Natural Language Processing Course Project

Discourse-based Sentence Representations in Hindi

Ujwal Narayan

20171170

Saujas Vaduguru

20171098

Abstract

Learning effective general-purpose sentence representations is an important core task in NLP. Existing methods for require very large amounts of data, or extensive manual annotation. In this project, we apply a method inspired by Nie et al. (2017) to build sentence representations for Hindi sentences that performs comparably to existing methods, using much less data. The method leverages explicit discourse relations to set up a discourse marker prediction task. This task is used to train a BiLSTM based sentence encoder model. The model is evaluated on a transfer task – sentiment analysis – and results are compared to existing sentence embedding methods.

1 Introduction

General-purpose sentence embedding models have a wide variety of applications in NLP, including tasks such as sentiment analysis, natural language inference, paraphrase detection, summarisation, and many more.

Existing approaches to sentence embeddings are trained using the task of predicting a random omitted word given its context Devlin et al. (2018) in a very large corpus, or using a specific task like natural language inference Conneau et al. (2017). For Hindi, we do not have extensive data for natural language inference, and using large transformer-based language models like Devlin et al. (2018) is highly resource intensive.

Nie et al. (2017) proposed selectively omitting words in a way that allows identification of deep relations between words in a sentence, and training a model to predict the omitted word. These models do not learn to predict randomly omitted words in the corpus, but instead learn to predict a specific set of words. But, unlike predicting NLI like in Conneau et al. (2017), this approach does not require extensive amounts of labelled data.

2 Discourse Prediction Task

Nie et al. (2017) proposed that discourse markers, which mark conceptual relations between ideas in a sentence are words that humans explicitly use to indicate deep conceptual relations. Additionally, there are a small number of discourse markers that are used, and predicting the discourse markers is a strong training task for learning sentence representations.

The discourse prediction task is therefore the task of predicting the missing discourse marker in a sentence, given the context of words in the sentence around the discourse marker. For example, given

इस क्षेत्र की व्यवस्था पहले बहुत अच्छी नहीं थीं __ यह पहले कभी किसी राज्य का अभिन्न अंग नहीं रहा था

we can complete the sentence with क्योंकि.

We split each sentence containing a discourse marker into three parts, which we label S_1, S_2, and DM, where S_1 is the part of the sentence preceding the discourse marker, S_2 the part of the sentence after the discourse marker, and DM is the dicourse marker itself. The task then becomes the prediction of DM given S_1 and S_2.

It must be kept in mind that it is not always possible to predict the discourse marker given the context. In a sentence like *I missed my flight ____ there was heavy traffic on the way*, we can predict that the sentence is completed by *there*, but *I went home ____ I missed the class* could be completed by either *so* or *because*. Nevertheless, in many of the cases, it is possible to predict the discourse relation, so we use it as a training task.

3 Related Work

Our method and experiments are based on Nie et al. (2017), which introduces the idea of using discourse prediction to train sentence embeddings.

Another paper that uses a supervised task to train general-purpose sentence embeddings is by Conneau et al. (2017). The train sentence embeddings by training an encoder to encode the premise and the hypothesis, and use the sentence encodings to recognise entailment. This informed training of sentence embeddings shows good results and improvements over baselines, but large NLI datasets are not available for many languages. The best model from Conneau et al. (2017) is used as the basis for the model in both Nie et al. (2017) and our work.

We compare our results against unsupervised sentence embedding method presented – sent2vec – in Pagliardini et al. (2018), and against the baseline method of averaging word embeddings. sent2vec embeddings are trained in an unsupervised manner using n-gram based features.

Arora et al. (2016) show a simple embedding method that provides a strong baseline for sentence embeddings, but also show the power of simpler methods like unweighted averaging of word embeddings.

4 Data Extraction

We use dependency parsing to extract sentences containing discourse markers and split them into the two parts and the label used to train the model on the discourse prediction task. Each of these parts maybe a sentence, or a subordinate clause, and the label is the discourse marker. We limit our extraction to sentences that have an explicitly marked discourse marker, and further use only discourse markers that occur contiguously in Hindi sentences, as opposed to discourse markers that may occur in different places within a sentence.

