# Discourse Parsing via Weighted Bag-of-Words, Coattention Neural Encoder, and Coattentative Convolutional Neural Network

**Borui Wang**
Department of Computer Science
Stanford University
Stanford, CA 94305
wbr@stanford.edu

**Yang Yuan**
Department of Computer Science
Stanford University
Stanford, CA 94305
yyuan16@stanford.edu

**Alex Fu**
Department of Computer Science
Stanford University
Stanford, CA 94305
xiaofu@stanford.edu

## Abstract

Discourse parsing is an important task in NLP for understanding logical relations between adjacent sentences. In this report, we experiment with three models to improve classification task on discourse parsing: 1) weighted bag of words model using scoring DNN; 2) coattention neural encoder model using affinity matrix; 3) coattentative convolutional neural network model. We then show in our experiments that these three models all improve significantly over baseline model and give competitive accuracy on the task of discourse parsing over Penn Discourse Treebank .

## 1 Introduction

In any language context, sentences are the basic components to form meaningful discourse. Sentences must be pieced together by logical discourse relations to form coherent text. Although many obvious relations are signaled by discourse connectives, human readers can distinguish and understand many more discourse relations between sentences without any connectives. Therefore, it becomes important for any natural language understanding system to correctly recognize discourse relations.

Currently, discourse parsing is often a bottleneck for many downstream NLP tasks such as text summary and machine translation. The current state-of-art methods for discourse parsing still cannot yield satisfactory performance results on the standard datasets, and are performing worse on classifying implicit relations, that is, discourse relations without connectives.

In this project, we aim to investigate how to improve discourse parsing by obtaining better sentence-level vector embeddings. We build on previous approaches in literature and, implement and experiment with different neural-network models introduced in class. The three main models we experiment with are 1) WBOW model; 2) co-attention neural encoder model that includes LSTM and affinity matrix; 3) coattentative convolutional neural network model.

We train and evaluate our models on the classical Penn Discourse Treebank (PDTB) dataset. We obtain competitive accuracy performance on the classification of the discourse relations with PDTB testing set and compare our result with the baseline model.

## 2 Background and Related Work

Early attempts on discourse parsing mostly rely on handcraft feature sets and language models. Rutherford and Xue (2014) employs Brown cluster pairs to represent discourse relation and incorporate coreference patterns to identify senses of implicit discourse relations in naturally occurring text. These attemps often result in feature sparsity problems and it is often hard to build a rich and reliable feature set.

More recent models commonly use neural-network based methods. Liu et al. (2016) uses related discourse classification tasks specific to a corpus to exploit the combination of different discourse corpora, and propose a Convolutional Neural Network embedded multi-task learning system to synthesize these tasks by learning both unique and shared representations for each task. This approach improves significantly on the PDTB implicit discourse relation classification. Ji, Haffari and Eisenstein (2016) uses latent variable recurrent neural network architecture for jointly modeling sequences of words and (possibly latent) discourse relations between adjacent sentences. They successfully improves the performance on implicit discourse relation classification and obtain a discourse informed language model.

## 3 Models

In this project, we implemented and tested a simple baseline model and three advanced models. We now descibe them in more details below:

### 3.1 Baseline Model: Bag of Words

We use a simply average of word embeddings as the sentence embedding in the baseline model. We use the pre-processed GloVe word vectors as the initial word embeddings. This is the same for all the following models.

### 3.2 Weighted Bag of Words

The weighted bag of words model improves upon the simple bag-of-words model. Instead of using a simple average over all words in the sentence, we use a scoring deep neural network on each word to produce a weight per word.

Specifically, let $X^L = (x_1^L, x_2^L, \cdots, x_m^L)$ denote the sequence of word vectors for left sentence, and $X^R = (x_1^R, x_2^R, \cdots, x_n^R)$ denote the same for right sentence. We run an independent DNN on each word: scores for left sentence is $s^L = (s_1^L, s_2^L, \cdots, s_m^L) \in \mathbb{R}^m$, scores for right sentence is $s^R = (s_1^R, s_2^R, \cdots, s_n^R) \in \mathbb{R}^n$.

Then we normalize within each sentence to get weights for every word in that sentence.

$$w^L = \text{softmax}(s^L) \in \mathbb{R}^m, w^R = \text{softmax}(s^R) \in \mathbb{R}^n \tag{1}$$

We take the weighted average of word vectors in each sentence as the sentence embedding.

$$L = w^L \circ X^L, R = w^R \circ X^R \tag{2}$$

We concatenate the sentence embeddings of left and right and predict the classification with DNN classifier.

