance, for news discourse profiling, (Choubey et al., 2020; Choubey and Huang, 2021) use 2 Long Short-Term Memory networks (LSTM) (Hochreiter and Schmidhuber, 1997) with self-attention modules to obtain context-aware sentence embeddings. For RST style discourse parsing, (Kobayashi et al., 2020; Koto et al., 2021; Yu et al., 2022) use several LSTM or Transformer (Chorowski et al., 2015) layers for building sentence representations. For PDTB style discourse parsing, (Dai and Huang, 2018; Bai and Zhao, 2018; Liu et al., 2021; Munir et al., 2021) also use LSTM layers, and some of them use additional convolution layers or extra memory bank to create sentence embeddings.

However, we find these models overfitting the training data quickly, presumably because their complex feature extractors need large datasets to train while discourse parsing tasks often lack an-
notated data. This quick overfitting is undesirable, resulting in a poor generalization ability of these models.

Since neural language models (Peters et al., 2018; Devlin et al., 2019; Raffel et al., 2020) have been trained on numerous corpora, they generalize well and may be already very capable in generating useful sentence representations. Thus, we propose to remove additional feature extractors and only utilize the self-attention mechanism to further exploit pretrained neural language models to produce properly adjusted sentence embeddings for discourse-level tasks. The weights of pretrained language models are fixed to preserve their generalization abilities, and only parameters of self-attention layers are to be learned. Take the News Discourse Profiling task as an example, the two recent models (Choubey et al., 2020; Choubey and Huang, 2021) are susceptible to severe overfitting problems as shown in Figure 1. For these two models, the gaps between their training and validation loss curves grow dramatically within 10 epochs, which indicates these two models overfit with training data easily (near zero training loss) and fail to generalize to new data (high validation loss). In contrast, the validation loss of our simplified model remains stable all the time and the gap between its training and validation loss is kept small, indicating our simplified model achieves better generalizability.

We name our simple self-attention based model as LiMNet, where LiM stands for Less is More. In this model, all additional feature extractors are removed from the encoder and two self-attention modules (Bahdanau et al., 2014; Chorowski et al., 2015) are utilized to get the local sentence embedding $1$ and the global sentence shift. These two embeddings of different reception fields are fused and sent to the task-specific predictors. Extensive experiments and analysis on three discourse parsing tasks including News Discourse Profiling, RST style and PDTB style Discourse Parsing show that LiMNet largely prevents the overfitting problem. Meanwhile, powered by recent pretrained language models, our simplified model achieves comparable or even better system performances with fewer learnable parameters and less processing time on the tasks.

The contributions of this paper are threefold:

- To our best knowledge, we are the first proposing to reduce overfitting by removing additional feature extractors and making greater use of pretrained language models
- Our model alleviates the severe overfitting problem and improves generalization abilities on all the three discourse parsing tasks mentioned above
- Our model is simple yet effective, and it achieves promising performances on these tasks with less processing time and fewer learnable parameters

2 Discourse Parsing Tasks and Models

In this section, we introduce the tasks and models in neural discourse parsing that we focus on and compare the differences between baseline models and our model.

2.1 News Discourse Profiling

News Discourse Profiling (Choubey et al., 2020) is a task focusing on the role of each sentence in news articles (Van Dijk, 1986, 2013; Pan and Kosicki, 1993). It aims to understand the whole discourse in a higher-level by identifying the main event with supporting content.

Baseline: We use (Choubey et al., 2020) as our baseline model as shown in Figure 2(a). The first Bi-LSTM layer with self-attention module is used upon word embeddings to calculate sentence embeddings, and the second is used to obtain document embedding. Then the final output vector for each sentence is calculated by the combination of sentence embedding and document embedding.

Updated Baseline: The language model used in the original baseline is ELMo (Peters et al., 2018). For fair comparison, we utilize word embeddings from T5 (Raffel et al., 2020) language model as the input while the baseline model keeps unchanged. Further experiments are conducted in ablation study to explore the effects of different language models.