The discourse markers are chosen to be the frequently occurring explicit markers in the Hindi Discourse Treebank. The Hindi Discourse Treebank has ∼700 explicit discourse relations marked, and we choose the markers that occur more than 10 times in the discourse treebank.

For each discourse marker, we provide a template that is used to check whether the dependency tree is of a particular form. The template is a tree which has the dependency marker as one of its nodes, and two other nodes corresponding to S_1 and S_2. If the template is matched, it specifies dependency nodes that are the roots of subtrees corresponding to S_1 and S_2. We can then extract the three elements we need for the discourse prediction task. Since the subtrees might have short noun phrases or extremely short verb phrases, we set a minimum length threshold and extract only those pairs where S_1 and S_2 are longer than the threshold length. Some examples of extracted are shown in Table 2.

Nie et al. (2017) use a corpus of books, where discourse relations can be extracted not only within as a sentence, as we do in this project, but also from the sentence immediately before. In some cases, a sentence might have only S_2 and the discourse marker, in which case S_1 si the previoius sentence. The corpus we extract the discourse prediction task from isn't composed of clear discourse units like a book. In this corpus, we are not guaranteed to find S_1 in the previous sentence, since the sentence may be unrelated. So, we chose to extract only pairs within the same sentence.

4.1 Dataset Statistics

We used a portion around (10%) of the Hindi Monlongual Corpus Kunchukuttan et al. (2017), totalling around 46,99,914 sentences. Extracting the sutiable sentence pairs based on the handwritten rules, we have around 364076 sentence pairs for the HinDiSent model.

Count of Discourse Markers	Discourse Marker
650	इसलिए
282486	और
8413	क्योंकि
6401	जबकि
22781	तथा
13210	तो
29920	लेकिन
215	हालांकि

Table 1: Dataset statistics

5 Model

We note the similarity of the discourse prediction task to natural language inference, which is predicting a class given two sentences (in the inference setting it is the premise and hypothesis, here it is S_1 and S_2) and follow Nie et al. (2017)

S_1	DM	S_2
अंग्रेज तो वह साहस नही कर पाए	लेकिन	मालवीय जी की आशंका स्वतंत्र भारत में सही साबित हुई
प्रेम को हम आदर्शवाद के रूप में चुनते	जबकि	नफरत हमारे अन्दर एक प्रवृति के रूप में होती है
जब केवल पंचगंगा घाट में ही देव दीपावली मनाई जाती थी	और	बगल के दुर्गाघाट में दुर्गाघाटी

Table 2: Examples of extracted sentences

in using the InferSent model presented in Conneau et al. (2017). We use the best model from InferSent, which is a BiLSTM with max-pooling. The model is defined as follows:

$$\vec{h}_t = \text{LSTM}_t(w_1, \ldots, w_n|\theta_1) \quad (1)$$

$$\overleftarrow{h}_t = \text{LSTM}_t(w_1, \ldots, w_n|\theta_1) \quad (2)$$

$$h_t = [\vec{h}_t; \overleftarrow{h}_t] \quad (3)$$

$$s_i = \text{MaxPool}(h_1, \ldots, h_n) \quad (4)$$

$$s_{\text{avg}} = \frac{1}{2}(s_1 + s_2) \quad (5)$$

$$s_{\text{sub}} = s_1 - s_2 \quad (6)$$

$$s_{\text{mul}} = s_1 * s_2 \quad (7)$$

$$S = [s_1, s_2, s_{\text{sub}}, s_{\text{mul}}, s_{\text{avg}}] \quad (8)$$

To allow the classifier to use non-linear combinations of features, we use the element-wise product s_{mul}. To capture order-invariant information, we use s_{avg}, and to get order-specific information, we use s_{sub}.

