Incremental Neural Lexical Coherence Modeling

Sungho Jeon and **Michael Strube**
Heidelberg Institute for Theoretical Studies gGmbH
{sungho.jeon, michael.strube}@h-its.org

Abstract

Pretrained language models, neural models pretrained on massive amounts of data, have established the state of the art in a range of NLP tasks. They are based on a modern machine-learning technique, the Transformer which relates all items simultaneously to capture semantic relations in sequences. However, it differs from what humans do. Humans read sentences one-by-one, incrementally. Can neural models benefit by interpreting texts incrementally as humans do? We investigate this question in coherence modeling. We propose a coherence model which interprets sentences incrementally to capture lexical relations between them. We compare the state of the art in each task, simple neural models relying on a pretrained language model, and our model in two downstream tasks. Our findings suggest that interpreting texts incrementally as humans could be useful to design more advanced models.

1 Introduction

Coherence describes the semantic relation between elements of a text. It distinguishes a text as either a unified whole or a collection of unrelated sentences. Lexical coherence represents the cohesive effect achieved by lexical relations (Halliday and Hasan, 1976).

Earlier work mainly focuses on capturing lexical relations using external resources (Morris and Hirst, 1991). Mesgar and Strube (2016) introduce a graph model, the latest model for lexical coherence, to represent lexical relations between sentences on the graph. It encodes sentences as nodes and lexical relations between sentences as edges. This model, nevertheless, considers lexical items independently.

Recent neural models adopt a modern machine learning-based technique (Liu and Lapata, 2019; Gupta and Durrett, 2019), the Transformer (Vaswani et al., 2017). It relates all items simultaneously to capture semantic relations in sequences. More recently, large-scale pretrained language models, Transformer-based models pretrained on the massive amounts of text, have led to significant improvements in a range of NLP tasks (Devlin et al., 2019).

However, the Transformer processes texts in a way which is different from the way humans do it. Psycholinguistic experiments show that humans read texts incrementally (Marslen-Wilson, 1975; Kamide et al., 2003; Gibson and Warren, 2004). Köhn (2018) claim that NLP systems which follow this theory should interpret texts incrementally, too. Do neural models benefit from both pretrained language models and incremental sentence processing?

To investigate this question, we propose a coherence model which interprets sentences incrementally to capture lexical relations. For the ongoing sentence being read, our model first captures a semantic centroid vector which represents the centroid of preceding sentences. The centroid vector is computed as averaged representations of sentences. The model then measures semantic similarity between the centroid vector and the current sentence. Our model iterates this procedure for all sentences.

Proceedings of the 28th International Conference on Computational Linguistics, pages 6752–6758
Barcelona, Spain (Online), December 8-13, 2020

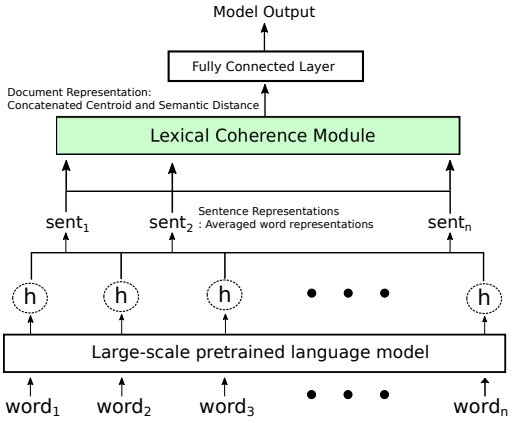

Figure 1: An overview of our model.

Algorithm 1 Incremental processing module.

procedure IPM($sent\_list$)
  $dist\_vec \leftarrow \{\}$
  for $sent_i$ in $sent\_list$ **do**
    $centroid_i$ = avg($sent_1 : sent_{i-1}$)
    $dist_i$ = dist($centroid_i$, $sent_i$)
    $dist\_vec$.append($dist_i$)
  end for
  $dist\_vec$ = conv($dist\_vec$)
  $dist\_vec$ = max_pool($dist\_vec$)
  return $centroid_{last}$, $dist\_vec$
end procedure

We evaluate our model on two tasks: assessing discourse coherence and automated essay scoring. We compare our model with the state of the art in each task and two variants of a simple baseline relying on a pretrained language model: the first baseline encoding sentences individually, and the second baseline encoding a whole text at once[1].

