Triplet Loss based Siamese Networks for Automatic Short Answer Grading

Nagamani Yeruva
JNTU Kakinada
Kakinada, India
yeruvanagamani@gmail.com

Sarada Venna
JNTU Kakinada
Kakinada, India
vennasarada13@gmail.com

Hemalatha Indukuri
SRKR Engineering College (A)
Bheemavaram, Andhra Pradesh, India
indukurihemalatha@gmail.com

Mounika Marreddy
IIIT-Hyderabad
Hyderabad, India
mounika.marreddy@research.iiit.ac.in

ABSTRACT

Grading student work is critical for assessing their understanding and providing necessary feedback. However, answer grading can become monotonous for teachers. On the standard ASAG data set, our system shows substantial improvements in classification disparity of correct and incorrect answers from a reference answer compared to baseline methods. Our supervised model (1) utilizes recent advances in semantic word embeddings and (2) implements ideas from one-shot learning methods, which are proven to work with minimal. We present experimental results from a model based on different approaches and demonstrates decent performance on standard benchmark dataset.

KEYWORDS

Automated short answer grading, word embeddings, one-short learning

ACM Reference Format:
Nagamani Yeruva, Sarada Venna, Hemalatha Indukuri, and Mounika Marreddy. 2022. Triplet Loss based Siamese Networks for Automatic Short Answer Grading. In *Forum for Information Retrieval Evaluation (FIRE '22), December 9–13, 2022, Kolkata, India.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3574318.3574337

1 INTRODUCTION

Grading student work is pivotal to judging the understanding level of students in the subject and helps to provide the instructor feedback. The student's ability can be measured at either subject level (short or long answers) or object level (multiple choice). Subject-level grading provides good feedback for the student, but it becomes tedious and monotonous for the instructor. There are specific scenarios, such as worldwide sites with limited teacher availability or individual or group study sessions done outside the class where the instructor is unavailable, yet students need feedback for their knowledge of the subject.

Automatic grading systems such as computer-assisted assessments have been very popular for many years in educational institutions, where students have to choose the correct answer from multiple options for a given question. However, these grading systems do not capture the reasoning and self-explanation [3, 28]. Some automatic grading systems have been specially designed for subjective-based tests such as Automated Essay Scoring (AES), which includes checking the style, grammaticality [7] (AES) and Automated Short Answer Scoring [12, 22] (ASAS). The basic difference between AES and ASAS is in the length and focus of the assessment.

In this paper, we are primarily interested in automated short answer scoring of the student-constructed answer with reference to the instructor's answer. The main challenge in grading student-constructed short answers, given the instructor's answer, is due to its complex natural language understanding. This complexity varies from the linguistic level (the same answer can be written in multiple ways) to the subjective level (one question has multiple answers). This motivates us to introduce word embeddings (GloVe and Meta-Embeddings) as features to bring the semantically related information between student answers and instructor answers.

2 RELATED WORK

In literature, there are a number of approaches that have been proposed for automated short answer grading. Early, the short answer grading relies on the manually extracted patterns (e.g., regular expressions) from instructor-student reference answer pairs [17, 20, 26]. Such patterns encode key concepts representative of good answers. If an annotated corpus is available, these patterns can be supplemented by learning additional patterns semi-automatically. This step requires human intervention that natural language processing can help to eliminate. In [22], describes a comparative study of several machine learning techniques, including inductive logic programming, decision tree learning, and Bayesian learning, to the earlier pattern matching approaches, with encouraging results.

As opposed to pattern matching, direct semantic matching has been explored in early work like [11]. This method matches the syntactical features of student responses (i.e., subject, object, verb) to that of a set of correct responses. With the recent advances in Natural Language Processing techniques, semantic approaches have

FIRE '22, December 9–13, 2022, Kolkata, India

ACM ISBN 979-8-4007-0023-1/22/12…$15.00
https://doi.org/10.1145/3574318.3574337

Table 1: Top 10 features for the words evaporate, crystallize, and generate from the three methods

