Encouraging Paragraph Embeddings to Remember Sentence Identity Improves Classification

Tu Vu and **Mohit Iyyer**
College of Information and Computer Sciences
University of Massachusetts Amherst
{tuvu,miyyer}@cs.umass.edu

Abstract

While paragraph embedding models are remarkably effective for downstream classification tasks, what they learn and encode into a single vector remains opaque. In this paper, we investigate a state-of-the-art paragraph embedding method proposed by Zhang et al. (2017) and discover that it cannot reliably tell whether a given sentence occurs in the input paragraph or not. We formulate a *sentence content* task to probe for this basic linguistic property and find that even a much simpler bag-of-words method has no trouble solving it. This result motivates us to replace the reconstruction-based objective of Zhang et al. (2017) with our sentence content probe objective in a semi-supervised setting. Despite its simplicity, our objective improves over paragraph reconstruction in terms of (1) downstream classification accuracies on benchmark datasets, (2) faster training, and (3) better generalization ability.[1]

1 Introduction

Methods that embed a paragraph into a single vector have been successfully integrated into many NLP applications, including text classification (Zhang et al., 2017), document retrieval (Le and Mikolov, 2014), and semantic similarity and relatedness (Dai et al., 2015; Chen, 2017). However, downstream performance provides little insight into the kinds of linguistic properties that are encoded by these embeddings. Inspired by the growing body of work on sentence-level linguistic *probe tasks* (Adi et al., 2017; Conneau et al., 2018), we set out to evaluate a state-of-the-art paragraph embedding method using a probe task to measure how well it encodes the identity of the sentences within a paragraph. We discover that the method falls short of capturing this basic property, and that implementing a simple objective to fix this issue improves classification performance, training speed, and generalization ability.

We specifically investigate the paragraph embedding method of Zhang et al. (2017), which consists of a CNN-based encoder-decoder model (Sutskever et al., 2014) paired with a reconstruction objective to learn powerful paragraph embeddings that are capable of accurately reconstructing long paragraphs. This model significantly improves downstream classification accuracies, outperforming LSTM-based alternatives (Li et al., 2015).

How well do these embeddings encode *whether or not a given sentence appears in the paragraph*? Conneau et al. (2018) show that such identity information is correlated with performance on downstream sentence-level tasks. We thus design a probe task to measure the extent to which this *sentence content* property is captured in a paragraph embedding. Surprisingly, our experiments (Section 2) reveal that despite its impressive downstream performance, the model of Zhang et al. (2017) substantially underperforms a simple bag-of-words model on our sentence content probe.

Given this result, it is natural to wonder whether the sentence content property is actually useful for downstream classification. To explore this question, we move to a semi-supervised setting by pre-training the paragraph encoder in Zhang et al.'s model (2017) on either our sentence content objective or its original reconstruction objective, and then optionally fine-tuning it on supervised classification tasks (Section 3). Sentence content significantly improves over reconstruction on standard benchmark datasets both with and without fine-tuning; additionally, this objective is four times faster to train than the reconstruction-based variant. Furthermore, pre-training with sentence content substantially boosts generalization ability: fine-tuning a pre-trained model on just 500 labeled

[1] Source code and data are available at https://github.com/tuvuumass/SCoPE.

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 6331–6338
Florence, Italy, July 28 - August 2, 2019.

reviews from the Yelp sentiment dataset surpasses the accuracy of a purely supervised model trained on 100,000 labeled reviews.

Our results indicate that incorporating probe objectives into downstream models might help improve both accuracy and efficiency, which we hope will spur more linguistically-informed research into paragraph embedding methods.

2 Probing paragraph embeddings for sentence content

In this section, we first fully specify our probe task before comparing the model of Zhang et al. (2017) to a simple bag-of-words model. Somewhat surprisingly, the latter substantially outperforms the former despite its relative simplicity.

