Making the Most of Your Data: Few-Shot Learning for Automated Essay Scoring

Abel John and Samarth Kadaba
{abeljohn, skadaba}@stanford.edu

Motivation

Developing content-based, classroom-oriented automated essay scoring systems gives teachers the ability to bias large scale score estimation models with their own preferences for essay writing. This preserves the personal nature of **student-teacher relationships** however poses an algorithmic challenge due to the inherent scarcity of teacher-provided reference essays. Here, we aim to tackle this problem through learning context-dense embeddings which more closely reflect teacher-provided scores from **limited training samples**.

To this end, we:

1. Demonstrate novel methods of augmenting reference samples using **semantic substitution**.
2. Analyze **performance trade-offs using different pairwise loss functions**.
3. Investigate recurrent architectures for **second-order document embeddings**.

We show that in classifying essay samples according to a non-binary rubric:

1. We **outperform baseline models** evaluated with the same data scarcity constraints
2. Our learned embeddings **perform well in clustering** reflecting their applicability towards Semantic Textual Similarity tasks and giving instructors the ability to quickly identify groups of students in need of greater support.
3. Ensemble methods are **well-suited for dealing with data scarcity**

Methods

1 Multicriterion Optimization

- Coupling cross-entropy loss with pairwise distance metrics between samples helps faster and more reliable convergence.
- Our loss is a combination of three terms: the contrastive (1) or triplet loss (2) between two samples scalarized to reflect optimization priority of classification versus embedding accuracy, and the classification cross-entropy loss (3).

$$\mathcal{L}(x_1, x_2) = \mathbb{1}_{x_1=x_2} \cdot \|\mathbf{x_1} - \mathbf{x_2}\|_2{}^2 + (1 - \mathbb{1}_{x_1=x_2}) \cdot \max(0, \alpha - \|\mathbf{x_1} - \mathbf{x_2}{}_2\|)^2 \qquad \textbf{(1)}$$

$$\mathcal{L}_{triplet}(x^a, x^p, x^n) = \left[\|x^a - x^p\|^2 - \|x^a - x^n\|^2 + \alpha\right]_+ \qquad \textbf{(2)}$$

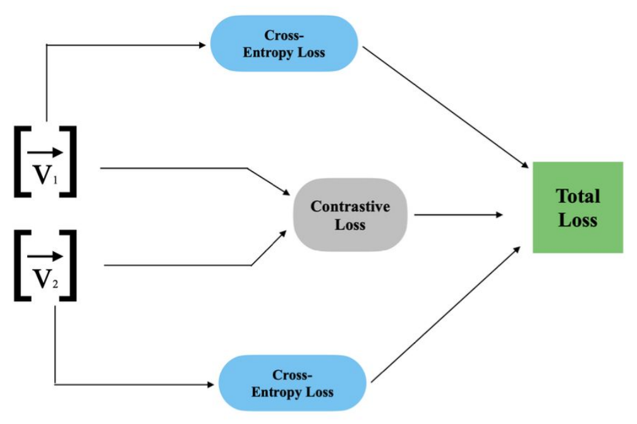

Figure 1: Shared-weight architecture for contrastive loss.

$$\mathcal{L}_{CE}(\hat{y}, y) = -\sum_{i=1}^{n} y_i \log \hat{y}_i \qquad \textbf{(3)}$$

2 Semantic Substitution

- To mitigate the risk of overfitting on sparse data, we introduce a form of adversarial training in which robustness is introduced by adding noise to our training samples.
- We compute the differential sensitivity of class predictions to each token, using these as a heuristic for probabilistically weighting semantic substitution of the given word (Fig. 4).