Word	Meta-Embeddings (Top 10)	GloVe (Top 10)	Word2Vec (Top 10)
evaporate	evaporates, dissipate, solvents disappear, disintegrates, seep dissolve, dries, dwindle vaporize	evaporates, dissipate, vaporize condense, seep, disintegrate vanish, volatilize, dissolve dries	evaporated, vanish, dwindle dissipate, disappear, disintegrate erode, shrivel, wither fade
crystallize	crystallizes, amorphous, solidify condense, coalesce, precipitate refocus, semiconductors decompose, metal	crystallise, crystalize, solidify congeal, precipitate, dissociate evaporate, coalesce distill, coagulate	crystallizing, coalesce, encapsulate elucidate, formulate, congeal synthesize, distill, condense internalize
generate	generating, produce, elicit create, generator, derive emit, build, engender attract	generating, create, produce efficiently, utilize, derive enable, construct, provide resultant	generating, to generate, create produce, garner, derive engender, elicit, translate sustain

gained popularity over time [6, 9, 18, 19]. These systems typically use a large set of similarity measures as features such as word and character n-grams for string similarity, TF-IDF for vector space measures [24], WordNet for deeper semantic similarity [13] and latent semantic analysis (LSA) for distributional methods [10] used in a supervised learning model. However, a more extensive set of features leads to implementation difficulty and higher system run time. ASAS is closely related to text similarity, which is essentially the problem of detecting and comparing the features of two texts. Semantic text similarity systems can serve as a source of significant new features and design elements for ASAS [1, 4, 5, 14].

Recent advances in distributional word semantics and neural methods for word embeddings have leveraged the ASAS systems. In [23, 27], uses pre-trained word vectors [16] for both student and reference answers to train regression and classification models. There are two NLP tasks related to ASAS such as Automated Essay Scoring (AES), which scores the essays based on composition, fluency, grammatical correctness, etc. [15], and Paraphrase Detection (PD), the objective is to detect if two sentences or passages have the same meaning [25]. However, unlike PD, ASAS asymmetric, and there may not be a one-to-one correspondence between concepts in the model and student answers.

The structure of the paper is as follows. Section 3 describes the approach we are using to build the model, while Section 4 discusses the experimental setup. Section 5 presents the comparative results of various models along with the analysis of the results. Section 6 presents concluding remarks and future work.

3 OUR APPROACH

We first describe the following things in our approach. (i) Pre-processing of data (ii) Word embeddings (iii) Baseline evaluation. (iv) Automated evaluation (v) Network architecture.

3.1 Pre-processing of data

At the initial stage, pre-processing steps performed for the corpus were tokenization and stop-word removal (ignore function words and high-frequency, low-content verbs) by which essential keywords can be formed. To properly adjust our network to accept the variable length of answers, we need to find the optimal number of words (tokens) from which we either truncate (for the too-long elements) or pad (for the too-short ones) to simultaneously limit the loss of information (truncating) and the number of empty tokens (padding) as much as possible. (ii) selection of smaller and less complicated answers (by short answer definition). A cross-validation technique is applied for the comparison with answer size and question-answer keywords overlap.

3.2 Word embeddings

Word embeddings like Word2Vec, Glove, Meta-Embeddings, etc., are known to capture the semantics of words based on the context and the co-occurrence of different words. We use these as pre-trained vectors at the embedding layer. Global vectors for word representations (GloVe) [21] model combines the benefits of the Word2Vec skip-gram model when it comes to word analogy tasks with the help of matrix factorization methods that can exploit global statistical information. In [30], the idea of Meta-Embeddings has been proposed and has two benefits compared to individual embedding sets: enhancement of performance and improved vocabulary coverage. So, we need a distributional representation of word embeddings for query expansion. For e.g., considering a reference answer, "water evaporates in the sun leaving the salt behind," for the question "how to separate salt from water", we observed student answers contain the keywords (dried, dries) not found in the morphological expansion of word2vec. Hence we need advanced methods like GloVe [21] and Meta-Embeddings [30] for better word senses. Table 1 describes top-10 similar words for “evaporate", "crystallize", and 'generate" based on three word embeddings. These are the words we observed from student answers, where a student used multiple semantics for these words.