2 Related Work

Morris and Hirst (1991) propose lexical chains which identify sequences of related words using a lexical knowledge base. To identify lexical relations without human annotation, generative models have been developed, which learn lexical distributions. However, they may not generalize well across multiple datasets drawn from different distributions (Eisenstein and Barzilay, 2008; McNamara et al., 2010).

Mesgar and Strube (2016) propose a graph-based model to overcome these limitations using word embeddings pretrained on a large-scale dataset. They introduce a graph model to represent lexical relations between sentences, which encodes sentences as nodes and lexical relations between sentences as edges. This graph-based model captures k-node subgraphs of this graph and represents coherence patterns by the frequency of subgraphs. However, their model neglects context to capture lexical relations.

Modeling lexical coherence has proven to be effective in diverse NLP applications like summarization (Erkan and Radev, 2004), translation (Xiong et al., 2013), and discourse parsing (Jia et al., 2018). We believe that our study for lexical coherence can be beneficial in these applications.

3 Our Model

Figure 1 shows our model architecture. Our model takes sentence representations using a pretrained language model. The model then feeds sentences into the lexical coherence module to produce the semantic centroid vector and the semantic similarity vector. We concatenate the two vectors to generate a model output through a feed-forward network.

Sentence representations: We first encode input sentences using a pretrained language model to produce word representations. We take a sentence representation as the average of all word representations in a sentence. We then feed the sentence representations to the lexical coherence module.

Incremental processing module: Algorithm 1 describes our lexical coherence module. To interpret the sentence being read, we update two components: a semantic centroid vector and a semantic similarity vector. The semantic centroid vector takes averaged representations of preceding sentences, and this vector represents their central point. We then measure the semantic similarity between the current sentence representation and the centroid vector. We use cosine similarity to measure semantic similarity. We iterate this procedure for all sentences.

[1]Our code is available at: https://github.com/sdeva14/coling20-inc-lexi-cohe

Model	Yahoo	Clinton	Enron	Yelp	Avg Acc
Barzilay and Lapata (2008)	38.0	43.0	46.0	45.5	43.1
Guinaudeau and Strube (2013)	40.0	56.0	43.5	53.0	48.1
Li and Jurafsky (2017)	53.5	61.0	54.4	49.1	51.7
∗Mesgar and Strube (2018)	47.3	57.7	50.6	54.6	52.6
Lai and Tetreault (2018)	54.9	60.2	53.2	54.4	55.7
Avg-XLNet-Sent	58.0	57.6	54.3	55.9	56.4
Avg-XLNet-Doc	**60.5**	**65.9**	**56.9**	**59.0**	**60.6**
Our Model-Sent	57.3	61.7	54.5	56.9	57.6

Table 1: GCDC Accuracy performance comparison (∗: our re-implementation).

A convolutional layer is applied to the semantic similarity vector to extract a feature map which represents the patterns of changes in semantic similarities. Max-pooling is applied to the feature map, and this lets the model capture features semantically relevant to the centroid vector.

Document representation: We concatenate the semantic centroid vector, updated on the last sentence, and the semantic similarity vector. Finally, a feed-forward network is applied on the representation to produce the output value.

4 Experiments

4.1 Implementation Details

We implement our model using the PyTorch library and use the Stanford Stanza library[2] for sentence tokenization. For the baselines that do not use the pretrained language model, we use Glove for word embeddings, the pretrained word embeddings trained on Google News (Pennington et al., 2014). For our model, we apply a convolutional layer whose kernel size is 3, stride is 2, and padding is 2 and an adaptive max-pooling layer reducing a vector to the length of 5 (see the supplementary material for more details).

Many pretrained language models cannot encode long texts due to their training settings, or require a massive amount of memory to encode them. In this work, we employ XLNet for the pretrained-language model (Yang et al., 2019). Unlike BERT (Devlin et al., 2019), since XLNet can handle any input sequence length, which is required for our datasets to encode a whole text at once.

