Supplementary Material: PARANMT-50M: Pushing the Limits of Paraphrastic Sentence Embeddings with Millions of Machine Translations

John Wieting[1] **Kevin Gimpel**[2]
[1]Carnegie Mellon University, Pittsburgh, PA, 15213, USA
[2]Toyota Technological Institute at Chicago, Chicago, IL, 60637, USA
jwieting@cs.cmu.edu, kgimpel@ttic.edu

1 Paraphrase Lexicon

While PARANMT-50M consists of sentence pairs, we demonstrate how a paraphrase lexicon can be extracted from it. One simple approach is to extract and rank word pairs $\langle u, v\rangle$ using the **cross-sentence pointwise mutual information (PMI)**:

$$\mathrm{PMI}_{\mathrm{cross}}(u, v) = \log \frac{\#(u, v)\#(\cdot, \cdot)}{\#(u)\#(v)}$$

where joint counts $\#(u, v)$ are incremented when u appears in a sentence and v appears in its paraphrase. The marginal counts (e.g., $\#(u)$) are computed based on single-sentence counts, as in ordinary PMI. This works reasonably well but is not able to differentiate words that frequently occur in paraphrase pairs from words that simply occur frequently together in the same sentence. For example, “Hong” and “Kong” have high cross-sentence PMI. We can improve the score by subtracting the ordinary PMI that computes joint counts based on single-sentence co-occurrences. We call the result the **adjusted PMI**:

$$\mathrm{PMI}_{\mathrm{adj}}(u, v) = \mathrm{PMI}_{\mathrm{cross}}(u, v) - \mathrm{PMI}(u, v)$$

Before computing these PMIs from PARANMT-50M, we removed sentence pairs with a paraphrase score less than 0.35 and where either sentence is longer than 30 tokens. When computing the ordinary PMI with single-sentence context, we actually compute separate versions of this PMI score for translations and references in each PARANMT-50M pair, then we average them together. We did this because the two sentences in each pair have highly correlated information, so computing PMI on each half of the data would correspond to capturing natural corpus statistics in a standard application of PMI.

Table 2 shows an evaluation of the resulting score functions on the SimLex-999 word similarity dataset (Hill et al., 2015). As a baseline, we use the lexical portion of PPDB 2.0 (Pavlick et al., 2015), evaluating its ranking score as a similarity score and assigning a similarity of 0 to unseen word pairs.[1] Our adjusted PMI computed from PARANMT-50M is on par with the best PPDB lexicon.

Table 1 shows examples from PPDB and our paraphrase lexicon computed from PARANMT-50M. Paraphrases from PPDB are ordered by the PPDB 2.0 scoring function. Paraphrases from our lexicon are ordered using our adjusted PMI scoring function; we only show paraphrases that appeared at least 10 times in PARANMT-50M.

2 General-Purpose Sentence Embedding Evaluations

We evaluate our sentence embeddings on a range of tasks that have previously been used for evaluating sentence representations (Kiros et al., 2015). These include sentiment analysis (MR, Pang and Lee, 2005; CR, Hu and Liu, 2004; SST, Socher et al., 2013), subjectivity classification (SUBJ; Pang and Lee, 2004), opinion polarity (MPQA; Wiebe et al., 2005), question classification (TREC; Li and Roth, 2002), paraphrase detection (MRPC; Dolan et al., 2004), semantic relatedness (SICK-R; Marelli et al., 2014), and textual entailment (SICK-E). We use the SentEval package from Conneau et al. (2017) to train models on our fixed sentence embeddings for each task.[2]

Table 3 shows results on the general sentence embedding tasks. Each of our individual models produces 300-dimensional sentence embeddings, which is far fewer than the several thousands (often 2400-4800) of dimensions used in most prior

[1] If both orderings for a SimLex word pair appear in PPDB, we average their PPDB 2.0 scores. If multiple lexical entries are found with different POS tags, we take the first instance.