Figure 1 illustrates the WBOW model with a DNN classifier on top of it.

### 3.3 Co-attention Neural Encoder

The co-attention neural encoder model is inspired by the design of encoder in [2]. It mainly focus on learning sentence embeddings with affinity matrix representing relationship between word pairs from left and right sentences.

Let $(x_1^L, x_2^L, \cdots, x_m^L)$ denote the sequence of word vectors for left sentence, and $(x_1^R, x_2^R, \cdots, x_n^R)$ denote the same for right sentence. Sentences are then encoded using separate LSTMs: $l_t =$

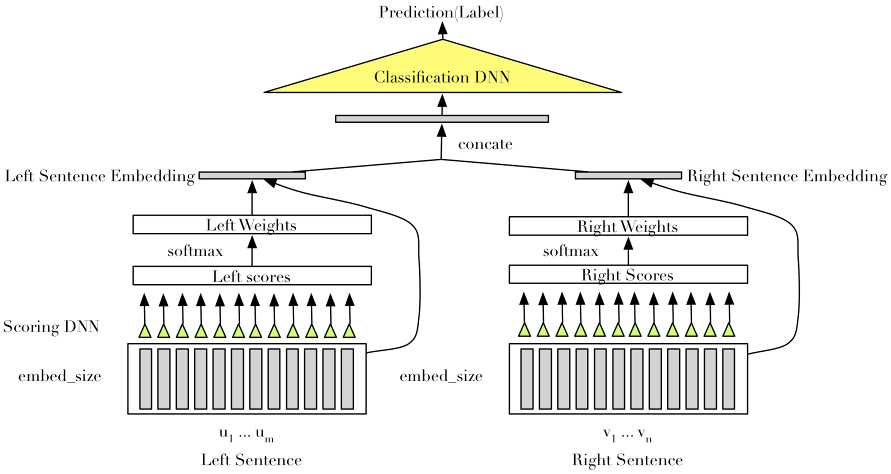

Figure 1: Weighted Bag of Words Model

LSTM$_{left}(l_{t-1}, x_t^L), r_t$ = LSTM$_{right}(r_{t-1}, x_t^R)$. The left sentence matrix is $L = [l_1 \cdots l_m] \in \mathbb{R}^{h \times m}$. Similarly, right sentence matrix is $R = [r_1 \cdots r_n] \in \mathbb{R}^{h \times n}$.

We then compute affinity matrix $A = R^T L \in \mathbb{R}^{n \times m}$, with entry $A[i, j]$ representing affinity score of i-th word in right sentence and j-th word in left sentence. Then we normalize A row-wise to obtain the attention weights $A^L$ across the right sentence for each word in the left sentence, and column-wise to produce $A^R$ across the left sentence for each word in the right sentence.

$$A^L = \text{softmax}(A) \in \mathbb{R}^{n \times m}, A^R = \text{softmax}(A^T) \in \mathbb{R}^{m \times n} \tag{3}$$

We use the normalized affinity matrices $A^L$ and $A^R$ to get the summary of the left(right) sentence in light of each word of the right(left) sentence, and use the average as the final embedding of the sentences.

$$C^L = LA^R \in \mathbb{R}^{h \times n}, C^R = RA^L \in \mathbb{R}^{h \times m} \tag{4}$$

Figure 2 illustrates the co-attention model with a DNN classifier on top of it.

### 3.4 Coattententive Convolutional Neural Network

In this co-attentative convolutional neural network model, we propose this novel idea of using a weighted combination of the outer product matrices of different pairs of word embedding vectors as a compact and rich 2-dimensional image-like representation of the inter-sentential relationship information between a pair of sentences or paragraphs. We build upon the model of coattention encoder to learn a meaningful set of weights for each potential pair of words from the two sentences, and then use this set of weights to combine the word embedding outer product matrices into a single composite sentence-pair embedding matrix, which is then fed into a convolutional neural network as input for the discourse relation classification task.