3.3 Baseline evaluation

The baseline method explains simple comparison using query expansion of keywords (nouns) using morphological forms and Glove vectors and selecting a subset of keywords using word-net graph distance. For e.g., after removing the stop-words from the reference answer, the keywords present are [“high", “string", “pitch", and “short"]. Using these keywords, find the morphological words for every keyword say for the word (“high" : “mellow", “heights", “high-school", “luxuriously", and “high-pitched") are some morphological words. The query expansion should be done by identifying the top 5 words which are having high similarity scores with morphological words and other keywords [“string", “pitch", “short"] using GloVe embeddings. This process is repeated for all the keywords.

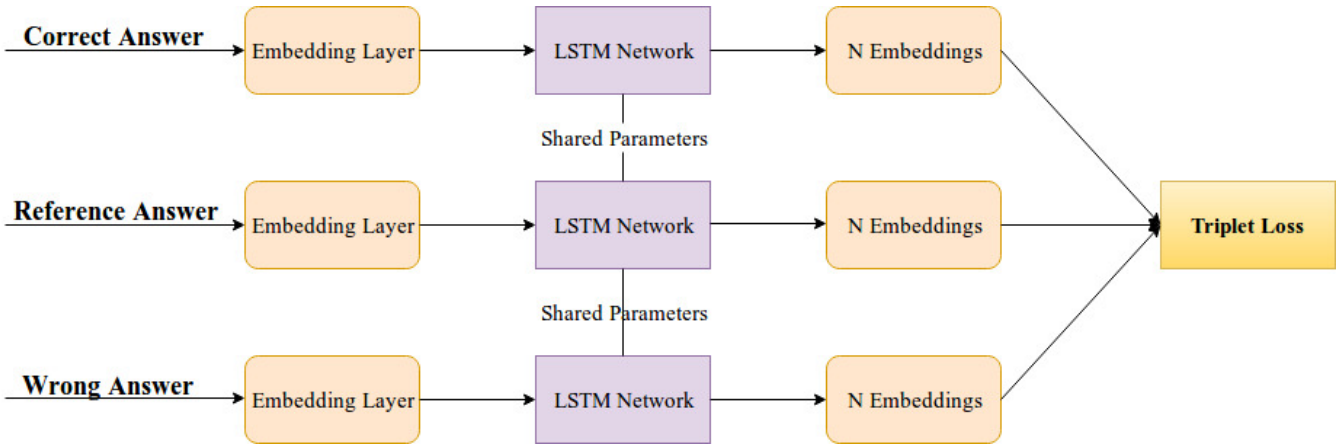

Figure 1: Siamese LSTM Network with Triplet Loss

3.4 Automated evaluation and architectures

Begin with the essence of using glove embedding pplus Siamese: bringing similar embeddings together in hyperspace without bothering about query expansion or semantic similarity.

3.4.1 Current method using L2 loss. The Siamese LSTM model we explore captures the semantic relatedness among student reference answer pairs. For the input to this model, we use student reference answer pairs (x_i^r, x_i^s). It should be noted that the network weights in two Siamese sub-networks are shared, and we use L2 distance as the negative similarity function to express the degree of relatedness between the pair of answers on the top layer of the network.

$$L2(x_i^r, x_i^s) = \left|\left|f(x_i^r) - f(x_i^s)\right|\right|_2^2 \tag{1}$$

3.4.2 Current method using Triplet loss. The triplet loss is motivated from [29]. Here we want to ensure that a reference answer x_i^r of a specific query is closer to all student correct answers x_i^c of the same query than it is to any student wrong answers x_i^w of any other query. Thus we want

$$\left|\left|f(x_i^r) - f(x_i^c)\right|\right|_2^2 + \alpha < \left|\left|f(x_i^r) - f(x_i^w)\right|\right|_2^2 \tag{2}$$

$\forall(f(x_i^r), f(x_i^c), f(x_i^w)) \in \tau$, where α is a margin enforced between positive and negative pairs. τ is the set of all possible triplets in the training set and has cardinality N. The loss that is being minimized is then L.

$$\sum_i^N \left[||f(x_i^r) - f(x_i^c)||_2^2 - ||f(x_i^r) - f(x_i^w)||_2^2 + \alpha\right] \tag{3}$$

The benefits of triplet loss over L2 loss are (i) similar examples brought together, and dissimilar ones have moved apart, (ii) an increase in data set size due to triple comparisons, (iii) the possibility of success with the independent dataset (other subjects) due to domain independent training nature of Siamese network.

