Textual Coherence using different Neural Networks

Yash Agrawal
2020114005
yash.a@research.iiit.ac.in

Eshika Khandelwal
2020114018
eshika.k@research.iiit.ac.in

Adith John Rajeev
2020114010
adith.r@research.iiit.ac.in

I. Introduction

Textual coherence in Linguistics is what makes the text semantically meaningful. Through this project we analyse the performance of different models over various data-sets to measure the coherence levels.

II. Datasets used

A. Grammarly Corpus of Discourse Coherence

The dataset is annotated into classes 1,2 and 3 where 3 denotes the most coherent paragraph. It consists of four training datasets (1000 paragraphs each) and four testing datasets (200 paragraphs each). We have merged the four training datasets along with three testing datasets to train our model (4600 paragraphs) and used the remaining test data to test the accuracy of our model. Some datasets being more open-domained than the others, we decided to vary the test dataset chosen in order to compare the results.

Additional details regarding the dataset can be found here

B. Wikipedia-CNN Dataset

The dataset consists of coherent text from Wikipedia and the CNN/Daily news sets. The text dataset also contains respective replacements to make every paragraph incoherent. These replaced sets are used as negative samples in our model.

The dataset can be found here and this paper can be referred for the same.

III. Models

A. RNN

A fully-connected RNN where the output from previous timestep is to be fed to next timestep.

B. GRU

Gated recurrent units (GRUs) are a gating mechanism in recurrent neural networks. The GRU is like a long short-term memory (LSTM) with a forget gate, but has fewer parameters than LSTM, as it lacks an output gate. GRU is more memory efficient and gives results faster than LSTM. GRU exposes the complete memory and hidden layers but LSTM doesn't.

C. LSTM

LSTM networks are a type of RNN that uses special units in addition to standard units. LSTM units include a 'memory cell' that can maintain information in memory for long periods of time. This memory cell lets them learn longer-term dependencies. LTSMs are used for larger datasets and usually give better results than other recurrent neural networks.

IV. Approaches Used

The following approaches were used on the GCDC corpus to fine tune the models:

A. Varying Test data

Some test data in the GCDC corpus was more closed-domained than the others leading to varying accuracies. For example, Yahoo corpus was based on the Comprehensive question and answers and was thus more open domained.

B. Three way classifier to binary classifier

We trained our model on the GCDC corpus using the default annotation to make a three-way multi-classifier. Later, we switched this approach, remodelled our data to binary labels- coherent and incoherent- and used this data on a binary classifier to observe better results.

C. Similarity

Similarity between sentences is useful in scoring coherence as a higher degree of similarity between neighbouring sentences indicates a more coherent text.

- Initially we implementing a function that calculates the cosine similarity between two input strings.
- Using the first sentence of the paragraph as the base, we recursively check its similarity with the subsequent sentence, and then added it to the base. This approach led to a decrease in accuracy.
- In the second approach, we averaged the similarity obtained between adjacent pairs of sentences, and used it as a parameter in the classification. This approach of mean similarity still gave us negligible gains in the results.
- Finally, we took the minimum of the all the obtained similarity scores within a text. This approach was based on the fact that 'a chain is as strong as its weakest link', the slightest occurrence of incoherence in a paragraph makes it incoherent and it cannot be compensated by a sensible structure before or after it. Using this minimum

similarity as a parameter we observed a considerable increase in our accuracy.

V. Results and Conclusion

A. Accuracy Obtained

a) Varying test data in GCDC Corpus: Results obtained after running the LSTM model with Binary classifier and minimum similarity as a parameter was as follows:

TABLE I
Accuracy Results with different test data

	Corpus used for Test Data			
	Clinton	***Enron***	***Yahoo***	***Yelp***
Accuracy %	61.0%	66.0%	54.5%	56.5%

b) LSTM:

- GCDC Corpus
  - Without using similarity as a parameter
    - 3000 training data, 600 testing data, three way classifier: approx **30%**
    - 4600 training data, Yahoo test data (200), three way classifier: approx **36.5%**
    - 4600 training data, Yahoo test data (200), binary classifier: approx **55%**
    - 4600 training data, Clinton test data (200), binary classifier: approx **64.99%**
  - Using average similarity as a parameter
    - 4600 training data, Clinton test data (200), three way classifier: approx **34%**
  - Using minimum similarity as a parameter
    - 4600 training data, Clinton test data (200), three way classifier: approx **39.5%**
    - 4600 training data, Clinton test data (200) , binary classifier: approx **67%**

  Binary classifier performed significantly better that three way multi classifier, thus we used only binary classifiers for the Wikipedia-CNN corpus.
- Wikipedia-CNN Corpus
  - Without using any similarity parameter: approx **71.66%**
  - Using minimum similarity parameter: approx **74.55%**

c) GRU:

- GCDC Corpus
  - Without using similarity as a parameter
    - 4600 training data, Clinton test data (200), three way classifier: approx **41.99%**
    - 4600 training data, Clinton test data (200), binary classifier: approx **57.99%**
  - Using minimum similarity as a parameter
    - 4600 training data, Clinton test data (200) , binary classifier: approx **63.99%**
- Wikipedia-CNN Corpus

  Binary classifier was used for analysis.
  - Without using any similarity parameter: approx **69.89%**
  - Using minimum similarity parameter: approx **77.76%**

d) RNN:

- GCDC Corpus
  - Without using similarity as a parameter
    - 4600 training data, Clinton test data (200), three way classifier: approx **32.49%**
    - 4600 training data, Clinton test data (200), binary classifier: approx **55.5%**
  - Using minimum similarity as a parameter
    - 4600 training data, Clinton test data (200) , binary classifier: approx **53.5%**

B. Observations

Using Minimum Similarity as a parameter, we observe a significant rise of around 2-5% in the accuracy. This is in line with with the definition of coherence that demands a level of similarity in the flow of the text.

While varying the test data in the GCDC corpus on the LSTM Binary classifier with minimum similarity as a parameter we observe that the Enron dataset has the most accuracy, which implies that it probably has the most closed domain, while Yahoo has the most open domain.

Expanding our dataset by including some test data into training increased the accuracy of our model around by 6-7%. Using a much larger dataset, such as the Wikipedia-CNN corpus (100x larger than GCDC) showed an even larger increase in the accuracy.

References

[1] Jiwei Li and Dan Jurafsky. 2017. Neural Net Models of Open-domain Discourse Coherence. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 198–209, Copenhagen, Denmark. Association for Computational Linguistics.

[2] Peng Xu, Hamidreza Saghir, Jin Sung Kang, Teng Long, Avishek Joey Bose, Yanshuai Cao, and Jackie Chi Kit Cheung. 2019. A Cross-Domain Transferable Neural Coherence Model. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 678–687, Florence, Italy. Association for Computational Linguistics.

[3] Farag, Y. (2020). Neural approaches to discourse coherence: modeling, evaluation and application (Doctoral thesis).https://doi.org/10.17863/CAM.70024

[4] Abhishek, Tushar Rawat, Daksh Gupta, Manish Varma, Vasudeva. (2021). Transformer Models for Text Coherence Assessment.

[5] Aili Shen, Meladel Mistica, Bahar Salehi, Hang Li, Timothy Baldwin, Jianzhong Qi. Evaluating Document Coherence Modeling. Transactions of the Association for Computational Linguistics 2021; 9 621–640. doi: https://doi.org/10.1162/tacl$_{a}$0388