The detailed design of our coattentative CNN model can be described as follow:

Let $(x_1^L, x_2^L, \cdots, x_m^L)$ denote the sequence of word vectors for left sentence, and $(x_1^R, x_2^R, \cdots, x_n^R)$ denote the same for right sentence. Sentences are then encoded using separate LSTMs: $l_t$ = LSTM$_{left}(l_{t-1}, x_t^L), r_t$ = LSTM$_{right}(r_{t-1}, x_t^R)$. The left sentence matrix is $L = [l_1 \cdots l_n] \in \mathbb{R}^{h \times n}$. Similarly, right sentence matrix is $R = [r_1 \cdots r_m] \in \mathbb{R}^{h \times m}$.

We then compute affinity matrix $A = L^T R \in \mathbb{R}^{m \times n}$, with entry $A[i, j]$ representing affinity score of i-th word in left sentence and j-th word in right sentence. Next we flatten the affinity matrix $A$ into a $(m \times n)$-dimensional affinity vector, and then take softmax over this affinity vector to obtain

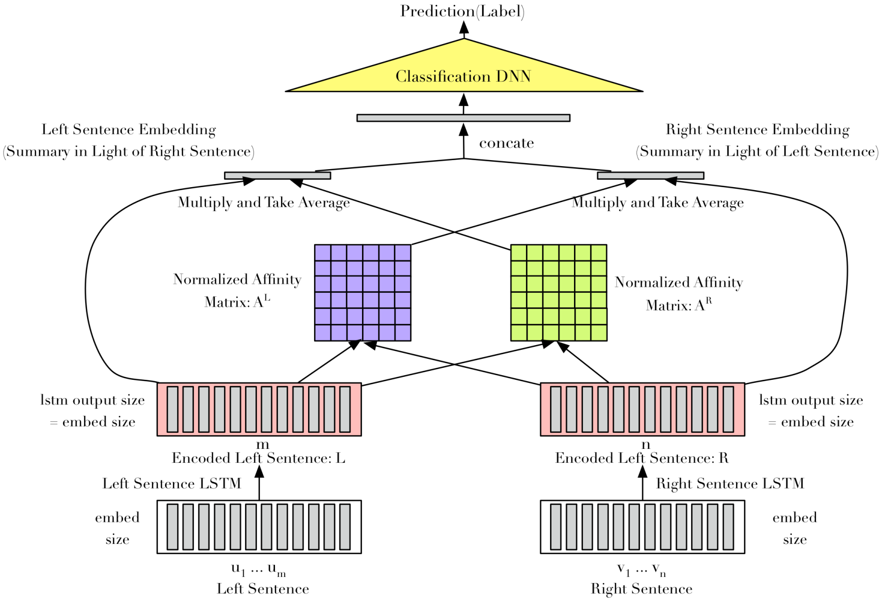

Figure 2: Co-attention Model

a weight vector $[w_{1,1}, w_{1,2}, ..., w_{1,n}, w_{2,1}, w_{2,2}, ..., w_{2,n}, w_{3,1}, ......, w_{m,1}, ..., w_{m,n}]$, indicating the importance weight of each cross-pair of word embeddings from the left and right sentences.

Now for $i \in \{1, ..., m\}$, $j \in \{1, ..., n\}$, we compute the outer product between the pair of word embeddings $x_i^L$ (the i-th word in the left sentence) and $x_j^R$ (the j-th word in the right sentence): $x_i^L \otimes x_j^R = x_i^L \cdot x_j^{R^T}$. We then combine all these $m \times n$ different outer product matrices into a single composite matrix $H = \sum_{i=1}^{m} \sum_{j=1}^{n} w_{i,j} \cdot x_i^L \otimes x_j^R$. Now this composite matrix $H$ is in the form of a two-dimensional spatial representation of the semantic relationship information between the pair of sentences, and we can feed $H$ into a convolutional neural network to classify it into one of discourse relation sense label through recognizing certain regularities and patterns in this image-like representation $H$.

Figure 3 illustrates the overall architecture of our coattentative convolutional neural network model.

## 4 Experiments

### 4.1 Datasets

We train and evaluate our models on Penn Discourse Tree Bank(PDTB) dataset.

Each discourse(sentence) pair is classified into one of the 5 discourse relations:

- Explicit: relations realized explicitly by connectives.
- Implicit: relations between two adjacent sentences in the absense of an explicit connective.
- AltLex: the insertion of an implicit connective to express an inferred relation leads to a redundancy due to the relation being alternatively lexicalized by some non-connective expression.
- EntRel: only an entity-based coherence relation could be perceived between the sentences.
- NoRel: no discourse relation or entity-based relation could be perceived between the sentences.