3.5 Network Architectures

Siamese networks [2] are dual-branch networks with the weights, i.e., they consist of the same network copied and merged with an energy function. A Triplet network (inspired by "Siamese network") is comprised of three instances of the same feed forward network (with shared parameters). As shown in Figure 1, the network is fed with three input samples: the correct answer, reference answer, and wrong answer. The network outputs two intermediate values - the triplet loss between the embedded representation of two of its inputs from the representation of the third. We use GloVe pre-trained embeddings as input features for each network at the initial embedding layer. The Long Short-Term Memory [8] variant of RNNs, in particular, has had success in tasks related to natural language processing used as the hidden network has shared parameters between three networks. The embedded representation of the network is f(x). The one before the last layer will be the vector. The network architecture allows the task to be expressed as a two-class classification problem, where the objective is to correctly classify which of x^c and x^w is of the same class as x.

4 EXPERIMENTS

4.1 Dataset

We used the dataset from SemEval-2013 Task 7: The Joint Student Response Analysis [1] for training our system. There is a total of three subtasks in Task 7. In this paper, we worked on both subtask 2way response analyses for SciEntsBank corpora. In this 2way task, the system must classify the student's answer according to one of the two categories: correct or incorrect. The dataset contains 135 questions and 5000 student answers. The total dataset we used in experiments was 4969 reference student answer pairs, eliminating unseen answers and unseen questions.

[1] https://www.cs.york.ac.uk/semeval-2013/task7/index.php%3Fid=data.html

5 RESULTS AND DISCUSSION

5.1 Baseline: simple GloVe vectors based evaluation.

To compare the performance of our model against the baseline simple glove vectors based evaluation. In this setting, we eliminate all the stop words in reference student answer pairs, which results in the necessary keywords. For each keyword, we identified the most similar words using WordNet and using these similar words. We calculated the similarity score between the other keywords present in the sentence using GloVe. The top-5 words with high similarity scores are considered for each keyword. Stemming is performed on each word list and by combining all word lists to get the final words for the answer. The similarity score is calculated by simply matching the reference answer word list and the student answer word list. We ran this setup for one word matching and more than 5-words matching and compared it with actual labels. Figure 2 display the precision, recall, F1-score, and accuracy for both baselines like 1-word match and 5-word match.

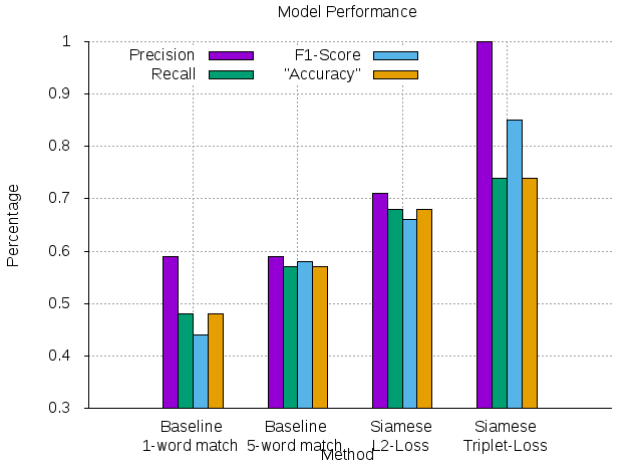

Figure 2: Comparison of Siamese Triplet Network results with Siamese L2-loss and Baseline methods

5.2 Siamese network with L2 loss based evaluation.

In this setup, we pass the two sets of data: reference answer features for one LSTM network and student answer features for another LSTM network. Since the LSTM network has shared parameters, the output of the two networks is passed to the L2 loss function to measure the error. We achieved an F1-score of 68% using this network, which is better than the baseline evaluation.