2.1 Probe task design

Our proposed *sentence content* task is a paragraph-level analogue to the word content task of Adi et al. (2017): given embeddings[2] $\boldsymbol{p}, \boldsymbol{s}$ of a paragraph p and a candidate sentence s, respectively, we train a classifier to predict whether or not s occurs in p. Specifically, we construct a binary classification task in which the input is $[\boldsymbol{p}; \boldsymbol{s}]$, the concatenation of $\boldsymbol{p}$ and $\boldsymbol{s}$. This task is balanced: for each paragraph p in our corpus, we create one positive instance by sampling a sentence s^+ from p and one negative instance by randomly sampling a sentence s^- from another paragraph p'. As we do not perform any fine-tuning of the base embedding model, our methodology is agnostic to the choice of model.

2.2 Paragraph embedding models

Armed with our probe task, we investigate the following embedding methods.[3]

Zhang et al. (2017) (CNN-R): This model uses a multi-layer convolutional encoder to compute a single vector embedding $\boldsymbol{p}$ of an input paragraph p and a multi-layer deconvolutional decoder that mirrors the convolutional steps in the encoding stage to reconstruct the tokens of p from $\boldsymbol{p}$. We refer readers to Zhang et al. (2017) for a detailed description of the model architecture. For a more intuitive comparison in our experiments, we denote this model further as CNN-R instead of CNN-DCNN as in the original paper. In all experiments, we use their publicly available code.[4]

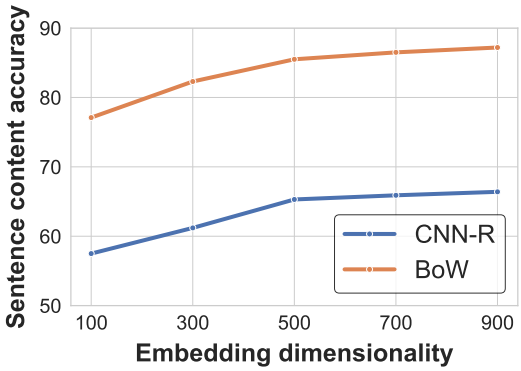

Figure 1: Probe task accuracies across representation dimensions. BoW surprisingly outperforms the more complex model CNN-R.

Bag-of-words (BoW): This model is simply an average of the word vectors learned by a trained CNN-R model. BoW models have been shown to be surprisingly good at sentence-level probe tasks (Adi et al., 2017; Conneau et al., 2018).

2.3 Probe experimental details

Paragraphs to train our classifiers are extracted from the Hotel Reviews corpus (Li et al., 2015), which has previously been used for evaluating the quality of paragraph embeddings (Li et al., 2015; Zhang et al., 2017). We only consider paragraphs that have at least two sentences. Our dataset has 346,033 training paragraphs, 19,368 for validation, and 19,350 for testing. The average numbers of sentences per paragraph, tokens per paragraph, and tokens per sentence are 8.0, 123.9, and 15.6, respectively. The vocabulary contains 25,000 tokens. To examine the effect of the embedding dimensionality d on the results, we trained models with $d \in \{100, 300, 500, 700, 900\}$.

Each classifier is a feed-forward neural network with a single 300-d ReLu layer. We use a mini-batch size of 32, Adam optimizer (Kingma and Ba, 2015) with a learning rate of $2e$-4, and a dropout rate of 0.5 (Srivastava et al., 2014). We trained classifiers for a maximum of 100 epochs with early stopping based on validation performance.

2.4 BoW outperforms CNN-R on sentence content

Our probe task results are displayed in Figure 1. Interestingly, BoW performs significantly better

[2] computed using the same embedding method

[3] We experiment with several other models in Appendix A.1, including an LSTM-based encoder-decoder model, a variant of Paragraph Vector (Le and Mikolov, 2014), and BOW models using pre-trained word representations.