We report the results by the mean of 10 cross-validation runs with different random seeds. We validate statistical significance with a one-sample t-test with p-value < 0.01. We use 23GB GPU memory of a NVidia P40 for each run.

4.2 Simple Baselines relying on a pretrained language model

To investigate the influence of a pretrained language model on the tasks, we present two simple baselines relying on the pretrained language model. The first model encodes an input document at the sentence level and averages the encoded representations (Averaged-XLNet-Sent). The second model has the same architecture but it encodes an input document at the document level at once (Averaged-XLNet-Doc). We compare these baselines with other models for both tasks.

4.3 Task 1: Assessing Discourse Coherence

Dataset: We first evaluate our model on the Grammarly Corpus of Discourse Coherence (GCDC) dataset (Lai and Tetreault, 2018). While previous work evaluates coherence models on formal texts (Barzilay and Lapata, 2008), GCDC is designed to evaluate coherence models on informal texts, such as emails or online reviews. The dataset contains four domains: Clinton and Enron for emails, Yahoo for questions and answers in an online forum, and Yelp for online reviews of businesses. The quality of the dataset is controlled to have evenly-distributed scores and a low correlation between discourse length and scores[3].

[2]https://stanfordnlp.github.io/stanza/

[3]The Pearson correlation between text length and scores is lower than 0.12 in all domains.

Model	Prompt								Avg Acc
	1	2	3	4	5	6	7	8	
∗Dong et al. (2017)	69.3	66.5	65.8	66.4	68.9	64.2	67.1	65.7	66.7
∗Mesgar and Strube (2018)	54.9	56.4	52.4	56.1	55.3	55.5	56.0	57.3	55.5
∗Nadeem et al. (2019)	58.9	55.8	65.6	61.3	57.8	57.5	52.4	52.8	57.8
Avg-XLNet-Sent	70.7	69.5	69.0	67.5	72.4	70.9	70.1	69.0	69.9
Avg-XLNet-Doc	74.7	74.4	73.0	73.8	75.6	75.7	71.9	71.0	73.8
Our Model-Sent	**75.6**	**73.4**	**75.0**	**73.5**	**76.8**	**75.2**	**73.5**	**72.8**	**74.5**

Table 2: TOEFL Accuracy performance comparison (∗: our re-implementation).

Experimental setup: For GCDC, we perform the experiments following previous work (Lai and Tetreault, 2018). We perform 10-fold cross-validation, use the same evaluation measure, accuracy for 3-class classification, and use the same loss function, cross-entropy loss.

Baseline models: Barzilay and Lapata (2008) propose the entity grid, based on Centering Theory (Grosz et al., 1995). This model considers the distribution of entities over sentences. Guinaudeau and Strube (2013) convert the supervised entity grid into an unsupervised graph-based model. Li and Jurafsky (2017) propose a neural model which uses cliques, sets of adjacent sentences, to discriminate the difference of sentences extracted from original articles and randomly permutated ones. Mesgar and Strube (2018) propose a neural coherence model which finds the two most similar RNN outputs to determine the most salient part of sentences to connect adjacent sentences. Lai and Tetreault (2018) show that a simple neural model which uses paragraph information outperforms previous coherence models on GCDC.

Results: Table 1 shows the performance of coherence models on GCDC. The first baseline outperforms the previous models. Our model, which encodes sentences individually using the pretrained language model and interprets sentences incrementally, outperforms the first baseline.

However, the second baseline, which –unlike humans– encodes a whole text at once, outperforms our model. We suspect that the characteristics of GCDC lead to this. Lai and Tetreault (2018) observe that many texts with low coherence are not well-organized and have unexpected topic switching more than others. The texts on GCDC mostly consist of several sentences, and the model might distinguish these cases well on relatively short sequences. To investigate this further, we next compare models on TOEFL where texts are written in an academic style.