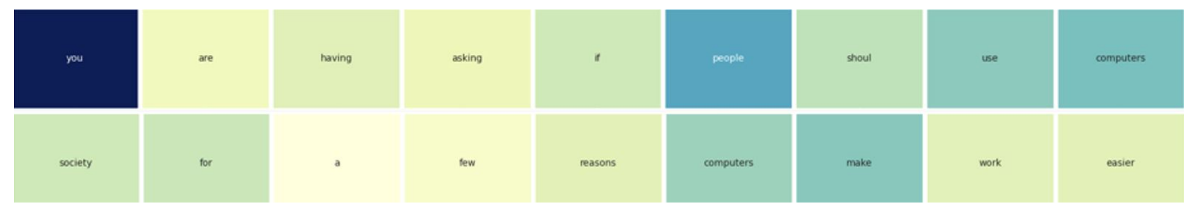

Figure 4: Two excerpts from sample essays showing the weights of each token with respect to model outputs (class predictions). The magnitude of the gradient sheds light on how perturbation of high-weight words may affect training. We observe **prompt-relevant words weighted highly** such as “computers.”

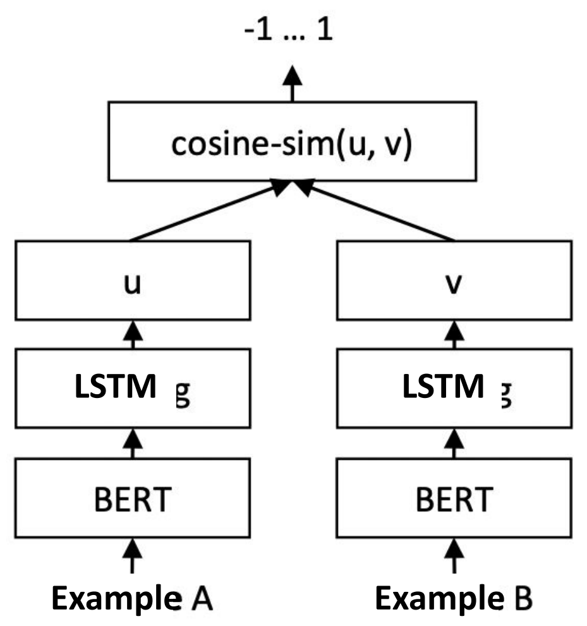

Figure 2: Diagram representing our BERT-LSTM architecture at a high level.

3 RNNs + Transfer Learning

- We test the addition of an LSTM layer (Fig. 2, Fig. 6) for the capture of contextual meaning, better performance on downstream tasks, and a robustness to noise.
- Due to our few-shot paradigm restricting our training capability, we instead train BERT on a dataset of TOEFL essay responses and freeze the weights for use with our model.
- The intent is for less context-dependent earlier layers to be better optimized for essay scoring.

Model Evaluation

Multicriterion Loss Evaluation

- We show that Siamese networks with both contrastive and triplet loss have marginal improvements compared to baseline models in few-shot settings (Table. 1). In fact, excluding methods for dealing with sparse data, baseline models seem to outperform pairwise methods in multi-class settings.
- We interpret these results as an **inability to generalize pairwise comparisons from just one example** of each class. The resulting overfitting is reflected by non-convergence of validation loss.
- Semantic textual similarity, however, is marginally reflected in down-projected clusters by the **spatial separation of samples from different classes** (Fig. 3).

Loss Architecture vs. Number of Classes			
	2	3	5
Baseline (CE Loss Only)	0.59	0.42	0.31
Contrastive Loss	0.5	0.45	0.30
Triplet Loss	0.79	0.39	0.09

Table 1: Average accuracies and F1 scores for our three main methods: CNN, LSTM, and pretrained transformer