[4] https://github.com/dreasysnail/textCNN_public

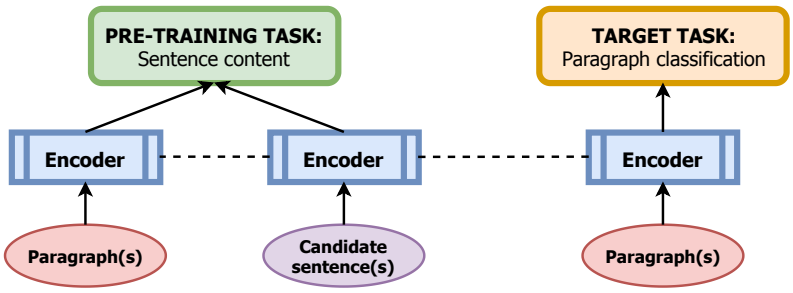

Figure 2: A visualization of our semi-supervised approach. We first train the CNN encoder (shown as two copies with shared parameters) on unlabeled data using our sentence content objective. The encoder is then used for downstream classification tasks.

Setting	CNN-R	BoW
Without s^+ excluded from p	61.2	**82.3**
With s^+ excluded from p	57.5	**61.7**

Table 1: Probe task accuracies without and with s^+ excluded from p, measured at $d = 300$. BoW's accuracy degrades quickly in the latter case, suggesting that it relies much more on low-level matching.

Dataset	Type	# classes	# examples
Yelp	Sentiment	2	560K
DBpedia	Topic	14	560K
Yahoo	Topic	10	1.4M

Table 2: Properties of the text classification datasets used for our evaluations.

than CNN-R, achieving an accuracy of 87.2% at 900 dimensions, compared to only 66.4% for CNN-R. We hypothesize that much of BoW's success is because it is easier for the model to perform approximate string matches between the candidate sentence and text segments within the paragraph than it is for the highly non-linear representations of CNN-R.

To investigate this further, we repeat the experiment, but exclude the sentence s^+ from the paragraph p during both training and testing. As we would expect (see Table 1), BoW's performance degrades significantly (20.6% absolute) with s^+ excluded from p, whereas CNN-R experiences a more modest drop (3.6%). While BoW still outperforms CNN-R in this new setting, the dramatic drop in accuracy suggests that it relies much more heavily on low-level matching.

3 Sentence content improves paragraph classification

Motivated by our probe results, we further investigate whether incorporating the sentence content property into a paragraph encoder can help increase downstream classification accuracies. We propose a semi-supervised approach by pre-training the encoder of CNN-R using our sentence content objective, and optionally fine-tuning it on different classification tasks. A visualization of this procedure can be seen in Figure 2. We compare our approach (henceforth **CNN-SC**) without and with fine-tuning against CNN-R, which uses a reconstruction-based objective.[5] We report comparisons on three standard paragraph classification datasets: Yelp Review Polarity (Yelp), DBPedia, and Yahoo! Answers (Yahoo) (Zhang et al., 2015), which are instances of common classification tasks, including sentiment analysis and topic classification. Table 2 shows the statistics for each dataset. Paragraphs from each training set without labels were used to generate training data for unsupervised pre-training.

Sentence content significantly improves over reconstruction on both in-domain and out-of-domain data We first investigate how useful each pre-training objective is for downstream classification without any fine-tuning by simply training a classifier on top of the frozen pre-trained CNN encoder. We report the best downstream performance for each model across different numbers of pre-training epochs. The first row of Table 3 shows the downstream accuracy on Yelp when the whole unlabeled data of the Yelp training set is used for unsupervised pre-training. Strikingly,

[5]Here, we use unsupervised pre-training as it allows us to isolate the effects of the unsupervised training objectives. Zhang et al. (2017) implemented auxiliary unsupervised training as an alternative form of semi-supervised learning. We tried both strategies and found that they performed similarly.