4.4 Task 2: Automated Essay Scoring

Dataset: To examine the effectiveness of our model in a downstream task with formal texts, we evaluate our model on the Test of English as a Foreign Language dataset (TOEFL) dataset. TOEFL has an overall higher quality of essays compared to essays in a standard dataset for AES, the Automated Student Assessment Prize (ASAP) dataset[4]. The prompts in ASAP are written by students in grade levels 7 to 10 of US middle schools, whereas the prompts in TOEFL are submitted for the standard English test for the entrance to US universities by non-native students. The prompts in TOEFL do not vary so much, the student population is more controlled, and the essays have a similar length.

Experimental setup: We evaluate performance in-domain at the prompt level. We perform 5-fold cross-validation. For 3-class classification, we use cross-entropy loss to train models and measure accuracy to evaluate models. We evaluate performance for 30 epochs on the validation set. Following previous work on AES (Taghipour and Ng, 2016), the model which reaches the best performance on the validation set is then applied to the test set (see the supplementary material for details).

Baseline models: Dong et al. (2017) introduce a model which consists of a convolutional layer, followed by a recurrent layer, and an attention layer (Bahdanau et al., 2015). We also compare with the state of the art on TOEFL, Nadeem et al. (2019). Inspired by Dong et al. (2017), Nadeem et al. (2019) propose

[4]https://kaggle.com/c/asap-aes/

a model which uses an attention layer to decide the relative weights automatically in adjacent words as well as sentences. However, we notice that Nadeem et al. (2019) evaluate their model in a different experimental setup. They filter out content with sentences longer than 40 words or documents longer than 25 sentences; they also evaluate performance without cross-validation[5]. To ensure a fair comparison, we changed the experimental setup in their implementation. Mesgar and Strube (2018) evaluate their coherence model on the AES task as well as the task of assessing readability.

Results: Table 2 summarizes the performance of models on TOEFL. The first baseline outperforms the previous models, and the second baseline shows better performance than them. Our model sets a new state of the art at this dataset. Texts included in TOEFL are organized better than those in GCDC where the second baseline outperforms our model. We suspect that the pretrained language model captures some patterns on long sequences to predict scores, rather than capturing relations between sentences. This suggests that our model benefits more from incremental language processing on long sequences.

5 Conclusions

We propose a coherence model which encodes sentences individually using a pretrained language model and interprets sentences incrementally. The simple baseline, which encodes a whole text at once unlike humans do, outperforms our model on GCDC which includes informal texts such as online reviews. However, our model outperforms this model on TOEFL whose texts are organized better. Our findings suggest that it could be useful to constrain models to be exposed limited information as humans do to design more advanced neural models with a pretrained language model.

Acknowledgments

The authors would like to thank the anonymous reviewers for their comments. This work has been funded by the Klaus Tschira Foundation, Heidelberg, Germany. The first author has been supported by a Heidelberg Institute for Theoretical Studies Ph.D. scholarship.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *In Proceedings of the ICLR Conference*.

Regina Barzilay and Mirella Lapata. 2008. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Fei Dong, Yue Zhang, and Jie Yang. 2017. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 153–162, Vancouver, Canada, August. Association for Computational Linguistics.

Jacob Eisenstein and Regina Barzilay. 2008. Bayesian unsupervised topic segmentation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 334–343, Honolulu, Hawaii, October. Association for Computational Linguistics.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479.

Edward Gibson and Tessa Warren. 2004. Reading-time evidence for intermediate linguistic structure in long-distance dependencies. *Syntax*, 7(1):55–78.

Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225.

[5]We confirm this by examining their implementation and emailing the first author.

Camille Guinaudeau and Michael Strube. 2013. Graph-based local coherence modeling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 93–103, Sofia, Bulgaria, August. Association for Computational Linguistics.

Aditya Gupta and Greg Durrett. 2019. Effective use of transformer networks for entity tracking. 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 759–769, Hong Kong, China, November. Association for Computational Linguistics.

M. A. K. Halliday and Ruqaiya Hasan. 1976. *Cohesion in English*. London, U.K.: Longman.

Yanyan Jia, Yuan Ye, Yansong Feng, Yuxuan Lai, Rui Yan, and Dongyan Zhao. 2018. Modeling discourse cohesion for discourse parsing via memory network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 438–443, Melbourne, Australia, July. Association for Computational Linguistics.