[2] github.com/facebookresearch/SentEval

laughed	PPDB	giggled, smiled, funny, used, grew, bust, ri, did
	ParaNMT-50M	chortled, guffawed, pealed, laughin, laughingstock, cackled, chuckled, snickered, mirthless, chuckling, jeered, laughs, laughing, taunted, burst, cackling, scoffed,...
respectful	PPDB	respect, respected, courteous, disrespectful, friendly, respecting, respectable, humble, environmentally-friendly, child-friendly, dignified, respects, compliant, sensitive,...
	ParaNMT-50M	reverent, deferential, revered, respectfully, awed, respect, respected, respects, respectable, politely, considerate, treat, civil, reverence, polite, keeping, behave, proper, dignified,...

Table 1: Example lexical paraphrases from PPDB ranked using the PPDB 2.0 scoring function and from the paraphrase lexicon we induced from ParaNMT-50M ranked using adjusted PMI.

Dataset	Score	$\rho \times 100$
PPDB L	PPDB 2.0	37.97
PPDB XL	PPDB 2.0	52.32
PPDB XXL	PPDB 2.0	60.44
PPDB XXXL	PPDB 2.0	61.47
ParaNMT-50M	cross-sentence PMI	52.12
ParaNMT-50M	adjusted PMI	**61.59**

Table 2: Spearman's $\rho \times 100$ on SimLex-999 for scored paraphrase lexicons.

work. While using higher dimensionality does not improve correlation on the STS tasks, it does help on the general sentence embedding tasks. Using higher dimensionality leads to more trainable parameters in the subsequent classifiers, increasing their ability to linearly separate the data.

To enlarge the dimensionality, we concatenate the forward and backward states prior to averaging. This is similar to Conneau et al. (2017), though they used max pooling. We experimented with both averaging ("BLSTM (Avg., concatenation)") and max pooling ("BLSTM (Max, concatenation)") using recurrent networks with 2048-dimensional hidden states, so concatenating them yields a 4096-dimension embedding. These high-dimensional models outperform SkipThought (Kiros et al., 2015) on all tasks except SUBJ and TREC. Nonetheless, the InferSent (Conneau et al., 2017) embeddings trained on AllNLI still outperform our embeddings on nearly all of these general-purpose tasks.

We also note that on five tasks (SUBJ, MPQA, SST, TREC, and MRPC), all sentence embedding methods are outperformed by supervised baselines. These baselines use the same amount of supervision as the general sentence embedding methods; the latter actually use far more data overall than the supervised baselines. This suggests that the pretrained sentence representations are not capturing the features learned by the models engineered for those tasks.

We take a closer look of how our embeddings compare to InferSent (Conneau et al., 2017). InferSent is a supervised model trained on a large textual entailment dataset (the SNLI and MultiNLI corpora (Bowman et al., 2015; Williams et al., 2017), which consist of nearly 1 million human-labeled examples).

While InferSent has strong performance across all downstream tasks, our model obtains better results on semantic similarity tasks. It consistently reach correlations approximately 10 points higher than those of InferSent.

Regarding the general-purpose tasks, we note that some result trends appear to be influenced by the domain of the data. InferSent is trained on a dataset of mostly captions, especially the model trained on just SNLI. Therefore, the datasets for the SICK relatedness and entailment evaluations are similar in domain to the training data of InferSent. Further, the training task of natural language inference is aligned to the SICK entailment task. Our results on MRPC and entailment are significantly better than SkipThought, and on a paraphrase task that does not consist of caption data (MRPC), our embeddings are competitive with InferSent. To quantify these domain effects, we performed additional experiments that are described in Section 2.1.

There are many ways to train sentence embeddings, each with its own strengths. InferSent, our models, and the BYTE mLSTM of Radford et al. (2017) each excel in particular classes of downstream tasks. Ours are specialized for semantic similarity. BYTE mLSTM is trained on review data and therefore is best at the MR and CR tasks. Since the InferSent models are trained using entailment supervision and on caption data, they excel on the SICK tasks. Future work will be needed to combine multiple supervision signals to generate embeddings that perform well across all tasks.