Explicit,Implicit and AltLex relations have sense annotations. We use first-level sense tags: ”TEMPORAL”, ”CONTINGENCY”, ”COMPARISON” and ”EXPANSION” with ”NONE”(for the other

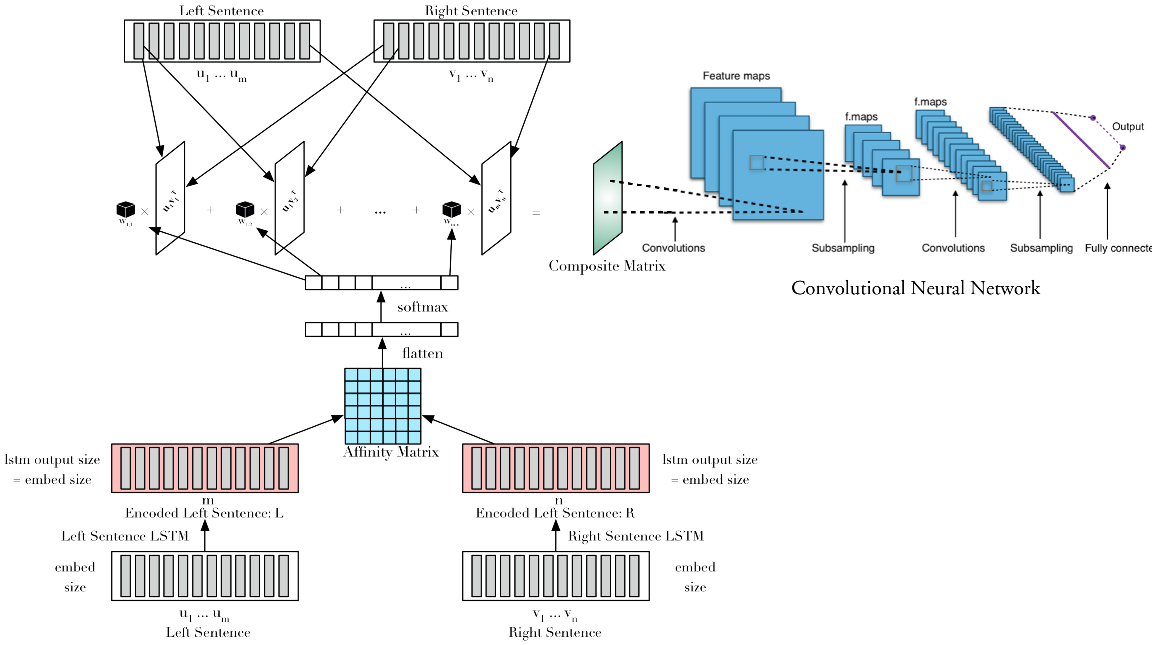

Figure 3: Our Coattentative Convolutional Neural Network Model

2 discourse relations) as 5 labels for prediction. If a pair have multiple interpretations, we use only the first semantic class.

Table 1: Distribution of Relations in PDTB

| PDTB Relations | No. of tokens |
|---|---|
| Explicit | 18459 |
| Implicit | 16224 |
| AltLex | 624 |
| EntRel | 5210 |
| NoRel | 254 |
| Total | 40600 |

Table 2: Distribution of ”CLASS” sense tags in PDTB

| ”CLASS” | Explicit(18459) | Implicit(16224) | AltLex(624) | Total |
|---|---|---|---|---|
| ”TEMPORAL” | 3612 | 950 | 88 | 4650 |
| ”CONTINGENCY” | 3581 | 4185 | 276 | 8042 |
| ”COMPARISON” | 5516 | 2832 | 46 | 8394 |
| ”EXPLANATION” | 6424 | 8861 | 221 | 15506 |
| Total | 19133 | 16828 | 634 | 36592 |

### 4.2 Implementation Details

We use 300-dimensional GloVe word vectors pretrained on Wikipedia 2014 and Gigaword 5 corpus. We set embeddings for out-of-vocabulary words to random vectors (drawing each term from standard Gaussian distribution). We use a max sentence length of 100, classifier's hidden layer size of 100, dropout rate 0.5, LSTM hidden state size in the co-attention model same as word embedding size 300. We use Sections 2 to 21 of PDTB data for training, Section 22 for development and Section 23 for testing.

### 4.3 Results

#### 4.3.1 Evaluation

We trained all of our model on Microsoft Azure with tensorflow 1.0 and GPU acceleration. The final classification accuracy results on the PDTB test set is summarized in table 3.