Our model: As discussed above, introducing extra feature extractors requires more data for training while discourse parsing tasks often lack annotated data, which leads to quick overfitting on training data. Thus we remove the word-level and sentence-level Bi-LSTM layers in the baseline, and only utilize 2 self-attention modules to better exploit the

---

1 We call Elementary Discourse Unit embeddings in RST, Discourse Unit embeddings in PDTB both as sentence embeddings and do not distinguish them strictly.
Figure 2: A brief illustration of feature extractor structures of baseline models and our model. (a) is the baseline model for news discourse profiling task, (b) is the baseline model for RST discourse parsing task, (c) is the baseline model for PDTB discourse parsing task and (d) is our LiMNet. Word represents input word embeddings, POS represents part-of-speech embeddings, Syntax represents syntax features. SA represents self-attention module. All the baseline models (a)(b)(c) utilize word-level and sentence-level LSTM feature extractors for sentence representation building, which may lead to severe overfitting. Our model (d) removes these additional feature extractors to prevent overfitting.

 pretrained language model shown in Figure 2(d). Specifically, the first self-attention module is used to build representative local sentence embeddings similar to the baseline. The second self-attention module is utilized to calculate the global sentence shifts, aiming to model global contextual information for each sentence. This simplification not only alleviates the overfitting problem, but also reduces the number of parameters and processing time simultaneously.

2.2 RST Discourse Parsing

The purpose of Rhetorical Structure Theory based Discourse Parsing (RST)(Mann and Thompson, 1988; Carlson et al., 2001) is to present a discourse by using a hierarchical rhetorical tree, where each leaf represents an elementary discourse unit (EDU).

Baseline: We use (Koto et al., 2021) (LSTM version with static oracle) as our baseline model as shown in Figure 2(b). The first Bi-LSTM followed by an average pooling is utilized upon the concatenation of word and part-of-speech (POS) embeddings in each elementary discourse unit (EDU). The second Bi-LSTM is utilized upon the syntax features from (Dozat and Manning, 2017). The sentence embeddings are obtained by combining

the above 2 embeddings with additional EDU type embeddings. Then a third Bi-LSTM layer is further deployed to get the final output embeddings.

Updated Baseline: The language model used in the baseline is GloVe (Pennington et al., 2014), which is outdated and may not contain enough contextual information to support the removal of feature extractors. Therefore, we update the baseline to support T5 language model. Both POS embedding and syntax features used in baseline are word-specific, while the tokens from T5 are not word-specific, leading to mismatch problems, thus we discard the POS embeddings, syntax features and the Bi-LSTM with it. In updated baseline, the first Bi-LSTM layer with average pooling is used upon input T5 word embeddings for local sentence construction. And another Bi-LSTM layer is utilized to model contextual relation and the EDU type embeddings are added to the output.

Our model: Our model uses the same word embeddings as the updated baseline. However, compared with the updated baseline, our model further replaces all the LSTM layers with self-attention modules as shown in Figure 2(d). For the discourse unit segmentation step, we use the same segmenter as in their experiments.

2.3 PDTB Discourse Parsing

Penn Discourse Treebank based Discourse Parsing (PDTB)(Prasad et al., 2008) focuses more on the relation between 2 local discourse units.

Baseline: Most of PDTB models only consider the relation between adjacent arguments and neglect the global contextual relation of the whole discourse. (Dai and Huang, 2018) is the first context aware method which models the interdependencies between discourse units at paragraph level. The input of this model is the continuous discourse units (DUs) in the same paragraph instead of a single DU pair. Thus we choose (Dai and Huang, 2018) as our PDTB baseline where a word-level Bi-LSTM followed by a max pooling is utilized to construct the sentence representations. Then a second Bi-LSTM is further utilized upon the obtained sentence embeddings to obtain the final vector output, as shown in Figure 2(c).

Updated Baseline: The language model used in the original baseline is google word2vec (Mikolov et al., 2013), which is outdated and may not contain enough contextual information to support the
Figure 3: The structure of our LiMNet. $A^L$ represents self-attention module used for calculating local sentence embeddings and $A^G$ represents self-attention module used for calculating global sentence shifts. The input of $A^G$ contains the $i_{th}$ local sentence embedding and all the word embeddings. $\oplus$ represents addition operation. $W_{i,j}$, $L_i$, $G_i$ and $M_i$ represent word embedding, local sentence embedding, global sentence shift and mixed sentence embedding, respectively. The mixed sentence embeddings will be later sent to the task-specific predictors.

removal of feature extractors. Therefore, we utilize word embeddings from T5 language model as the input while the baseline model keeps unchanged. Specifically, the whole discourse is sent to T5 model for capturing global contextual information and the discourse units that need to be predicted are sent to model in paragraph level. 

**Our model:** Our model uses the same word embeddings as updated baseline. Furthermore, compared with updated baseline, our model discards all the LSTM layers and utilizes self-attention modules instead shown in as shown in Figure 2(d).

3 LiMNet

In this section, we introduce the detailed structure of our LiMNet as shown in Figure 3. Input document will be first sent to the pretrained language model to get all the contextual word embeddings. Upon these word embeddings, 2 self-attention modules are implemented to calculate the local sentence embedding and global sentence shift for each sentence. Then the mixed sentence embedding obtained by adding the 2 embeddings will be sent to the task-specific predictors.

