INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Off-Topic Spoken Response Detection Using Siamese Convolutional Neural Networks

Chong Min Lee, Su-Youn Yoon, Xihao Wang, Matthew Mulholland, Ikkyu Choi, Keelan Evanini

Educational Testing Service
660 Rosedale Road, Princeton, NJ, USA

clee001, syoon, xwang002,
mmulholland, ichoi001, kevanini@ets.org

Abstract

In this study, we developed an off-topic response detection system to be used in the context of the automated scoring of non-native English speakers' spontaneous speech. Based on transcriptions generated from an ASR system trained on non-native speakers' speech and various semantic similarity features, the system classified each test response as an on-topic or off-topic response. The recent success of deep neural networks (DNN) in text similarity detection led us to explore DNN-based document similarity features. Specifically, we used a siamese adaptation of the convolutional network, due to its efficiency in learning similarity patterns simultaneously from both responses and questions used to elicit responses. In addition, a baseline system was developed using a standard vector space model (VSM) trained on sample responses for each question. The accuracy of the siamese CNN-based system was 0.97 and there was a 50% relative error reduction compared to the standard VSM-based system. Furthermore, the accuracy of the siamese CNN-based system was consistent across different questions.

1. Introduction

In this study, we developed an off-topic response detection system for the automated scoring of non-native English speakers' oral proficiency. In a speaking proficiency test that elicits unconstrained spontaneous speech, some responses may include sub-optimal characteristics. Especially in practice tests, where the test score does not contribute to a consequential decision, test takers may not provide good-faith responses if they are fatigued, unmotivated, distracted, etc. For instance, students may recite their response to a previous question (which would be off-topic), and this may help them reduce disfluencies and generate fluent speech. As a result, it may make it more difficult for the automated scoring system to provide a valid assessment of the proficiency of the speakers. In order to address this issue, previous studies such as [1, 2, 3] researched automated systems for filtering out off-topic responses. By filtering out these responses and preventing automated scoring systems from generating erroneous scores, the robustness of the automated scoring system can be improved.

The task of off-topic response detection consists of creating word hypotheses using a speech recognition system, converting a spoken response into a vector of semantic units, and calculating their similarities with a set of sample responses or the questions used to elicit responses. Next, based on the various semantic similarity features, responses are either clustered into the predefined topic groups (closed-set topic-clustering) or classified into binary classes: on-topic or off-topic. This task is analogous to document clustering or query-by-example document retrieval.

Topic similarity between two documents has been frequently modeled by vector space models (VSMs) [4, 5], relative-frequency measures [5, 6], and document fingerprinting [7, 6]. These approaches have shown promising performance in various semantic modeling tasks. In particular, VSMs have been frequently used in information retrieval, and *tf-idf* (term frequency-inverse document frequency) has often been used as the method to weight each word in the VSM. However, these approaches rely on exact word matching, and they face challenges in measuring the similarity between words that are topically relevant but not identical.

Most of the previous studies about off-topic detection have been based on question-specific content models, such as a standard vector space model (VSM) built for each question. For the content model building, two different approaches have been used. First, for each question, corresponding sample responses are collected, and content models are trained on these responses (hereafter, response-based VSM) (e.g., [8]). In contrast, [9, 10] developed content models exclusively using the question texts for test questions for which no sample responses were available (hereafter, question-based VSM). The response-based VSM resulted in superior performance to the question-based VSM, but the question-based VSM had an important advantage over the response-based VSM because the system was immediately applicable to new questions.

Recently, deep neural networks (DNN) have been applied successfully to various natural language processing tasks, and DNN-based approaches have also resulted in substantial improvements for the task of measuring similarity between texts. In the DNN-based approaches, word embeddings were usually applied together in order to overcome the limitation of exact word matching in VSM approaches. For example, [11] applied DNNs for a sentence completion task, [12, 13] used them for the task of paraphrase identification, and [14] used them in the task of scoring similarities between pairs of sentences. In particular, [15, 16, 11] demonstrated that it was possible to achieve state-of-the-art performance in similarity-scoring tasks using siamese networks, which were first introduced by [17]. Siamese networks are characterized by shared weights between two subnetworks modeling inputs and a distance calculation layer following the two subnetworks. The structural characteristics make it possible to directly learn a function optimized to minimize the distance for a semantically similar pair while maximizing the distance for a semantically dissimilar pair. This advantage has led to the application of the siamese network to document similarity detection tasks.