Yuki Kamide, Gerry TM Altmann, and Sarah L Haywood. 2003. The time-course of prediction in incremental sentence processing: Evidence from anticipatory eye movements. *Journal of Memory and language*, 49(1):133–156.

Arne Köhn. 2018. Incremental natural language processing: Challenges, strategies, and evaluation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2990–3003, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Alice Lai and Joel Tetreault. 2018. Discourse coherence in the wild: A dataset, evaluation and methods. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 214–223, Melbourne, Australia, July. Association for Computational Linguistics.

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, September. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Hierarchical transformers for multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5070–5081, Florence, Italy, July. Association for Computational Linguistics.

William D Marslen-Wilson. 1975. Sentence perception as an interactive parallel process. *Science*, 189(4198):226–228.

Danielle S McNamara, Max M Louwerse, Philip M McCarthy, and Arthur C Graesser. 2010. Coh-metrix: Capturing linguistic features of cohesion. *Discourse Processes*, 47(4):292–330.

Mohsen Mesgar and Michael Strube. 2016. Lexical coherence graph modeling using word embeddings. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1414–1423, San Diego, California, June. Association for Computational Linguistics.

Mohsen Mesgar and Michael Strube. 2018. A neural local coherence model for text quality assessment. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4328–4339, Brussels, Belgium, October-November. Association for Computational Linguistics.

Jane Morris and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 17(1):21–48.

Farah Nadeem, Huy Nguyen, Yang Liu, and Mari Ostendorf. 2019. Automated essay scoring with discourse-aware neural models. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 484–493, Florence, Italy, August. Association for Computational Linguistics.

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, October. Association for Computational Linguistics.

Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1882–1891, Austin, Texas, November. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Deyi Xiong, Yang Ding, Min Zhang, and Chew Lim Tan. 2013. Lexical chain based cohesion models for document-level statistical machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1563–1573, Seattle, Washington, USA, October. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R. Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, pages 5754–5764.

6 Appendix A. Dataset Details

Table 3 describes statistics on two datasets, GCDC[6] and TOEFL[7]. We split a text at the sentence level by Stanford Stanza library, and tokenize them by the XLNet tokenizer. Table 4 describes the topic of each prompt in TOEFL. They are all open-ended tasks, that do not have given context but require students to submit their opinion.

Dataset	#Texts	Avg len (Std)	Max len	Scores
G-Y	1,200	173 (48)	378	1-3
G-C	1,200	200 (65)	385	1-3
G-E	1,200	203 (67)	388	1-3
G-P	1,200	198 (58)	374	1-3
T-P1	1,656	401 (97)	902	1-3
T-P2	1,562	423 (97)	902	1-3
T-P3	1,396	407 (102)	837	1-3
T-P4	1,509	405 (99)	852	1-3
T-P5	1,648	424 (101)	993	1-3
T-P6	960	425 (101)	925	1-3
T-P7	1,686	396 (87)	755	1-3
T-P8	1,683	407 (92)	795	1-3

Table 3: Dataset statistics on tokenization: four domains in GCDC, Yahoo (G-Y), Clinton (G-C), Enron (G-E), Yelp (G-P), and each TOEFL prompt (T-P).

Prompt 1	Agree or Disagree: It is better to have broad knowledge of many academic subjects than to specialize in one specific subject.
Prompt 2	Agree or Disagree: Young people enjoy life more than older people do.
Prompt 3	Agree or Disagree: Young people nowadays do not give enough time to helping their communities.
Prompt 4	Agree or Disagree: Most advertisements make products seem much better than they really are.
Prompt 5	Agree or Disagree: In twenty years, there will be fewer cars in use than there are today.
Prompt 6	Agree or Disagree: The best way to travel is in a group led by a tour guide.
Prompt 7	Agree or Disagree: It is more important for students to understand ideas and concepts than it is for them to learn facts.
Prompt 8	Agree or Disagree: Successful people try new things and take risks rather than only doing what they already know how to do well.

Table 4: Topic description: TOEFL.

[6]https://github.com/aylai/GCDC-corpus
[7]https://catalog.ldc.upenn.edu/LDC2014T06