**Word Embedding:** Pretrained language models (Peters et al., 2018; Devlin et al., 2019) have been widely used in recent NLP tasks, which provide general and representative word embeddings for downstream tasks. These language models are mostly trained on large unannotated corpus, which outnumber the data used for downstream tasks by a great margin. In neural discourse parsing tasks, we find the use of additional feature extractors will limit the generalization of pretrained word embeddings and cause severe overfitting. Based on the above discussion, we design LiMNet which removes additional feature extractors and utilizes self-attention mechanisms to obtain the representative sentence embeddings. Considering these are discourse-level tasks, we choose T5 (Raffel et al., 2020) as our language model since it supports long input while most other language models only support inputs within 512 tokens. To preserve the generalization ability of pretrained language model, its weights are fixed in all experiments without fine-tuning.

**Local Sentence Embedding:** For discourse-level tasks, getting representative sentence embeddings is fundamental. Most models utilize word-level feature extractor followed by a combination method for it. However, with the purpose of simplifying model, we use only a self-attention module to calculate local sentence embeddings. Specifically, 2 feed forward networks (FFN), with a $\tanh$ function
in between, are utilized upon each word embedding to obtain a scalar representing the importance of this word. Then a softmax function is deployed to get the regularized attention weights and the final local sentence embedding is the weighted sum of all local embeddings.

Suppose there are \( n \) sentences in each discourse and each \( i_{th} \) sentence has \( l_i \) words. \( W_{i,j} \) represents the word embedding of the \( j_{th} \) word of the \( i_{th} \) sentence. \( \alpha^L \) represents self-attention weight for calculating local sentence embeddings. \( E^L \) represents the learnable weights for local sentence embeddings, including 2 learnable FFN and a tanh in between. Then the local sentence embedding for the \( i_{th} \) sentence can be calculated as follows:

\[
\alpha^L_{i,j} = \text{softmax}(E^L \cdot W_{i,j}) \quad (1)
\]

\[
L_i = \sum_{j=1}^{l_i} \alpha^L_{i,j} W_{i,j} \quad (2)
\]

**Global Sentence Shift:** Obtaining only local sentence embeddings without contextual information is insufficient for discourse-level tasks. Most existing models implement additional context-aware feature extractors for modeling contextual information. In addition to the overfitting concern, we believe the contextual information is sentence-specific, and each sentence should learn the complementary information of its own. To this end, we implement another self-attention module on the top of local sentence embedding to learn the global shift of each sentence.

Specifically, after getting the local sentence embedding \( L_k \), differences between this embedding and all other word embeddings, \( W_{i,j} \), in the discourse are calculated by subtraction. These new vectors represent the shifts between this exact sentence embedding and all other word embeddings. Then another self-attention module is implemented upon these sentence shift vectors, and thus the global sentence shift vector \( G_k \) of this specific sentence is obtained. The global sentence shift \( G_k \) is calculated as follows:

\[
\alpha^G_{k,i,j} = \text{softmax}(E^G \cdot (W_{i,j} - L_k)) \quad (3)
\]

\[
G_k = \sum_{i=1}^{n} \sum_{j=1}^{l_i} \alpha^G_{k,i,j} W_{i,j} \quad (4)
\]

where \( \alpha^G_{k,i,j} \) represents self-attention weight of word embedding \( W_{i,j} \) for calculating global sentence shift of the \( k_{th} \) sentence. \( E^G \) represents the learnable weights for global sentence shift, including 2 feed forward networks and a tanh function in between.

|          | Macro | Micro |
|----------|-------|-------|
|          | Precision | Recall | F1    | F1    |
| LiMNet   | 68.2   | 63.9  | 65.6  | 69.7  |
| LiMNet no Global | 69.9   | 59.9  | 62.6  | 69.2  |
| LiMNet no Local  | 62.2   | 55.9  | 57.9  | 62.7  |
| LiMNet no Either | 59.5   | 46.5  | 48.7  | 59.3  |

Table 1: The performance of different model configurations. All the models here use the pretrained T5 language model and the results are averaged over five random runs.

The final mixed sentence embeddings are fused by adding global shifts to local sentence embeddings, which is notated as \( M_k = L_k + G_k \). These mixed embeddings will be sent to task-specific predictors.

### 4 Evaluation

**4.1 Dataset**

**News Discourse Profiling:** The NewsDiscourse dataset (Choubeyy et al., 2020) we use is designed for the task of News Discourse Profiling, which consists of 802 news articles (18, 155 sentences). Each sentence in this corpus is labelled with one of eight content types reflecting what role it plays in reporting a news story following the news content schemata proposed by Van Dijk (Van Dijk, 1985, 1988). For fair comparison, we follow the same division of dataset as (Choubeyy et al., 2020; Choubeyy and Huang, 2021), which has 502 documents for training, 100 documents for validation and 200 documents for testing.