5.3 Siamese network with Triplet loss based evaluation.

The dataset we used here is having a total of 4969 reference-student answer pairs of which 2008 pairs are correct answer pairs and 2961 pairs are wrong answer pairs. We form the triplets by combining reference answer, correct answer, and incorrect answer which are in pair with same reference answer. The total number of triplets we generated are 36735 and we split this dataset into train, validation and testing sets. The triplet loss parameter (α) we use is 0.2 and some of the network parameters are (batch_size = 30, epochs = 5, optimizer = Adam, Maximum Sequence Length = 20). This model shows an improvement over baseline model and over Siamese network with the L2-loss model in terms of F1-score of 0.16 and 0.08 respectively.

5.4 Discussion

Our approach aimed to introduce a reliable, transferable, and scalable automated short answer grading system, which can account for the proximities of semantically similar words through word embeddings without explicit query expansion techniques. From our quantitative and qualitative results, it is evident that using Siamese networks with L2 loss and then with triplet loss has decent performance gains as intended. The Siamese network was meant to pull the embeddings of correct student answers closer to reference answers and to pull apart the incorrect answers from reference answers. This can be further improved by increasing the dataset size and expanding the dataset to different domains.

Without restricting ourselves to binary classification, the approach can be further enhanced to assign ordinal scores to student answers, given reference answers. The roadblock was a properly annotated dataset. We have gathered several short answers from students during university examinations. But this dataset still needed more additions and two rounds of validation to be of use for research. We plan to convert SemEval 5way and 7way validations into ordinal scores on a scale of 1 to 10, where the 'correct' answer gets a perfect score of 10, and the 'unrelated' answer receives a score of 1. We hope to add this approach to our future research.

Opportunities also exist to refine and improve the scaling aspect of automatic short answer grading. We can pick the answers rated closest to the reference answer and use them for cluster analysis. This can quickly bring together all the like answers and expand the vocabulary as a cyclic process.

6 CONCLUSION & FUTURE WORK

In this paper, we deal with the problem of automated short answer grading (ASAG) by utilizing the recent advances in semantic word embeddings like GloVe vectors and a one-shot learning method Siamese network with triplet loss. We use the SemEval-2013 task-7 sciEntsBank 2way dataset for building the model. On the standard ASAG dataset, our system shows substantial improvements in classification disparity of correct and incorrect answers from a reference answer compared to baseline methods. Experimental results show that our model better identifies the student reference answer pairs. We plan to extend this model to other available SemEval-2013 task-7 datasets (sciEntsBank 3way and 5way) for better ASAG feedback. The source code is publicly available at https://goo.gl/LdjMJU so that researchers and developers can collectively work on this exciting problem.

REFERENCES

[1] Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. 2012. Ukp: Computing semantic textual similarity by combining multiple content similarity measures. In *Proceedings of the First Joint Conference on Lexical and Computational*

Table 2: Siamese LSTM Model prediction results of reference-student answer pairs using Triplet Loss

Reference Answer	Student Answers	label predicted
A heat sink is any material that absorbs (a lot of) heat.	A heat sink is material that takes in more heat.	correct
Fabric A is smoother and has a finer texture. Fabric B is rougher and has a coarser texture.	A. Might be soft B. might be a little bumpy.	correct
When the string was shorter, the pitch was higher.	The sound had a higher pitch because it had faster vibrations.	incorrect
C. Black absorbs more heat (energy) than white. Pan C has the most dark surface area so C would heat up the fastest and have the highest temperature.	C. The one with 3 disks could of got hot because it had more disks than A or B. Since it has more disks then it was the one that got the hottest because the sun heat got in it more than the other 2.	incorrect

Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation. Association for Computational Linguistics, 435–440.

[2] Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, Vol. 1. IEEE, 539–546.

[3] Gráinne Conole and Bill Warburton. 2005. A review of computer-assisted assessment. *ALT-J* 13, 1 (2005), 17–31.

[4] Lushan Han, Abhay L Kashyap, Tim Finin, James Mayfield, and Jonathan Weese. 2013. UMBC_EBIQUITY-CORE: semantic textual similarity systems. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, Vol. 1. 44–52.