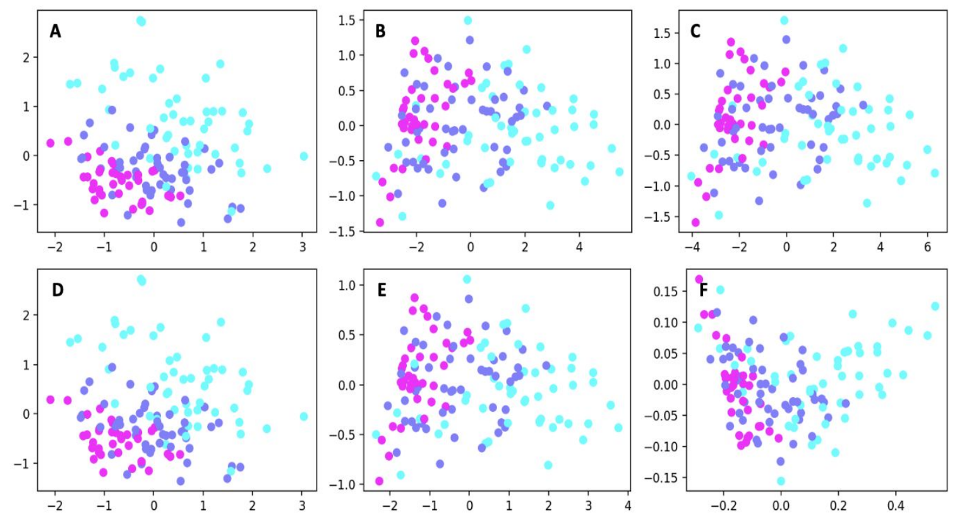

Figure 3: From left to right, Baseline, Contrastive, and Triplet clusters for 3 classes with one-shot learning. (A-C) show down-projected embeddings after training on one sample. It is clear that **contrastive and triplet networks better identify distinction between classes compared with the baseline models.** (B-F) Clusters for respective architectures shown with a single augmented sample. We observe that Triplet loss networks better adapt to euclidean representations of document embeddings from added data compared with Contrastive and Baseline models.

Data Augmentation

- We see different **utilization efficiencies** of augmented data based on the parameters of the input and model.
- When the augmented samples equal or outnumber the desired number of predicted classes, we empirically observe declining performance (Fig. 4).
  - This can be understood in the context of **Signal-to-Noise (SNR)** ratio.
- In all cases, we see that augmenting the dataset by at least one sample results in significant performance improvements both in terms of accuracy (Table. 2) and confusion (Fig. 5).

Loss Architecture vs. Number of Classes			
	2	3	5
Baseline + 1	0.64	0.48	0.24
Contrastive Loss + 1	0.73	0.51	0.30
Triplet Loss + 1	0.80	0.64	0.34

Table 2: 1 additional sample per class. Observe that for increasing number of target classes, pairwise architectures outperformed the baseline model. We reason that pairwise architectures, namely Contrastive and Triplet networks **better utilize added data and can discriminate noise between true signals when it comes to augmented samples.**

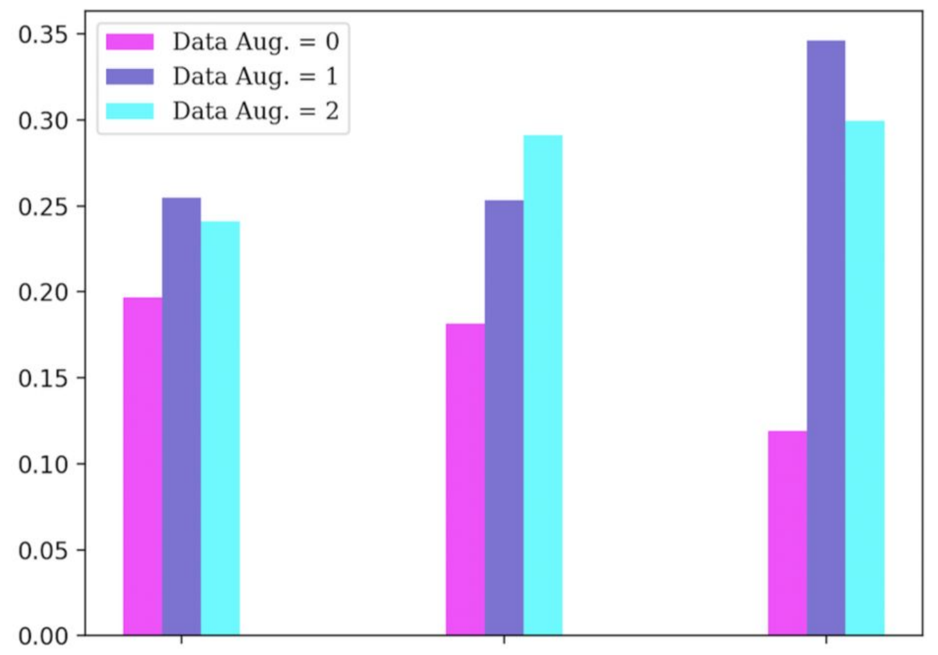

Figure 4: From left to right: Baseline, Contrastive, Triplet architectures. Note that in none of our trials did we see the absence of data augmentation outperform its inclusion.