Table 3: Accuracy

| Model | Accuracy |
|---|---|
| BOW (baseline) | 55.53 % |
| WBOW | 64.19 % |
| Coattention Neural Encoder | 67.27 % |

and the corresponding confusion matrices for the above three models on the PDTB test set are:

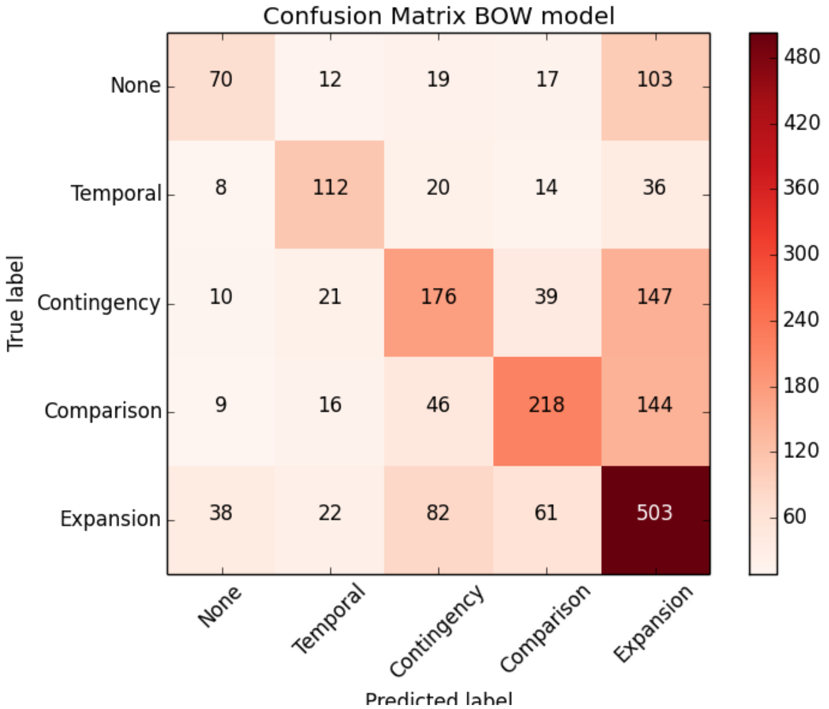

Figure 4: Confusion Matrix for the Baseline Bag-of-Words Model

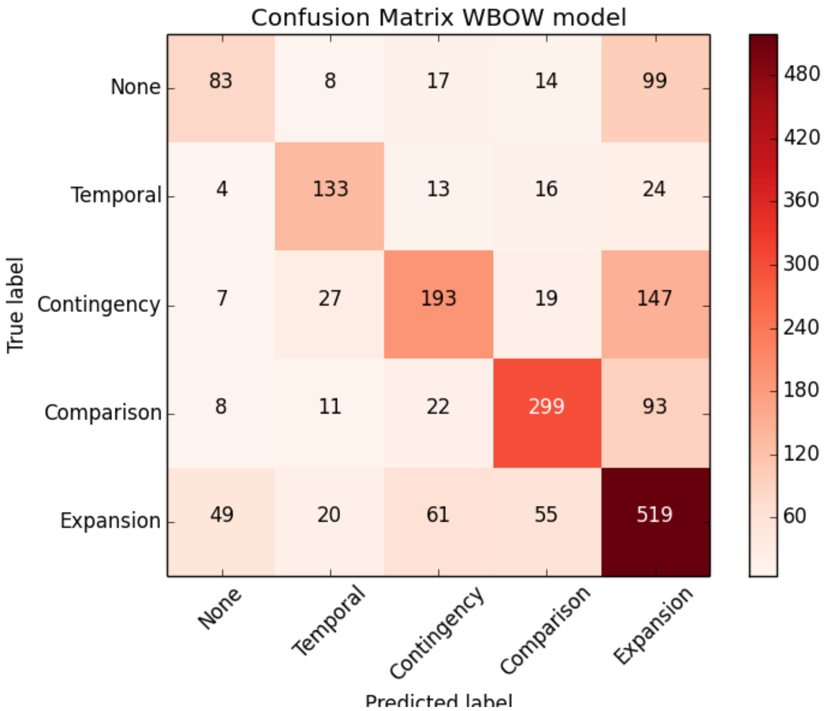

Figure 5: Confusion Matrix for the Weighted Bag-of-Words Model

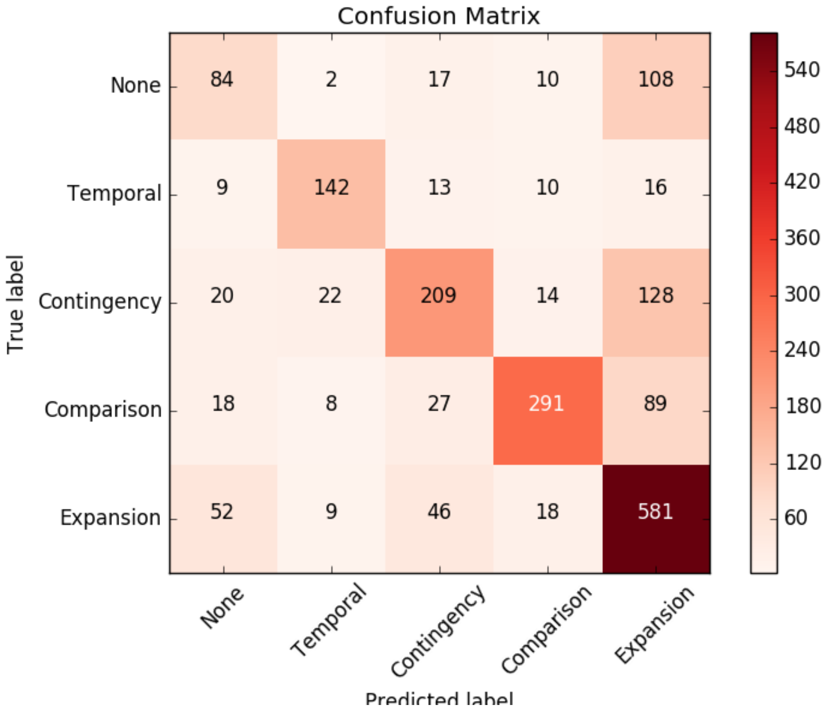

Figure 6: Confusion Matrix for the Co-attention Neural Encoder Model