[5] Christian Hänig, Robert Remus, and Xose De La Puente. 2015. Exb themis: Extensive feature extraction from word alignments for semantic textual similarity. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*. 264–268.

[6] Michael Heilman and Nitin Madnani. 2013. ETS: Domain adaptation and stacking for short answer scoring. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, Vol. 2. 275–279.

[7] Derrick Higgins, Jill Burstein, Daniel Marcu, and Claudia Gentile. 2004. Evaluating multiple aspects of coherence in student essays. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*.

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

[9] Sergio Jimenez, Claudia Becerra, and Alexander Gelbukh. 2013. SOFTCARDINALITY: Hierarchical text overlap for student response analysis. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, Vol. 2. 280–284.

[10] Thomas K Landauer, Peter W Foltz, and Darrell Laham. 1998. An introduction to latent semantic analysis. *Discourse processes* 25, 2-3 (1998), 259–284.

[11] Claudia Leacock and Martin Chodorow. 2003. C-rater: Automated scoring of short-answer questions. *Computers and the Humanities* 37, 4 (2003), 389–405.

[12] Claudia Leacock, George A Miller, and Martin Chodorow. 1998. Using corpus statistics and WordNet relations for sense identification. *Computational Linguistics* 24, 1 (1998), 147–165.

[13] Michael Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of the 5th annual international conference on Systems documentation*. ACM, 24–26.

[14] André Lynum, Partha Pakray, Björn Gambäck, and Sergio Jimenez. 2014. NTNU: Measuring semantic similarity with sublexical feature representations and soft cardinality. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. 448–453.

[15] Manvi Mahana, Mishel Johns, and Ashwin Apte. 2012. Automated essay grading using machine learning. *Mach. Learn. Session, Stanford University* 5 (2012).

[16] 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*. 3111–3119.

[17] Tom Mitchell, Terry Russell, Peter Broomhead, and Nicola Aldridge. 2002. Towards robust computerised marking of free-text responses. (2002).

[18] Michael Mohler, Razvan Bunescu, and Rada Mihalcea. 2011. Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 752–762.

[19] Michael Mohler and Rada Mihalcea. 2009. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 567–575.

[20] Rodney D Nielsen, Wayne Ward, and James H Martin. 2009. Recognizing entailment in intelligent tutoring systems. *Natural Language Engineering* 15, 4 (2009), 479–501.

[21] 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)*. 1532–1543.

[22] Stephen G Pulman and Jana Z Sukkarieh. 2005. Automatic short answer marking. In *Proceedings of the second workshop on Building Educational Applications Using NLP*. Association for Computational Linguistics, 9–16.

[23] Keisuke Sakaguchi, Michael Heilman, and Nitin Madnani. 2015. Effective feature integration for automated short answer scoring. In *Proceedings of the 2015 conference of the North American Chapter of the association for computational linguistics: Human language technologies*. 1049–1054.

[24] Gerard Salton and Michael J McGill. 1986. Introduction to modern information retrieval. (1986).

[25] Richard Socher, Eric H Huang, Jeffrey Pennin, Christopher D Manning, and Andrew Y Ng. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Advances in neural information processing systems*. 801–809.

[26] Jana Z Sukkarieh, Stephen G Pulman, and Nicholas Raikes. 2004. Auto-marking 2: An update on the UCLES-Oxford University research into using computational linguistics to score short, free text responses. *International Association of Educational Assessment, Philadephia* (2004).

[27] Md Arafat Sultan, Cristobal Salazar, and Tamara Sumner. 2016. Fast and easy short answer grading with high accuracy. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1070–1075.

[28] Hao-Chuan Wang, Chun-Yen Chang, and Tsai-Yen Li. 2008. Assessing creative problem-solving with automated text grading. *Computers & Education* 51, 4 (2008), 1450–1466.

[29] Kilian Q Weinberger and Lawrence K Saul. 2009. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research* 10, Feb (2009), 207–244.

[30] Wenpeng Yin and Hinrich Schütze. 2015. Learning meta-embeddings by using ensembles of embedding sets. *arXiv preprint arXiv:1508.04257* (2015).