There have been a few attempts to apply DNN approaches to the specific task of off-topic response detection. For example, [18] examined various word embedding-based features in calculating similarities between questions and responses. It showed

http://dx.doi.org/10.21437/Interspeech.2017-1174

that naive cosine similarity measures using word embeddings could show lower performance than similarity measures using *tf-idf*s. [3] developed topic models based on Recurrent Neural Network language models (RNNLM). In this study, responses were categorized as off-topic when the confidence score of a response using a target topic model was not among the N-best solutions; the results showed that RNNLM models achieved a substantial improvement over the standard VSM-based approach.

In this study, we define the task of off-topic response detection as a binary classification task. We describe an off-topic classification model using a siamese network and we demonstrate that the model performs better than a VSM-based classification model. Our main contributions are as follows. Firstly, our approach is the first attempt to model semantic relationships between on-topic responses and questions while simultaneously modeling relations between off-topic responses and questions. We build two types of pairs: a question and a corresponding on-topic response (i.e. on-topic pair), and a question and a corresponding off-topic response (i.e. off-topic pair). We apply a siamese network to learn semantic differences between the on-topic pairs and the off-topic pairs. Secondly, we use many more questions in evaluating off-topic detection models compared to previous studies. In [3], the assumption is made that students might use only a few limited topics for off-topic responses. Based on this assumption, closed-set off-topic detection was conducted using six question-based topic clustering models. However, in an actual large-scale assessment, the number of topics for off-topic responses will likely not be limited in this way. To address this issue, we created a set of off-topic responses elicited using a much larger number of questions.

2. Data

We used a large collection of spoken responses from a large-scale, standardized test of English speaking proficiency. The assessment was composed of questions in which speakers were prompted to provide approximately one minute of spontaneous speech. Each question asked test takers to provide information about or opinions on familiar topics based on their personal experiences and background knowledge.

100,000 responses were selected for the training and evaluation of an off-topic filtering model (hereafter, FM) set. First, we selected 20 test questions for on-topic responses (hereafter, on-topic questions). The questions were randomly selected from questions covering diverse topics such as classroom activities, education and learning, and jobs and work; 50,000 responses (2,500 responses per question) were collected for these questions. Next, we selected another 50 questions (hereafter, off-topic questions) which did not overlap with the on-topic questions. Since we did not have a large set of authentic off-topic responses collected from actual tests, responses from the off-topic questions were considered as off-topic responses. We collected 1,000 responses per off-topic question (50,000 responses in total). The amount of on-topic and off-topic responses in the data set was therefore balanced (50,000 each). The dataset was partitioned into training (60%) and evaluation (40%) partitions, and each partition was comprised of the same amount of on-topic and off-topic responses. There was no speaker overlap between the training and evaluation partitions.

In addition, we collected over 145,000 responses for training the $tf{\cdot}idf$ weighted VSM which was used in building the baseline classification model. The idf training dataset consisted of 125,000 responses elicited from 319 questions covering a wide range of diverse topics. We also created a tf training dataset containing 1,000 responses per on-topic question resulting in a total of 20,000 responses. In the beginning of this study, we examined two different idfs: an idf trained on idf training dataset and an idf trained on a dataset including both tf training and idf training datasets. In the examination, we could not find any meaningful performance improvement by using question-specific responses as a part of idf training data. Based on the finding, we decided to use a generic idf trained across responses for many different questions except questions used in the evaluation sets because we don't need to update idf model when we apply the model to new questions.

Each response was rated by trained human raters using a 4-point scoring scale, where 1 indicated a low speaking proficiency and 4 indicated a high speaking proficiency. The responses contained 106 words on average, but there was substantial variation in length among the responses, with word counts ranging from 1 to 218. There was also a strong positive correlation between test takers' oral proficiency scores and response length (test takers with low proficiency generally produced short responses); the Pearson correlation coefficient between the holistic proficiency scores and the number of words was 0.44 ($p < 0.01$). In addition, raters provided a score of 0 when test takers did not show any intention to directly respond to the question (the majority of these instances contain no spoken response). Finally, the raters also labeled responses as TD (technical difficulty) when they contained technical issues (e.g., background noise or audio distortion) that were substantial enough to make it impossible for the rater to provide a valid score. These TD and 0 responses were excluded from the current study since our focus is on the detection of topicality issues. The speakers, question information, and the average proficiency score for the FM set and VSM model training are presented in Table 1.

3. Methods

3.1. Automated Speech Recognition System (ASR)

A gender independent acoustic model (AM) was trained on 800 hours of spoken responses extracted from the same English proficiency test using the Kaldi toolkit [19]. The AM training dataset consisted of 52,200 spoken responses from 8,700 speakers. It was based on a 5-layer DNN with p-norm nonlinearity using layer-wise supervised backpropagation training. The language model (LM) was a trigram model trained using the same dataset used for AM training. This ASR system achieved a Word Error Rate of 23% on 600 held-out responses. Detailed information about the ASR system is provided in [20]. Word hypotheses were generated using this ASR system For each response in the FM set described in Table 1, the word hypotheses were generated using this recognizer and the set of features described in Sections 3.2 and 3.3 were generated from these ASR-based transriptions.