**RST Discourse Parsing:** The English RST Discourse Treebank (Carlson et al., 2001) we use is based on Wall Street Journal portion of the Penn Treebank(Marcus et al., 1993). It contains 347 documents for training, and 38 documents for testing. For fair comparison, we use a same validation set as (Koto et al., 2021; Yu et al., 2018) which contains 35 documents from the training set.

**PDTB Discourse Parsing:** The Penn Discourse Treebank v2.0 (Prasad et al., 2008) we use is the largest annotated corpus containing 36k discourse relations in 2,159 Wall Street Journal (WSJ) articles. We use the dataset partition same as (Dai and Huang, 2018; Rutherford and Xue, 2015), where sections 2-20 are training set, sections 21-22 are test set, and sections 0-1 are validation set. In this work we focus on the top-level discourse relations.
Table 2: The performance of using different pretrained language models. Updated Baseline represent (Choubey et al., 2020) model of using different pre-trained language models. All results are calculated by 5 random rounds. The weights of these pretrained language models are fixed without finetuning.

| Language Models         | Updated Baseline | LiMNet |
|-------------------------|------------------|--------|
|                         | Precision | Recall | F1  | Precision | Recall | F1  | Micro | Precision | Recall | F1  | Micro |
| ELMo (Peters et al., 2018) | 56.9      | 53.7   | 54.4 | 60.9      | 58.1   | 56.1 | 56.5 | 62.5      | 64.1   | 64.1 |
| BERT (Devlin et al., 2019) | 59.4      | 58.4   | 58.6 | 63.5      | 62.1   | 56.4 | 58.0 | 64.1   |
| RoBERTa (Liu et al., 2019) | 62.9      | 57.0   | 58.8 | 64.9      | 65.5   | 56.3 | 59.2 | 67.1 |
| T5 (Raffel et al., 2020)   | 67.4      | 60.4   | 62.5 | 68.4      | 68.2   | 63.9 | 65.6 | 69.7 |

4.2 Implementation Details

All experiments are implemented in PyTorch platform (Paszke et al., 2019) and all the training and inference times are calculated by using NVIDIA GeForce RTX 3090 graphic card. We use t5-large (Raffel et al., 2020) from huggingface (Wolf et al., 2019) as our pretrained language model in all the models. The weights of pretrained language model are fixed all the time without finetuning.

For each model of different task, we follow the same configuration of original baselines. News Discourse Profiling models are trained using Adam optimizer (Kingma and Ba, 2014) with the learning rate of $5e - 4$ for 100 epochs. RST models are trained using Adam optimizer (Kingma and Ba, 2014) with the learning rate of $1e - 3$, epsilon of $1e - 6$, gradient accumulation of 2 for 120 epochs. PDTB models are trained using Adam optimizer (Kingma and Ba, 2014) with the learning rate of $5e - 4$ for 100 epochs. The dropout rates (Srivastava et al., 2014) of all the models are set to 0.5.

We run each model for 5 rounds using different random seeds and report the averaged performance to alleviate the influence of randomness.

4.3 Ablation Study for Model Configuration

We conduct several ablation experiments on the News Discourse Profiling task to verify the effectiveness of each module in our LiMNet (Table 1).

LiMNet no Global represents the model in which the calculation of global sentence shift is removed and the local sentence embeddings are sent to the prediction layer directly. While seeing a little increase in precision, removing the global sentence shift causes a severe drop on the recall and thus yields a lower F1 score. LiMNet no Local represents the model in which the calculation of local sentence embedding is replaced by an average pooling. We can see that removing the sentence-level self-attention module greatly lowers both precision and recall. LiMNet no Either represents the model where both local and global embeddings are removed. The only learnable part in this model is the FFN in the final prediction layer whose result shows the intrinsic representation ability of T5 language model. The ablation analysis shows that each self-attention module in LiMNet is useful, and LiMNet exploits the language model and preserves its generalizability.

4.4 Experimental Results Using Different Language Models

In this subsection, we explore the effect of using different language models on the task of News Discourse Profiling. Table 2 presents the performances of using different language models for our baseline model and LiMNet. When using different language models, our simple self-attention based LiMNet still outperforms baseline model with relatively complex structures.