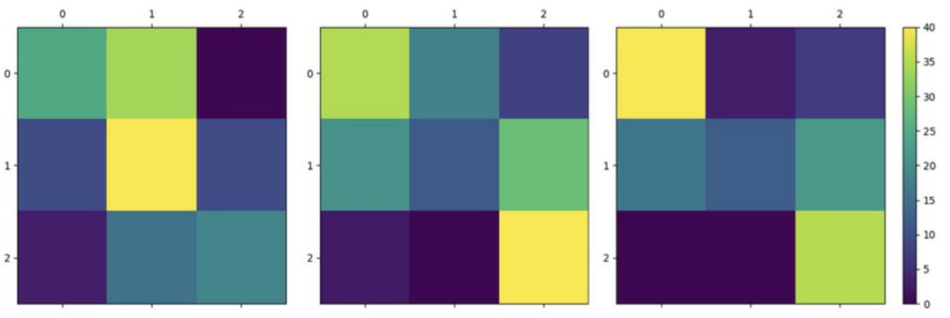

Figure 5: From left to right: Baseline, Contrastive, and Triplet confusion matrices for 3-class prediction via one-shot learning with 1 augmented sample per class. Lighter colors along the diagonal and darker off-diagonal entries signify lower confusion between classes. We observe that while true positive prediction for classes 1 and 3 are high amongst all three architectures, baseline models produce many more false predictions.

Loss Architecture vs Number of Classes for RNN+Siamese			
	2	3	5
Baseline (CE Loss Only)	0.84	0.60	0.35
Contrastive Loss	0.56	0.57	0.37
Triplet Loss	0.54	0.42	0.21

Table 3: 0 data augmentation with no transfer learning. Observe the general improvement in accuracy across all loss functions (compared with Table 1 above). LSTMs produce a denser feature space from which to make predictions on classes. This results in greater accuracy.

Loss Architecture vs Number of Classes for RNN+Siamese			
	2	3	5
Baseline (CE Loss Only)	0.83	0.60	0.33
Contrastive Loss	0.5	0.66	0.38
Triplet Loss	0.51	0.52	0.30

Table 4: 0 data augmentation with transfer learning. Note that the accuracy for triplet loss is significantly improved when transfer learning is used to re-weight the model (compare with Table 3 above). Also notice the relative consistency of accuracy across loss functions for the 5 class objective, whereas prior results indicate greater variance and particularly worse performance for triplet loss. We observe that transfer learning helps learn generalizable features for text that produce high-fidelity embeddings for downstream tasks such as classification and assessing semantic textual similarity.

Generalizability with RNNs

- Averaged across classes and loss functions, we see that **inclusion of the recurrent architecture leads to an 6% improvement compared to using only BERT and Siamese networks (Table 3).**
- Rather than the model’s size leading to overfitting, we see that the mapping of long-term dependencies helps with identifying STS (Semantic Textual Similarity).
- The inclusion of transfer learning sees an **average improvement of 8% over the baseline (Table 4)**, indicative of utility, but to a lesser extent when compared to the addition of RNN architecture (Fig. 6).