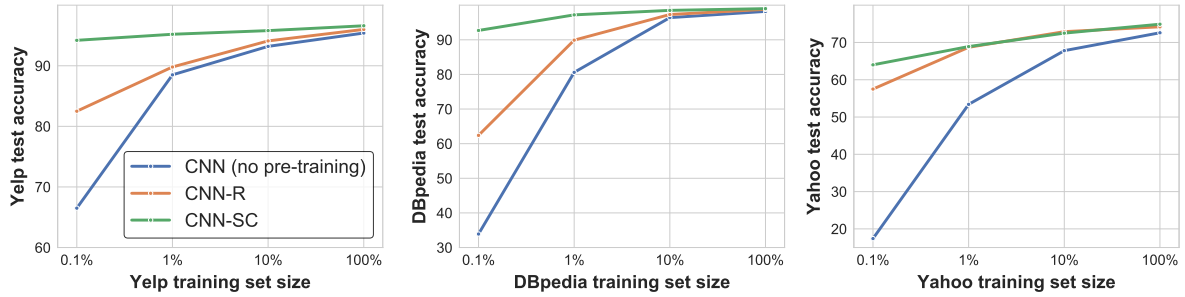

Figure 3: CNN-SC substantially improves generalization ability. Results of CNN-R are taken from Zhang et al. (2017).

Pre-training	CNN-R	CNN-SC
On Yelp	67.4	**90.0**
On Wikipedia	61.4	**65.7**
Wall-clock speedup	1x	**4x**

Table 3: Yelp test accuracy (without fine-tuning). CNN-SC significantly improves over CNN-R.

CNN-SC achieves an accuracy of 90.0%, outperforming CNN-R by a large margin. Additionally, sentence content is four times as fast to train as the computationally-expensive reconstruction objective.[6] Are representations obtained using these objectives more useful when learned from in-domain data? To examine the dataset effect, we repeat our experiments using paragraph embeddings pre-trained using these objectives on a subset of Wikipedia (560K paragraphs). The second row of Table 3 shows that both approaches suffer a drop in downstream accuracy when pre-trained on out-of-domain data. Interestingly, CNN-SC still performs best, indicating that sentence content is more suitable for downstream classification.

Another advantage of our sentence content objective over reconstruction is that it better correlates to downstream accuracy (see Appendix A.2). For reconstruction, there is no apparent correlation between BLEU and downstream accuracy; while BLEU increases with the number of epochs, the downstream performance quickly begins to decrease. This result indicates that early stopping based on BLEU is not feasible with reconstruction-based pre-training objectives.

With fine-tuning, CNN-SC substantially boosts accuracy and generalization We switch gears now to our fine-tuning experiments. Specifically, we take the CNN encoder pre-trained using our sentence content objective and then fine-tune it on downstream classification tasks with supervised labels. While our previous version of CNN-SC created just a single positive/negative pair of examples from a single paragraph, for our fine-tuning experiments we create a pair of examples from *every sentence* in the paragraph to maximize the training data. For each task, we compare against the original CNN-R model in (Zhang et al., 2017). Figure 3 shows the model performance with fine-tuning on 0.1% to 100% of the training set of each dataset. One interesting result is that CNN-SC relies on very few training examples to achieve comparable accuracy to the purely supervised CNN model. For instance, fine-tuning CNN-SC using just 500 labeled training examples surpasses the accuracy of training from scratch on 100,000 labeled examples, indicating that the sentence content encoder generalizes well. CNN-SC also outperforms CNN-R by large margins when only small amounts of labeled training data are

Model	Yelp	DBPedia	Yahoo
purely supervised w/o external data			
ngrams TFIDF	95.4	98.7	68.5
Large Word ConvNet	95.1	98.3	70.9
Small Word ConvNet	94.5	98.2	70.0
Large Char ConvNet	94.1	98.3	70.5
Small Char ConvNet	93.5	98.0	70.2
SA-LSTM (word level)	NA	98.6	NA
Deep ConvNet	95.7	98.7	73.4
CNN (Zhang et al., 2017)	95.4	98.2	72.6
pre-training + fine-tuning w/o external data			
CNN-R (Zhang et al., 2017)	96.0	98.8	74.2
CNN-SC (ours)	**96.6**	**99.0**	**74.9**
pre-training + fine-tuning w/ external data			
ULMFiT (Howard and Ruder, 2018)	97.8	99.2	NA

Table 4: CNN-SC outperforms other baseline models that do not use external data, including CNN-R. All baseline models are taken from Zhang et al. (2017).