Moreover, Figure 4 shows the training and validation loss curves for both the baseline models and LiMNet models with different language models.
We can see that across different language models, the gaps between training and validation loss grow rapidly over training epochs, which indicates the baseline models are susceptible to severe potential overfitting problems. In contrast, the validation loss of LiMNet are stable and gaps between their training and validation losses are kept much smaller, indicating that LiMNet models well preserve the generalizability of pre-trained language models.

4.5 Results on News Discourse Profiling

As shown in Table 3, our model has the best performance on all of the evaluation metrics (macro P/R/F and micro F1 score). In addition to the Baseline model (Choubey et al., 2020) and its T5 version (Updated Baseline), our model also outperforms the more recent model (Choubey and Huang, 2021) and its T5 version. It is impressive that the simple self-attention based model outperforms all those models with multiple additional feature extractors. Moreover, our model has fewer parameters.

Table 4: RST discourse parsing results, using original Parseval and RST Parseval metrics. S, N, R, F represents Span, Nuclearity, Relation and Full. (Koto et al., 2021) (baseline) represents the LSTM version with static oracle of original model, which is used as our RST baseline. Updated Baseline represents the updated version of its original model to match the use of T5 language model. The results of ours and baseline models are averaged over 5 runs. The highest performances are in bold and the second highest performances are underlined. Para, Train and Infer represents the number of learnable parameters, training and inference time, respectively. The number of learnable parameters does not include parameters of language model.

Table 3: News discourse profiling results, compared with previous methods in terms of performance and efficiency. All the results are averaged over 5 runs, and the standard deviation for both macro and micro F1 scores are provided in brackets. The original models (Choubey et al., 2020; Choubey and Huang, 2021) used ELMo language model, thus we updated these models by using T5 pretrained models for direct comparisons, notated with w/T5. Para, Train and Infer represents the number of learnable parameters, training and inference time, respectively. The number of learnable parameters does not include parameters of language model.

Figure 5: The training loss curves (dotted line) and validation loss curves (solid line) of (a) RST models and (b) PDTB models. The Baseline model and Updated Baseline model suffer from severe overfitting problem while our model largely reduces this problem.
4.6 Results on RST Discourse Parsing

As shown in Table 4, the Updated Baseline model achieves the best performance and outperforms the previous methods across all the metrics. Although with slightly lower scores, our model achieves the second best performance on most of the metrics. However, as shown in Figure 5(a), the Updated Baseline suffers from severe overfitting, where its validation curve begins to increase dramatically from around the 30th epoch before it converges. On the contrary, our model keeps the validation loss low throughout the training process and has fewer learnable parameters and less processing time. So, it is still worthy reducing overfitting, the number of parameters and processing time at the cost of a little performance drop.

4.7 Results on PDTB Discourse Parsing

As shown in Table 5, our model outperforms the Baseline and Updated Baseline models on both the macro F1 score and the accuracy for implicit relation recognition. Compared with the baseline model (Dai and Huang, 2018), the updated baseline greatly improves the performance, but suffers from even more severe overfitting as shown in Figure 5(b), where its validation curve increases more dramatically than the baseline. However, after removing the LSTM layers from the updated baseline, our model is able to largely reduce the overfitting as seen in Figure 5(b). Meanwhile, our LiMNet model achieves even better performance with fewer parameters, less training and inference time.

5 Related Work

Overfitting is a common problem in Machine Learning and Deep Learning, which represents the poor generalization ability of model (Dietterich, 1995; Hawkins, 2004), indicating their weak performance on recognizing different unseen data.

To solve the overfitting problem, different methods are proposed. (Lecun et al., 1998; Chawla et al., 2002; Han et al., 2005) use data augmentation. (Srivastava et al., 2014; Demyanov, 2015; Li and Liu, 2016; Wang and Manning, 2013; Gal et al., 2017) utilize dropout based methods. (Ioffe and Szegedy, 2015; Ioffe, 2017; Luo et al., 2018) utilize batch normalization based methods. (Krogh and Hertz, 1991; Tomar and Rose, 2014; Bejani and Ghatge, 2019, 2020) utilize weight decay based methods.