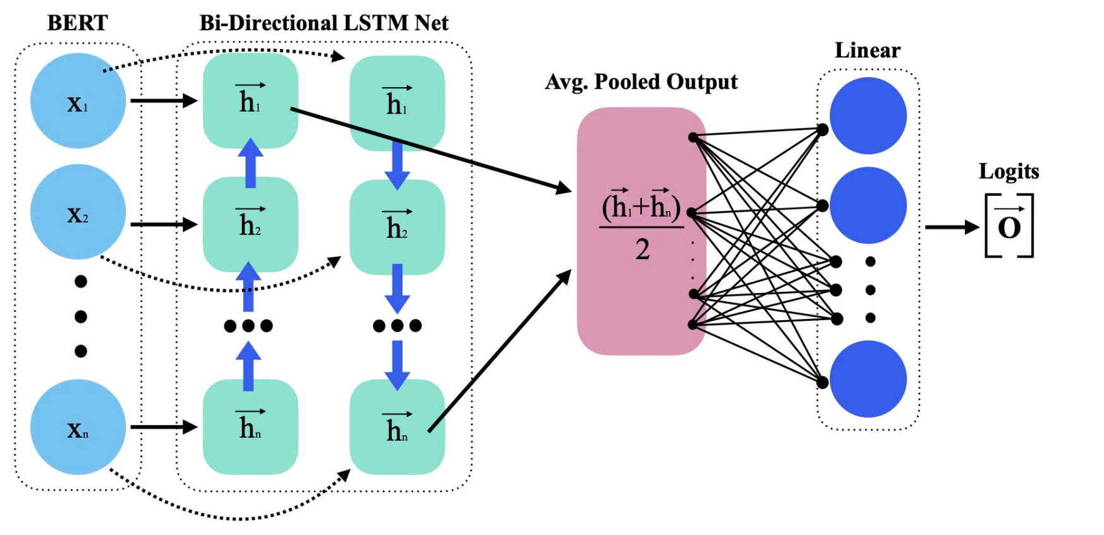

Figure 6: Diagram visualizing the model’s BERT-LSTM architecture at a more granular level than Fig. 2. Note the output of both ends of the LSTM are averaged to produce the new hidden state.

Analysis

Multitask training helps learn useful representations

- Pairwise distance metrics better capture semantic textual similarity that is beneficial towards downstream tasks such as classification.
- A multi-criterion loss was empirically observed to perform better than individually optimizing for pairwise distances or classification. We reason that this phenomenon occurs uniquely when the **losses are complementary in nature with respect to properties of the text.** Thus, they motivate optimizer steps in parallel directions contributing to faster and more robust convergence.

Semantic substitution benefits pairwise architectures

- We posit our model performs well with less data supplemented by an equivalent amount of augmented samples because they incorporate noise that adversarially updates weights.
  - **Improves generalization and reduces overfitting.**
- A sort of implicit "suspicion" emerges from weighting loss terms based on the euclidean distance of documents in their embedding-spaces (for pairwise loss).

RNNs build towards complex understanding of language

- Since BERT’s transformer architecture processes input in parallel, the inclusion of an LSTM layer builds on the contextualized embeddings produced by BERT to **capture the sequential relationships between words**.
- We reason that transfer learning on TOEFL essays was successful due to earlier layers of the model producing **embeddings that are a more abstract, domain-agnostic representation of the essay, and as such their weights are generalizable across essay-scoring models.**
- The TOEFL dataset in particular provides a number of examples written by English learners that delineate the essay structure for different scoring categories without respect to a specific prompt.

Conclusions and Future Work

Significance and Applications

- In this paper we investigated avenues for automated essay scoring in data-sparse contexts
- We found that for just a single sample per class (or in the case of triplet loss, two per class), we were able to reach 84% accuracy in binary classification, 66% for three classes, and 38% for five classes.
  - For example, we show that with just one example of a “good”, “average”, and “bad” essay, teachers can **accurately cluster students with their peers to plan group-based activities** that benefit those in need of additional assistance.
- We also show broad trend associated with augmenting data in pairwise systems. We empirically show that **Contrastive and Triplet architectures are better suited for learning generalizable functions from perturbed data**.
- We further demonstrate that generating second order embeddings and employing **transfer learning helps the model learn structural features to language that extend well to downstream tasks**.
- Our work has interesting interpretations from an information theoretic perspective as we reconcile **ideas of SNR and “suspicion”** to intuit model behavior.

Future Work

- We hope to investigate if our results still hold for the current generation of pretrained models, and to what extent classification improves by using a the latest innovations.
- We may experiment with additional augmentation strategies including “surface realization” or “reverse lemmatization” for substitution, random phrase addition/deletion, and selection of high-variance samples for few-shot training.

References

1. The Hewlett Foundation. 2012. Automated scoring algorithm for student-written essays. In Kaggle.
2. Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.
3. David Roa. 2018. Analysis of short text classification strategies using out of-domain vocabularies.