We then trained and tested our co-attentative convolutional neural network model on the much more challenging task of classifying only the sentence pairs in the “implicit” category. Our coattentative CNN model yields a pretty competitve accuracy of 54.7 %. The corresponding confusion matrix is:

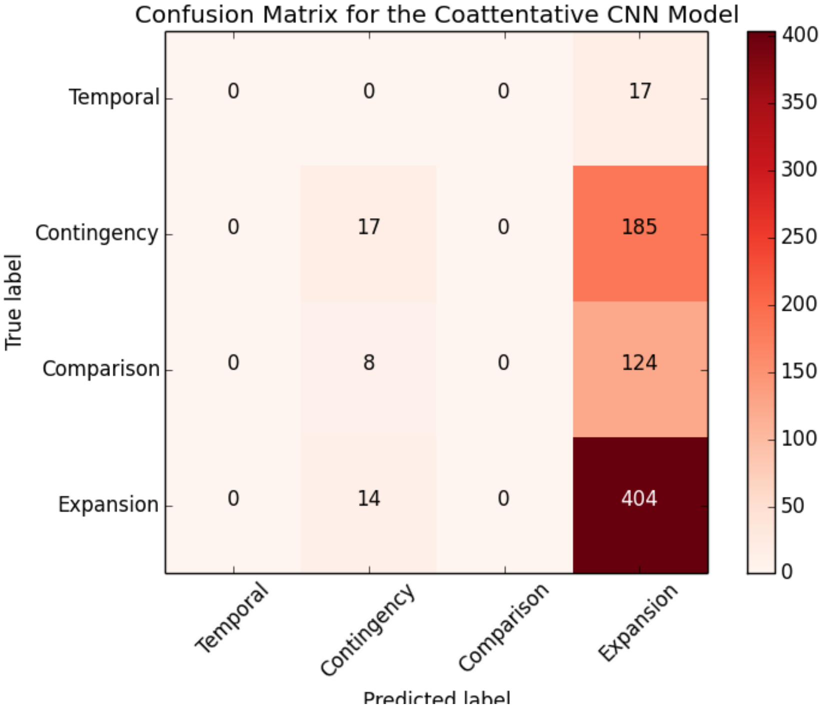

Figure 7: Confusion Matrix for the Co-attention Neural Encoder Model

### 4.3.2 Error Analysis

In this part, we analyze the error our model makes with respect to different sense labels and discourse relations. Because of the limit of space, the analysis is only based on co-attention model.