In NLP community, transfer learning (Shao et al., 2014; Weiss et al., 2016) and pretrained (Erhan et al., 2010; Qiu et al., 2020) are widely used. Pretrained language models (Pennington et al., 2014; Peters et al., 2018; Devlin et al., 2019; Clark et al., 2020; Lan et al., 2020; Liu et al., 2019) have been widely used in recent NLP tasks, which provide general and representative word embeddings for downstream tasks. Then the complex feature extractors designed for different downstream tasks are utilized upon these language models. Though augmenting parameters increases the performance of various tasks, a great number of parameters may cause overfitting (Cawley and Talbot, 2007; Tzafestas et al., 1996). This phenomenon motivates us to reduce the additional feature extractors to address the severe overfitting.

|                  | Implicit | Efficiency |
|------------------|----------|------------|
|                  | Macro    | Acc        |
|LiMNet (ours)     | 53.11    | -          |
|Updated baseline  | 56.48    | 65.74      |
|Baseline          | 48.69    | 58.20      |
|Dai and Huang, 2018 | 53.11    | -          |
|Gao et al., 2020  | 52.89    | 59.66      |
|Ningy et al., 2019 | 53.00    | -          |
|Varia et al., 2019 | 50.20    | 59.13      |
|Wu et al., 2020   | 55.72    | 65.26      |
|He et al., 2020   | 51.24    | 59.94      |
|Kishimoto et al., 2020 | 56.48 | 65.26 |
|Zhang et al., 2021 | 53.11    | -          |
|Dai and Huang, 2018 | 48.69    | 58.20      |

Table 5: PDTB discourse parsing results towards implicit discourse relation classification. Updated Baseline represents the updated version of original baseline model to match the use of T5 language model. The results of models are averaged over 5 runs. Para, Train and Infer represents the number of trainable parameters, training and inference time, respectively. The number of trainable parameters does not include parameters of language model. The training and inference time do not include the language model time, following the same code structure with our baseline (Dai and Huang, 2018).
6 Conclusions

Concerned with the severe overfitting problem of recent neural discourse parsing systems, we propose to remove additional complex feature extractors that need to be trained on large datasets, and add self-attention modules to make good use of the pretrained language models, retaining their generalization abilities. Extensive experiments and analysis on three discourse parsing tasks, including News Discourse Profiling, RST and PDTB Discourse Parsing, show that our simplified model LiMNet largely prevents the overfitting problem. In the meantime, our model achieves comparable or even better system performances with fewer learnable parameters and less processing time on the tasks mentioned above. Broadly speaking, beyond neural discourse parsing, there is a trend to create increasingly complex systems for various NLP tasks and applications. Our findings can raise awareness of the overfitting problem in NLP systems and motivate new researches on improving system generalizability.

Limitations

Although our model has fewer parameters and less training and inference time compared with previous baselines models, it requires large GPU memories. For discourse level processing tasks, the input can contain tens of sentences and thousands of words. In our model, the calculation of global sentence embeddings requires each sentence embedding to interact with every word embeddings in the discourse. This is currently implemented by simple matrix operation in a parallel manner, and the resulting matrix will be in the size of \( \text{number of sentence} \times \text{number of words} \times \text{embedding dimension} \). Thus, a large GPU memory is needed to store and process this matrix. For future work, we will investigate data compression and optimization algorithms to reduce memory uses in implementations.

References

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

Hongxiao Bai and Hai Zhao. 2018. Deep enhanced representation for implicit discourse relation recognition. In \textit{Proceedings of the 27th International Conference on Computational Linguistics}, pages 571–583, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Mohammad Mahdi Bejani and Mehdi Ghatee. 2019. Convolutional neural network with adaptive regularization to classify driving styles on smartphones. \textit{IEEE transactions on intelligent transportation systems}, 21(2):543–552.

Mohammad Mahdi Bejani and Mehdi Ghatee. 2020. Theory of adaptive svd regularization for deep neural networks. \textit{Neural Networks}, 128:33–46.

Chloë Braud, Maximin Coavoux, and Anders Søgaard. 2017. Cross-lingual RST discourse parsing. In \textit{Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers}, pages 292–304, Valencia, Spain. Association for Computational Linguistics.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurovsky. 2001. Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory. In \textit{Proceedings of the Second SIGdial Workshop on Discourse and Dialogue}.

Gavin C Cawley and Nicola LC Talbot. 2007. Preventing over-fitting during model selection via bayesian regularisation of the hyper-parameters. \textit{Journal of Machine Learning Research}, 8(4).

N. Chawla, K. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. Smote: Synthetic minority over-sampling technique. \textit{J. Artif. Intell. Res.}, 16:321–357.

Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. 2015. Attention-based models for speech recognition. In \textit{Advances in Neural Information Processing Systems}, volume 28. Curran Associates, Inc.

Prafulla Kumar Choubey and Ruihong Huang. 2021. Profiling news discourse structure using explicit subtopic structures guided critics. In \textit{Findings of the Association for Computational Linguistics: EMNLP 2021}, pages 1594–1605, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Prafulla Kumar Choubey, Aaron Lee, Ruihong Huang, and Lu Wang. 2020. Discourse as a function of event: Profiling discourse structure in news articles around the main event. In \textit{Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics}, pages 5374–5386, Online. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. \textit{ELECTRA: Pre-training text encoders as discriminators rather than generators}. In \textit{ICLR}.
Zeyu Dai and Ruihong Huang. 2018. Improving implicit discourse relation classification by modeling inter-dependencies of discourse units in a paragraph. 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 141–151, New Orleans, Louisiana. Association for Computational Linguistics.