3.2. Response-based VSMs

As a baseline system, we trained a vector space model for each question separately since each question covered a different topic. In this study, we used word unigrams as the terms and responses as the documents. idf values were trained using the idf model train set, while tfs were trained for each question using the tf model train set. Following [21] and [1], we trained proficiency-level-specific tfs (one for each score class). All responses which received the same human score were converted into a single vector and a tf was trained from this vector. This

Table 1: *Characteristics of the experimental datasets;* M *is the average proficiency score.*

Dataset	Purpose	Description	Number			Proficiency score				
			Responses	Speakers	Questions	M	1	2	3	4
FM	FM training and evaluation	on-topic	50,000	25,000	20	2.75	3%	32%	53%	12%
	Siamese CNN training	off-topic	50,000	25,000	50	2.72	3%	33%	52%	11%
VSM training	tf training	on-topic	20,000	10,000	20 (identical as FM on-topic set)	2.75	3%	33%	52%	11%
	idf training	on-topic	124,426	66,526	319	2.74	3%	33%	52%	11%

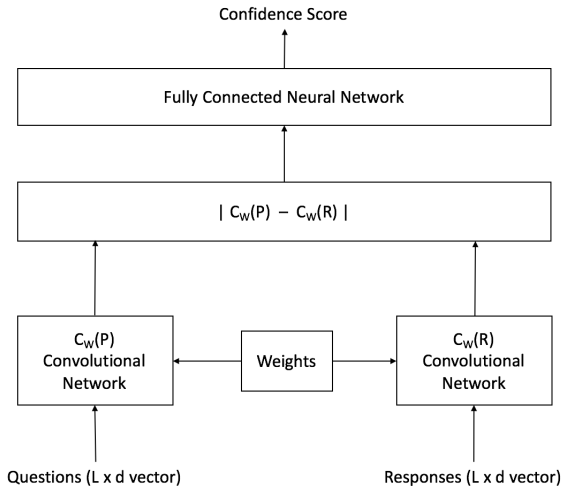

Figure 1: *Diagram of Siamese Convolutional Neural Network.*

resulted in four tfs for each question. The following five features measuring the lexical similarity between a test response and the representative responses for each proficiency level were generated:

- $cos1$, $cos2$, $cos3$, and $cos4$: the cosine similarity score between the test response and a score-specific VSM.
- $cosmax$: the score level of the VSM with the maximum similarity score; for instance, given the response, if $cos1$, $cos2$, $cos3$, and $cos4$ are 0.1, 0.2, 0.3, and 0.4 respectively, $cosmax$ is 4.

3.3. Siamese-CNN

Our model consisted of the three components shown in Figure 1: a sentence modeling layer using two convolutional neural networks, a similarity distance calculation layer, and a fully-connected neural network layer. The sentence modeling layer converted an input text into a representation for the following similarity layer and captured different granularties of information from input texts. The sentence modeling layer consisted of two convolutional neural networks (CNNs) which shared their weights. One CNN was for modeling questions and the other was for modeling responses. Each CNN contained three sequential sub-components: the conversion of input texts to embedding vectors, convolution filters and the max-poolings from filtered outputs. During the conversion to embedding vectors, tokenized input strings were converted to a 2D tensor using the Word2Vec [22] embedding vectors [1]. The converted vector had a shape of $L \times d$, where L was the maximum length of a sentence (i.e., 100 in this study) and d was the dimension of the Word2Vec (i.e., 300 in this study). Next, the converted vector was fed into convolution filters. Five filter sizes ranging from 3 to 7 were used. Each filter size corresponded to an n-gram order, i.e., tri-grams to 7-grams. Per filter size, 100 filters were used. Max pooling followed the filters in order to find the largest value from filter outputs.

Pairs of questions and responses went through the sentence modeling step and two vectors were generated. The two vectors went through the similarity layer to capture the semantic differences during training. In the similarity layer, the absolute difference was calculated. The formula of the difference was defined as $|C_w(P) - C_w(R)|$ where $C_w(P)$ and $C_w(R)$ were the representation vectors of questions and responses from the two CNNs. On top of the similarity layer, we stacked a fully-connected neural network containing one hidden layer (the vector size of the hidden layer was 40) with a *ReLU* activation layer in between, followed by a log-softmax layer as the final output layer. In the network, we applied 'dropout' regularization [23] (dropout rate was 0.5) to overcome overfitting. In addition, we used cross-entropy as the learning metric and Adadelta [24] ($\epsilon = 1e-08$, $\rho = 0.95$, learning rate is 1.0) as the optimizer. The final output of the fully-connected neural network was a confidence score which we converted into an on-topic or off-topic label based on a threshold of 0.5. The siamese-CNN was trained on the FM set [2]. The confidence score and the converted label were created using this model for both the FM train and the evaluation set. The siamese-CNN was implemented using Keras [25].