[6]This objective requires computing a probability distribution over the whole vocabulary for every token of the paragraph, making it prohibitively slow to train.

available. Finally, when all labeled training data is used, CNN-SC achieves higher classification accuracy than CNN-R on all three datasets (Table 4).

While CNN-SC exhibits a clear preference for target task unlabeled data (see Table 3), we can additionally leverage large amounts of unlabeled general-domain data by incorporating pre-trained word representations from language models into CNN-SC. Our results show that further improvements can be achieved by training the sentence content objective on top of the pre-trained language model representations from ULMFiT (Howard and Ruder, 2018) (see Appendix A.3), indicating that our sentence content objective learns complementary information. On Yelp, it exceeds the performance of training from scratch on the whole labeled data (560K examples) with only 0.1% of the labeled data.

CNN-SC implicitly learns to distinguish between class labels The substantial difference in downstream accuracy between pre-training on in-domain and out-of-domain data (Table 3) implies that the sentence content objective is implicitly learning to distinguish between class labels (e.g., that a candidate sentence with negative sentiment is unlikely to belong to a paragraph with positive sentiment). If true, this result implies that CNN-SC prefers not only in-domain data but also a representative sample of paragraphs from all class labels. To investigate, we conduct an additional experiment that restricts the class label from which negative sentence candidates s^- are sampled. We experiment with two sources of s^-: (1) paragraphs of the same class label as the probe paragraph (CNN-SC$^-$), and (2) paragraphs from a different class label (CNN-SC$^+$). Figure 4 reveals that the performance of CNN-SC drops dramatically when trained on the first dataset and improves when trained on the second dataset, which confirms our hypothesis.

4 Related work

Text embeddings and probe tasks A variety of methods exist for obtaining fixed-length dense vector representations of words (e.g., Mikolov et al., 2013; Pennington et al., 2014; Peters et al., 2018), sentences (e.g., Kiros et al., 2015; Conneau et al., 2017; Subramanian et al., 2018; Cer et al., 2018), and larger bodies of text (e.g., Le and Mikolov, 2014; Dai et al., 2015; Iyyer et al., 2015; Li et al., 2015; Chen, 2017; Zhang et al., 2017) that

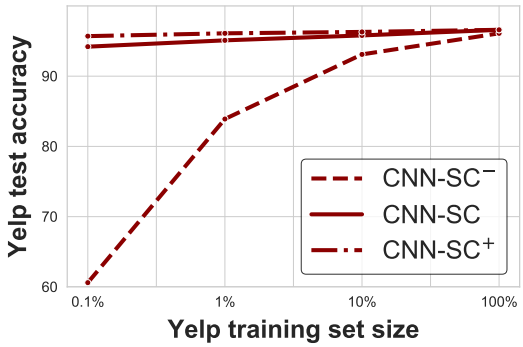

Figure 4: CNN-SC implicitly learns to distinguish between class labels.

significantly improve various downstream tasks. To analyze word and sentence embeddings, recent work has studied classification tasks that probe them for various linguistic properties (Shi et al., 2016; Adi et al., 2017; Belinkov et al., 2017a,b; Conneau et al., 2018; Tenney et al., 2019). In this paper, we extend the notion of probe tasks to the paragraph level.

Transfer learning Another line of related work is transfer learning, which has been the driver of recent successes in NLP. Recently-proposed objectives for transfer learning include surrounding sentence prediction (Kiros et al., 2015), paraphrasing (Wieting and Gimpel, 2017), entailment (Conneau et al., 2017), machine translation (McCann et al., 2017), discourse (Jernite et al., 2017; Nie et al., 2017), and language modeling (Peters et al., 2018; Radford et al., 2018; Devlin et al., 2018).