Zeyu Dai and Ruihong Huang. 2019. A regularization approach for incorporating event knowledge and coreference relations into neural discourse parsing. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 2976–2987, Hong Kong, China. Association for Computational Linguistics.

Sergey Demyanov. 2015. Regularization methods for neural networks and related models. Ph.D. thesis, University of Melbourne, Department of Computing and Information Systems.

Tom Dietterich. 1995. Overfitting and undercomputing in machine learning. Computing Surveys, 27:326–327.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings.

Dumitru Erhan, Aaron Courville, Yoshua Bengio, and Pascal Vincent. 2010. Why does unsupervised pretraining help deep learning? In Proceedings of the thirteenth international conference on artificial intelligence and statistics, pages 201–208. JMLR Workshop and Conference Proceedings.

Yarin Gal, Jiri Hron, and Alex Kendall. 2017. Concrete dropout. Advances in neural information processing systems, 30.

Fengyu Guo, Ruifang He, Jianwu Dang, and Jian Wang. 2020. Working memory-driven neural networks with a novel knowledge enhancement paradigm for implicit discourse relation recognition. Proceedings of the AAAI Conference on Artificial Intelligence, 34:7822–7829.

Hui Han, Wenyuan Wang, and Binhuan Mao. 2005. Borderline-smote: A new over-sampling method in imbalanced data sets learning. In ICIC.

Douglas M Hawkins. 2004. The problem of overfitting. Journal of chemical information and computer sciences, 44(1):1–12.

Katsuhiko Hayashi, Tsutomu Hirao, and Masaaki Nagata. 2016. Empirical comparison of dependency conversions for RST discourse trees. In Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 128–136, Los Angeles. Association for Computational Linguistics.

Ruifang He, Jian Wang, Fengyu Guo, and Yugu Han. 2020. TransS-driven joint learning architecture for implicit discourse relation recognition. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 139–148, Online. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. Neural computation, 9(8):1735–1780.

Sergey Ioffe. 2017. Batch normalization: Towards reducing minibatch dependence in batch-normalized models. Advances in neural information processing systems, 30.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning, pages 448–456. PMLR.

Yangfeng Ji, Gholamreza Haffari, and Jacob Eisenstein. 2016. A latent variable recurrent neural network for discourse-driven language models. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 332–342, San Diego, California. Association for Computational Linguistics.

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

Yudai Kishimoto, Yugo Murawaki, and Sadao Kurohashi. 2020. Adapting BERT to implicit discourse relation classification with a focus on discourse connectives. In Proceedings of the 12th Language Resources and Evaluation Conference, pages 1152–1158, Marseille, France. European Language Resources Association.

Naoki Kobayashi, Tsutomu Hirao, Hidetaka Kamigaito, Manabu Okumura, and Masaaki Nagata. 2020. Top-down rst parsing utilizing granularity levels in documents. Proceedings of the AAAI Conference on Artificial Intelligence, 34(05):8099–8106.

Fajri Koto, Jey Han Lau, and Timothy Baldwin. 2021. Top-down discourse parsing via sequence labelling. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 715–726, Online. Association for Computational Linguistics.
Anders Krogh and John Hertz. 1991. A simple weight decay can improve generalization. *Advances in neural information processing systems*, 4.

Zhengzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. *In International Conference on Learning Representations*.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Qi Li, Tianshi Li, and Baobao Chang. 2016. Discourse parsing with attention-based hierarchical neural networks. *In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 362–371, Austin, Texas. Association for Computational Linguistics.

Yinan Li and Fang Liu. 2016. Whiteout: Gaussian adaptive noise regularization in deep neural networks. *arXiv preprint arXiv:1612.01490*.

Xin Liu, Jiefu Ou, Yangqiu Song, and Xin Jiang. 2021. On the importance of word and sentence representation learning in implicit discourse relation classification. *In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI’20.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Ping Luo, Xinjiang Wang, Wenqi Shao, and Zhanglin Peng. 2018. Towards understanding regularization in batch normalization. *arXiv preprint arXiv:1809.00846*.