2.1 Effect of Training Domain on InferSent

We performed additional experiments to investigate the impact of training domain on down-

Model	Dim.	MR	CR	SUBJ	MPQA	SST	TREC	MRPC	SICK-R	SICK-E
Unsupervised (Unordered Sentences)										
Unigram-TFIDF (Hill et al., 2016)		73.7	79.2	90.3	82.4	-	85.0	73.6/81.7	-	-
SDAE (Hill et al., 2016)	2400	74.6	78.0	90.8	86.9	-	78.4	73.7/80.7	-	-
Unsupervised (Ordered Sentences)										
FastSent (Hill et al., 2016)	100	70.8	78.4	88.7	80.6	-	76.8	72.2/80.3	-	-
FastSent+AE (Hill et al., 2016)		71.8	76.7	88.8	81.5	-	80.4	71.2/79.1	-	-
SkipThought (Kiros et al., 2015)	4800	76.5	80.1	93.6	87.1	82.0	92.2	73.0/82.0	85.8	82.3
Unsupervised (Structured Resources)										
DictRep (Hill et al., 2016)	500	76.7	78.7	90.7	87.2	-	81.0	68.4/76.8	-	-
NMT En-to-Fr (Hill et al., 2016)	2400	64.7	70.1	84.9	81.5	-	82.8	-		
BYTE mLSTM (Radford et al., 2017)	4096	**86.9**	**91.4**	94.6	88.5	-	-	75.0/82.8	79.2	-
Individual Models (Our Work)										
Word	300	75.8	80.5	89.2	87.1	80.0	80.1	68.6/80.9	83.6	80.6
Trigram	300	68.8	75.5	83.6	82.3	73.6	73.0	71.4/82.0	79.3	78.0
LSTM	300	73.8	78.4	88.5	86.5	80.6	76.8	73.6/82.3	83.9	81.9
LSTM	900	75.8	81.7	90.5	87.4	81.6	84.4	74.7/83.0	86.0	83.0
BLSTM	900	75.6	82.4	90.6	87.7	81.3	87.4	75.0/82.9	85.8	84.4
Mixed Models (Our Work)										
Word + Trigram (addition)	300	74.8	78.8	88.5	87.4	78.7	79.0	71.4/81.4	83.2	80.6
Word + Trigram + LSTM (addition)	300	75.0	80.7	88.6	86.6	77.9	78.6	72.7/80.8	83.6	81.8
Word, Trigram (concatenation)	600	75.8	80.5	89.9	87.8	79.7	82.4	70.7/81.7	84.6	82.0
Word, Trigram, LSTM (concatenation)	900	77.6	81.4	91.4	88.2	82.0	85.4	74.0/81.5	85.4	83.8
BLSTM (Avg., concatenation)	4096	77.5	82.6	91.0	89.3	82.8	86.8	75.8/82.6	85.9	83.8
BLSTM (Max, concatenation)	4096	76.6	83.4	90.9	88.5	82.0	87.2	76.6/83.5	85.3	82.5
Supervised (Transfer)										
InferSent (SST) (Conneau et al., 2017)	4096	-	83.7	90.2	89.5	-	86.0	72.7/80.9	86.3	83.1
InferSent (SNLI) (Conneau et al., 2017)	4096	79.9	84.6	92.1	89.8	83.3	88.7	75.1/82.3	**88.5**	**86.3**
InferSent (AllNLI) (Conneau et al., 2017)	4096	81.1	86.3	92.4	90.2	84.6	88.2	76.2/83.1	88.4	**86.3**
Supervised (Direct)										
Naive Bayes - SVM		79.4	81.8	93.2	86.3	83.1	-	-	-	-
AdaSent (Zhao et al., 2015)		83.1	86.3	**95.5**	**93.3**	-	92.4	-	-	-
BLSTM-2DCNN (Zhou et al., 2016)		82.3	-	94.0	-	**89.5**	**96.1**	-	-	-
TF-KLD (Ji and Eisenstein, 2013)		-	-	-	-	-	-	**80.4/85.9**	-	-
Illinois-LH (Lai and Hockenmaier, 2014)		-	-	-	-	-	-	-	-	84.5
Dependency Tree-LSTM (Tai et al., 2015)		-	-	-	-	-	-	-	86.8	-