5 Conclusions and Future work

In this paper, we evaluate a state-of-the-art paragraph embedding model, based on how well it captures the sentence identity within a paragraph. Our results indicate that the model is not fully aware of this basic property, and that implementing a simple objective to fix this issue improves classification performance, training speed, and generalization ability. Future work can investigate other embedding methods with a richer set of probe tasks, or explore a wider range of downstream tasks.

Acknowledgments

We thank the anonymous reviewers, Kalpesh Krishna, Nader Akoury, and the members of the UMass NLP reading group for their helpful comments.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *International Conference on Learning Representations (ICLR)*.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017a. What do neural machine translation models learn about morphology? In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 861–872.

Yonatan Belinkov, Lluís Màrquez, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2017b. Evaluating layers of representation in neural machine translation on part-of-speech and semantic tagging tasks. In *Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP)*, pages 1–10.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder. *CoRR*, abs/1803.11175.

Minmin Chen. 2017. Efficient vector representation for documents through corruption. In *International Conference on Learning Representations (ICLR)*.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 670–680.

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2126–2136.

Andrew M. Dai, Christopher Olah, and Quoc V. Le. 2015. Document embedding with paragraph vectors. *CoRR*, abs/1507.07998.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

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

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 328–339.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 1681–1691.

Yacine Jernite, Samuel R. Bowman, and David Sontag. 2017. Discourse-based objectives for fast unsupervised sentence representation learning. *CoRR*, abs/1705.00557.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3294–3302.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 32, pages 1188–1196.

Jiwei Li, Thang Luong, and Dan Jurafsky. 2015. A hierarchical neural autoencoder for paragraphs and documents. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 1106–1115.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL)*, pages 142–150.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems (NIPS)*, pages 6294–6305.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3111–3119.

Allen Nie, Erin D. Bennett, and Noah D. Goodman. 2017. Dissent: Sentence representation learning from explicit discourse relations. *CoRR*, abs/1710.04334.

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

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 2227–2237.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Xing Shi, Inkit Padhi, and Kevin Knight. 2016. Does string-based neural mt learn source syntax? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1526–1534.

Nitish 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*, 15:1929–1958.

Sandeep Subramanian, Adam Trischler, Yoshua Bengio, and Christopher J Pal. 2018. Learning general purpose distributed sentence representations via large scale multi-task learning. In *International Conference on Learning Representations (ICLR)*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3104–3112.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations (ICLR)*.

John Wieting and Kevin Gimpel. 2017. Revisiting recurrent networks for paraphrastic sentence embeddings. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2078–2088.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems (NIPS)*, pages 649–657.

Yizhe Zhang, Dinghan Shen, Guoyin Wang, Zhe Gan, Ricardo Henao, and Lawrence Carin. 2017. Deconvolutional paragraph representation learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 4169–4179.

A Appendices

A.1 BoW models outperform more complex models on our sentence content probe

In addition to the paragraph embedding models presented in the main paper, we also experiment with the following embedding methods:

Model	Dimensionality	Accuracy
Random	–	50.0
trained on paragraphs from Hotel Reviews		
CNN-R	900	66.4
BoW (CNN-R)	900	87.2
LSTM-R	900	65.4
Doc2VecC	900	90.8
pre-trained on other datasets		
Word2Vec-avg	300	83.2
GloVe-avg	300	84.6
ELMo-avg	1024	88.1

Table 5: Sentence content accuracy for different paragraph embedding methods. BoW models outperform more complex models.

LSTM-R: We consider an LSTM (Hochreiter and Schmidhuber, 1997) encoder-decoder model paired with a reconstruction objective. Specifically, we implement a single-layer bidirectional LSTM encoder and a two-layer unidirectional LSTM decoder. Paragraph representations are computed from the encoder's final hidden state.