Amandla Mabona, Laura Rimell, Stephen Clark, and Andreas Vlachos. 2019. Neural generative rhetorical structure parsing. *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2284–2295, Hong Kong, China. Association for Computational Linguistics.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Thomas Meyer and Andrei Popescu-Belis. 2012. Using sense-labeled discourse connectives for statistical machine translation. *In Proceedings of the Joint Workshop on Exploiting Synergies between Information Retrieval and Machine Translation (ESIRMT) and Hybrid Approaches to Machine Translation (HyTra)*, pages 129–138, Avignon, France. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. *In Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS’13, page 3111–3119, Red Hook, NY, USA. Curran Associates Inc.

Kashif Munir, Hongxiao Bai, Hai Zhao, and Junhan Zhao. 2021. Memorizing all for implicit discourse relation recognition. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 21(3).

Linh The Nguyen, Linh Van Ngo, Khoat Than, and Thien Huu Nguyen. 2019. Employing the correspondence of relations and connectives to identify implicit discourse relations via label embeddings. *In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4201–4207, Florence, Italy. Association for Computational Linguistics.

Zhongdang Pan and Gerald M Kosicki. 1993. Framing analysis: An approach to news discourse. *Political Communication*, 10(1):55–75.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. *In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *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 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltysakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The penn discourse treebank 2.0. *In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC’08)*.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pretrained models for natural language processing: A
survey. *Science China Technological Sciences*, 63(10):1872–1897.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Attapol Rutherford and Nianwen Xue. 2015. Improving the inference of implicit discourse relations via classifying explicit discourse connectives. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 799–808, Denver, Colorado. Association for Computational Linguistics.

Ling Shao, Fan Zhu, and Xuelong Li. 2014. Transfer learning for visual categorization: A survey. *IEEE transactions on neural networks and learning systems*, 26(5):1019–1034.

Wei Shi and Vera Dembger. 2019. Learning to explicitate connectives with Seq2Seq network for implicit discourse relation classification. In *Proceedings of the 13th International Conference on Computational Semantics - Long Papers*, pages 188–199, Gothenburg, Sweden. Association for Computational Linguistics.

Nitin Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Vikrant Singh Tomar and Richard C Rose. 2014. Manifold regularized deep neural networks. In *Fifteenth Annual Conference of the International Speech Communication Association*.

SG Tzafestas, PJ Dalianis, and G Anthopoulos. 1996. On the overtraining phenomenon of backpropagation neural networks. *Mathematics and computers in simulation*, 40(5-6):507–521.

Teun A Van Dijk. 1985. Structures of news in the press. *Discourse and communication: New approaches to the analysis of mass media discourse and communication*, 10:69.

Teun A Van Dijk. 1986. 5news schemata.

Teun A Van Dijk. 1988. News analysis. *Case Studies of International and National News in the Press. New Jersey: Lawrence*.

Teun A Van Dijk. 2013. *News as discourse*. Routledge.

Siddharth Varia, Christopher Hidey, and Tuhin Chakrabarty. 2019. Discourse relation prediction: Revisiting word pairs with convolutional networks. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 442–452, Stockholm, Sweden. Association for Computational Linguistics.

Sida Wang and Christopher Manning. 2013. Fast dropout training. In *International conference on machine learning*, pages 118–126. PMLR.

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. 2016. A survey of transfer learning. *Journal of Big data*, 3(1):1–40.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond,克莱芒·德朗格, Anthony Moi, Pierrick Cistac, Tim Rault, Rémi Louf, Morgan Funkowicz, et al. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Changxing Wu, Chaowen Hu, Ruochen Li, Hongyu Lin, and Jinsong Su. 2020. Hierarchical multitask learning with crf for implicit discourse relation recognition. *Knowledge-Based Systems*, 195:105637.

Lei Yu, Laurent Sartran, Wojciech Stokowiec, Wang Ling, Lingpeng Kong, Phil Blunsom, and Chris Dyer. 2020. Better document-level machine translation with Bayes’ rule. *Transactions of the Association for Computational Linguistics*, 8:346–360.

Nan Yu, Meishan Zhang, and Guohong Fu. 2018. Transition-based neural RST parsing with implicit syntax features. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 559–570, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Nan Yu, Meishan Zhang, Guohong Fu, and Min Zhang. 2022. RST discourse parsing with second-stage EDU-level pre-training. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4269–4280, Dublin, Ireland. Association for Computational Linguistics.

Longyin Zhang, Yuqing Xing, Fang Kong, Peifeng Li, and Guodong Zhou. 2020. A top-down neural architecture towards text-level parsing of discourse rhetorical structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6386–6395, Online. Association for Computational Linguistics.

Yingxue Zhang, Fandong Meng, Peng Li, Ping Jian, and Jie Zhou. 2021. Context tracking network: Graph-based context modeling for implicit discourse relation recognition. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1592–1599, Online. Association for Computational Linguistics.