6 Experiments

The pipeline we use is shown in Figure 1. To train our models, we use stochastic gradient descent with initial learning rate 0.1, and anneal by the factor of 5 each time validation accuracy is lower than in the previous epoch. We train our sentence encoder model for 20 epochs, and use early stopping to prevent overfitting. We also clip the gradient norm to 5.0. We follow Nie et al. (2017) in not using dropout in the fully connected layer. All models we report used a 2048 hidden units.

We train different models for different subsets of the discourse markers – top 5 and top 8 – to observe the effect of adding more discourse markers.

We also create two baselines for comparison.

- **Sent2Vec** Pagliardini et al. (2018) trained on the same raw Hindi monolingual corpus from which the data for the HinDiSent was extracted.
- **Averaged Bag of Words** Arora et al. (2016) We use the 300 dimensional pretrained Hindi word embeddings from FastText, and for each sentence take the average of the embeddings for all the word in that sentence.

To evaluate the sentence embeddings, we use the downstream task of Sentiment Analysis. For this we use the "Aspect Review Corpus"Akhtar et al. (2016). We ignore the reviews where the annotators could not agree on the sentiment. We also ignore reviews with multiple sentiment, resulting in around 2000 reviews for classification.

7 Results and Discussion

7.1 Discourse Prediction Task

On the discourse prediction we observed the following accuracies.

Number of Discourse Markers	Validation Accuracy	Test Accuracy
Top 5	77.38	77.03
Top 8	77.15	76.8

As we can clearly see, the model seems to have learn to predict the discourse task reasonably well. The validation accuracy is in line with the test accuracy implying that the model has generalized well. Graphs for the performance of different models, including baselines, are shown in Figures 2, 3, 4, and 5.

7.2 Sentiment Analysis

As stated earlier, we use the Sentiment Analysis task to extrinscly evaluate our sentence embedding. As we can see Gaussian Proces is the best performing classifier for all the three embeddings, with it peaking out for HinDiSent with 5 discourse markers.