Table 3: General-purpose sentence embedding tasks, divided into categories based on resource requirements.

stream tasks. We first compare the performance of our "Word, Trigram (concatenation)" model to the InferSent SNLI and AllNLI models on all STS tasks from 2012-2016. We then compare the overall mean with that of the three caption STS datasets within the collection. The results are shown in Table 4. The InferSent models are much closer to our Word, Trigram model on the caption datasets than overall, and InferSent trained on SNLI shows the largest difference between its overall performance and its performance on caption data.

We also compare the performance of these models on the STS Benchmark under several conditions (Table 5). Unsupervised results were obtained by simply using cosine similarity of the pretrained embeddings on the test set with no training or tuning. Supervised results were obtained by training and tuning using the training and development data of the STS Benchmark.

Data	AllNLI	SNLI
Overall mean diff.	10.5	12.5
MSRvid (2012) diff.	5.2	4.6
Images (2014) diff.	6.4	4.8
Images (2015) diff.	3.6	3.0

Table 4: Difference in correlation (Pearson's $r \times 100$) between "Word, Trigram" and InferSent models trained on two different datasets: AllNLI and SNLI. The first row is the mean difference across all 25 datasets, then the following rows show differences on three individual datasets that are comprised of captions. The InferSent models are much closer to our model on the caption datasets than overall.

We first compare unsupervised results on the entire test set, the subset consisting of captions (3,250 of the 8,628 examples in the test set), and

Model	All	Cap.	No Cap.
Unsupervised			
InferSent (AllNLI)	70.6	83.0	56.6
InferSent (SNLI)	67.3	83.4	51.7
Word, Trigram	79.9	87.1	71.7
Supervised			
InferSent (AllNLI)	75.9	85.4	64.8
InferSent (SNLI)	75.9	86.4	63.1

Table 5: STS Benchmark results (Pearson's $r \times 100$) comparing our Word, Trigram model to InferSent trained on AllNLI and SNLI. We report results using all of the data (All), only the caption portion of the data (Cap.), and all of the data except for the captions (No Cap.).

the remainder. We include analogous results in the supervised setting, where we filter the respective training and development sets in addition to the test sets. Compared to our model, InferSent shows a much larger gap between captions and non-captions, providing evidence of a bias. Note that this bias is smaller for the model trained on AllNLI, as its training data includes other domains.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal.

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 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th international conference on Computational Linguistics*, page 350.

Felix Hill, Kyunghyun Cho, and Anna Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4).

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM.

Yangfeng Ji and Jacob Eisenstein. 2013. Discriminative improvements to distributional sentence similarity. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 891–896, Seattle, Washington, USA.

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 28*, pages 3294–3302.

Alice Lai and Julia Hockenmaier. 2014. Illinois-LH: A denotational and distributional approach to semantics. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 329–334, Dublin, Ireland.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics-Volume 1*, pages 1–7.

Marco Marelli, Luisa Bentivogli, Marco Baroni, Raffaella Bernardi, Stefano Menini, and Roberto Zamparelli. 2014. SemEval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pages 271–278, Barcelona, Spain.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevich, Benjamin Van Durme, and Chris Callison-Burch. 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Alec Radford, Rafal Jozefowicz, and Ilya Sutskever. 2017. Learning to generate reviews and discovering sentiment. *arXiv preprint arXiv:1704.01444*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642.

Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2):165–210.

Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.

Han Zhao, Zhengdong Lu, and Pascal Poupart. 2015. Self-adaptive hierarchical sentence model. In *Proceedings of IJCAI*.

Peng Zhou, Zhenyu Qi, Suncong Zheng, Jiaming Xu, Hongyun Bao, and Bo Xu. 2016. Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling. In *Proceedings of COLING*, pages 3485–3495, Osaka, Japan.