For explicit discourse relation, it’s more easy to predict the sense label and has a higher accuracy in fact. Errors occur when the explicit connective can have different meanings in different context. Heres an example:

> true label: Contingency, predicted label: ”Temporal”
> left sentence: But you learn to live with it
> right sentence: when something is inevitable

It should be labeled ”Contingency” but is misclassified as ”Temporal” because of polysemy of the word ”when”.

For implicit discourse relation, except the tendency to predict the most common label ”Expansion”, there are other types of misclassification. For example, the model fails to discover the ”Expansion” relationship (predict ”None”), confuses ”Contingency” and ”Comparison”, which are quite similar in some sense, and is not able to handle negative words.

> true label: ”Expansion”, predicted label: ”None”
> left sentence: The bills will be dated Oct. 26 and will mature Oct. 25, 1990
> right sentence: They will be available in minimum denominations of $10,000

> true label: ”Comparison”, predicted label: ”Contingency”
> left sentence: Hooker’s philosophy was to build and sell
> right sentence: We want to build and hold

true label: ”Expansion”, predicted label: ”Contingency”
left sentence: The underlying economy remains sound
right sentence: There is nothing wrong with the economy

## 5 Conclusion

We have proposed three different end-to-end neural-network models for discourse parsing classification on top of a simple baseline bag of words model: 1) a weighted bag of words model using DNNs to score and calculate weights; 2) a co-attention model using coattention affinity matrix on top of LSTMs; 3) a convolution neural network combined with coattention affinity matrix. We have shown that our implementation of weighted bag of words model, co-attention model and coattentive convolution neural network model perform on relatively similar accuracy in the overall classification task, and all perform much better than the simple baseline model. We have also shown that in both the explicit and implicit discourse relations, our models can pick up the logical clues in the sentences and give correct predictions.

Future work may include 1) Tuning on the hyper-parameters in the above given models. 2) Our models still perform much better in predicting explicit discourse relations but not so well in implicit discourse relations. We should investigate more on techniques to improve predicting implicit discourse relations. 3) In the prediction of implicit discourse relations, our models have a tendency to predict the label “Expansion” over all other labels. We should investigate why this is the case. 4) We also plan to experiment with using the idea skip-thought vector to help us initialize our LSTM parameters. This could potentially give us some information about the entire corpus before classification on local sentences.

### Contribution

Borui Wang (wbr): implementation of bag-of-words baseline model, implementation of WBOW model, implementation of coattentive CNN model, poster, error analysis, paper writing

Yang Yuan (yyuan16): PDTB dataset preprocessing, implementation of coattention model, poster making, error analysis, paper writing

Alex Fu (xiaofu): implementation of coattention model, experiment with skip-thought model, poster, paper writing, maintaining GPU

### Acknowledgments

We would like to thank our advisors Danqi Chen, Richard Socher and Professor Manning for their generous help and valuable guidance throughout the process of this project. We would also like to thank all the CS 244N teaching staff for a wonderful journey of deep learning in NLP this term. Thank you so much! :)

### References

[1] Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Skip-thought vectors. *Advances in Neural Information Processing Systems* (pp. 3294-3302).

[2] Xiong, C., Zhong, V., & Socher, R. (2016). Dynamic Coattention Networks For Question Answering. arXiv preprint arXiv:1611.01604.

[3] Ji, Y., Haffari, G., & Eisenstein, J. (2016). A latent variable recurrent neural network for discourse relation language models. arXiv preprint arXiv:1603.01913.

[4] Rutherford, A., & Xue, N. (2014, April). Discovering Implicit Discourse Relations Through Brown Cluster Pair Representation and Coreference Patterns. In EACL (Vol. 645, p. 2014).

[5] Liu, Y., Li, S., Zhang, X., & Sui, Z. (2016). Implicit discourse relation classification via multi-task neural networks. arXiv preprint arXiv:1603.02776.

[6] Prasad, R., Miltsakaki, E., Dinesh, N., Lee, A., Joshi, A., Robaldo, L., & Webber, B. L. (2007). The penn discourse treebank 2.0 annotation manual.

[7] Prasad, Rashmi, et al. Penn Discourse Treebank Version 2.0 LDC2008T05.

[8] Pennington J, Socher R, Manning C D. Glove: Global Vectors for Word Representation[C]//EMNLP. 2014, 14: 1532-1543.