Doc2VecC: This model (Chen, 2017) represents a document as an average of randomly-sampled words from within the document. The method introduces a corruption mechanism that favors rare but important words while suppressing frequent but uninformative ones. Doc2VecC was found to outperform other unsupervised BoW-style algorithms, including Paragraph Vector (Le and Mikolov, 2014), on downstream tasks.

Other BoW models: We also consider other BoW models with pre-trained word embeddings or contextualized word representations, including Word2Vec (Mikolov et al., 2013), Glove (Pennington et al., 2014), and ELMo (Peters et al., 2018). Paragraph embeddings are computed as the average of the word vectors. For ELMo, we take the average of the layers.

The results of our sentence content probe task are summarized in Table 5.

A.2 Sentence content better correlates to downstream accuracy than reconstruction

See Figure 5.

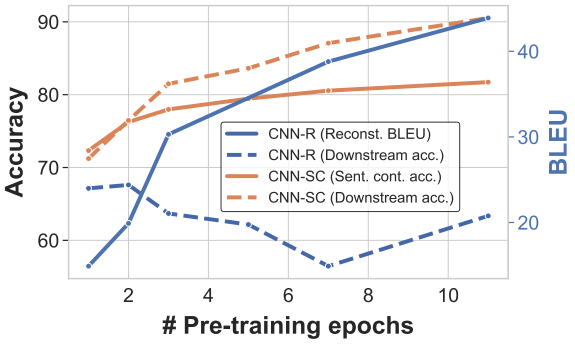

Figure 5: Pre-training performance vs. downstream accuracy on Yelp. Performance measured on validation data. There is no apparent correlation between BLEU and downstream accuracy.

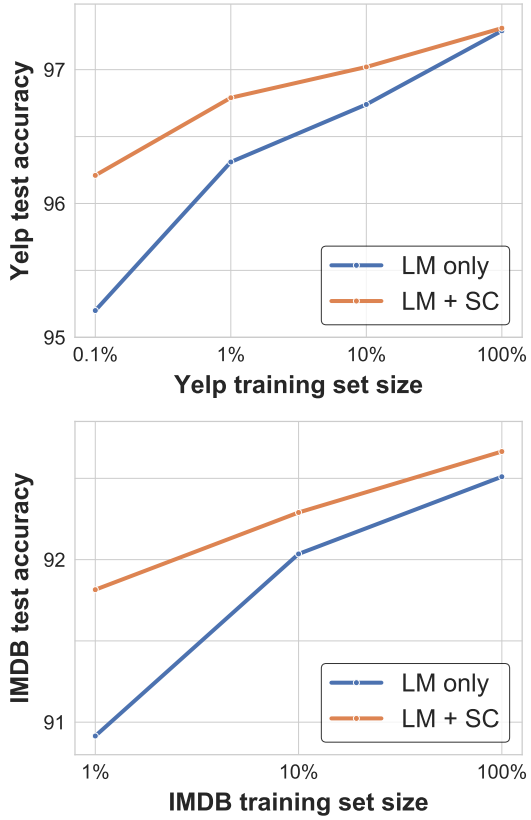

Figure 6: Further improvements can be achieved by training sentence content (SC) on top of the pre-trained language model (LM) representations from ULMFiT (Howard and Ruder, 2018).

A.3 Further improvements by training sentence content on top of pre-trained language model representations

Figure 6 shows that further improvements can be achieved by training sentence content on top of the pre-trained language model representations from ULMFiT (Howard and Ruder, 2018) on Yelp and IMDB (Maas et al., 2011) datasets, indicating that our sentence content objective learns complementary information.[7] On Yelp, it exceeds the performance of training from scratch on the whole labeled data (560K examples) with only 0.1% of the labeled data.

[7]Here, we do not perform target task classifier fine-tuning to isolate the effects of our sentence content objective.