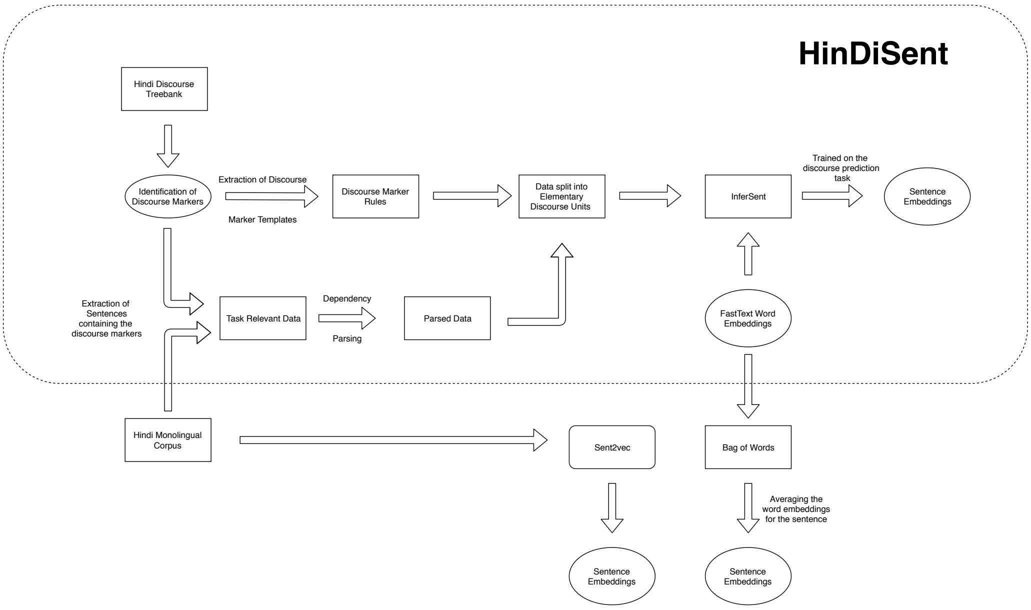

Figure 1: Pipeline

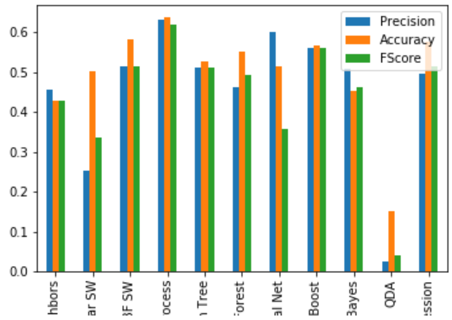

Figure 2: Performance of HinDiSent with 5 Discourse Markers

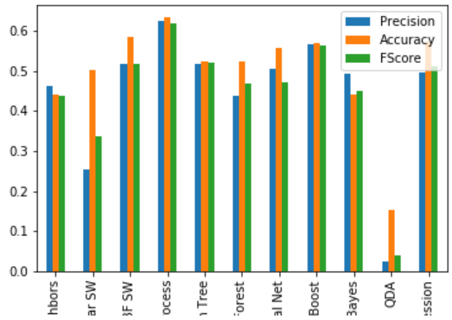

Figure 3: Performance of HinDiSent with all the chosen Discourse Markers

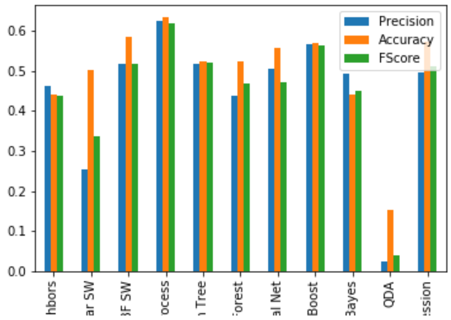

Figure 4: Performance of Sent2Vec

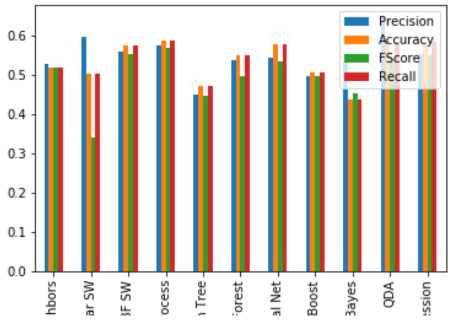

Figure 5: Performance of Avergaed Fasttext BoW

7.2.1 Length of the Sentence Embeddings

Furthermore, we split the test corpus based on the lengths of the reviews and grouped them into buckets of size 5. We evaluated the higher scoring models, namely Gaussian Process, Logistic Regression and Multi Layer Perceptrons. The number of sentences peaked out at the bucket containing 15-20 sentences, and at the higher end with sentences containing with 50+ words we found a scant few. Looking at the plots for the various metrics we see that, HinDiSent performs the highest with sentences having 0-5 words. Performance remains more or less constant for Gaussian Process, but Logistic Regression takes a steep dive. Performance varies minimally with increase in the sentences until you reach the larger end. This could be due to the lack of training data, as the model seems to scale well for the rest of the sentences.

8 Conclusions

In this project, we have explored the idea of leveraging discourse markers to improve sentence representations in Hindi. We use an automatic curation method to create a dataset for a discourse prediction task, and train a model to learn sentence representations by predicting the discourse markers given context. We then use these sentence representations to predict sentiment, and find that our model competes with unsupervised sentence embedding methods trained on much larger corpora.

Code and data is available over here `https://drive.google.com/drive/folders/1H6UTgqLxMJTHiHDfzpL7BsfKyxuUVT64?usp=sharing`

References

Md. Shad Akhtar, Asif Ekbal, and Pushpak Bhattacharyya. 2016. Aspect based sentiment analysis in hindi: Resource creation and evaluation. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2016. A simple but tough-to-beat baseline for sentence embeddings.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2017. The IIT bombay english-hindi parallel corpus. *CoRR*, abs/1710.02855.

Allen Nie, Erin D. Bennett, and Noah D. Goodman. 2017. Dissent: Sentence representation learning from explicit discourse relations.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised learning of sentence embeddings using compositional n-gram features. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*.