3.4. FM Training

We trained three different models to investigate the impact of various similarity features on off-topic detection: a model based on five features from response-based VSMs, a model based on both prediction label and confidence score from Siamese-CNNs, and finally a model based on the combination of both groups. In building the three models, we used the FM train set and the J48 algorithm (WEKA implementation of C4.5) of the WEKA machine learning toolkit [26]. The models were tested on the evaluation partition. The results are presented in Table 2.

[1] We downloaded it from `https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/edit?usp=sharing`

[2] During the training, the number of epoch was 10 and the batch size was 128. In addition, early stopping was applied based on the loss value changes. All the reported hyper parameters were selected using 10% of train data as a validation set.

4. Results

Table 2: *Performance of FMs in off-topic detection across all test questions.*

FM type	Acc.	Pre.	Rec.	F-score
Siamese-CNN	0.97	0.97	0.98	0.98
Response-based VSMs	0.94	0.93	0.95	0.94
Combination	0.98	0.97	0.98	0.98

We compared a decision model trained on two features (i.e., confidence scores and converted labels) from siamese-CNN to the model based on response-based VSMs. All results were computed on the evaluation partition containing 20,000 on-topic responses and 20,000 off-topic responses.

The response-based VSM models showed good performance. The accuracy was 94% and the recall was 2% higher than the precision. The model using siamese-CNN features showed 97% accuracy, which represents a 50% relative error reduction rate (from 6% errors to 3% errors) compared with the 94% accuracy of the model using response-based VSM features. 3% improvement in accuracy was significant using McNemar's test ($p < 0.005$). The recall of the siamese-CNN-based system was also 3% higher than the recall of the response-based VSMs, which was the same improvement as compared to the improvement in accuracy.

We also checked if we could get further improvement when we used all the features of both the response-based VSMs and the siamese-CNN. We could observe a 4% and 1% improvement in accuracy in the response-based VSMs and Siamese-CNN model, respectively. Both improvements were significant ($p < 0.005$ for the response-based VSMs and $p < 0.05$ for Siamese-CNN model based on McNemar's test). However, the precision and recall of the combined model was identical to the precision and recall of Siamese-CNN.

5. Discussion

When the models are used in evaluating test responses, it is important that the models show consistent performance throughout all questions. When a model shows fluctuations in performance across different questions, it may be challenging to deploy the model in an operational assessment, since another method may need to be developed to cover questions for which the model performs poorly. In order to examine the impact of different questions on off-topic detection, we calculated the accuracy for each question separately. Table 3 summarizes the average, standard deviation, and minimum accuracy across the 20 different questions contained in the evaluation set.

Table 3: *Question-specific accuracy of FMs*

	M	SD	Min	Max
Siamese-CNN	0.98	0.01	0.94	0.99
Response-based VSMs	0.94	0.04	0.83	0.99
Combination	0.98	0.01	0.94	0.99

As shown in the table, the accuracy of the response-based VSMs varied substantially across different questions, ranging from 0.83 to 0.99; the accuracy of the worst performing question was therefore 0.16 lower than the best performing question. Compared to the response-based VSMs, the siamese-CNN models showed relatively consistent performance across different questions; the lowest accuracy across the 20 different questions was 0.94, and the difference between the accuracy on the worst and the best questions was only $1/3$ of the difference for the response-based VSMs. Such consistent performance across different questions is an important advantage for the simaese-CNN approach. In this study, we used the approach (proficiency-level-specific tfs) used for automated content scoring to develop the response-based VSMs. This approach showed a promising performance in predicting proficiency level, but it was not optimized for off-topic response detection. In future study, we will further investigate the different approaches for training tfs.

For some binary classification tasks, only one type of instance (either positive or negative instances) are available, and other types of instances are rare or not available. In this study, negative instances (authentic off-topic responses) were rare and difficult to collect. Similar to previous studies such as [10, 3], we simulated negative instances by using responses elicited from different questions as off-topic responses. From a qualitative analysis of a small numbers of students' authentic off-topic responses, we found some evidence to support this approach for generating negative instances: some test takers did, in fact, appear to have used responses that had been prepared for other test questions.

However, in contrast to previous studies (e.g., [3]), we used a much larger number of questions in the set of off-topic responses. Although the number of questions we used is relatively large, it may still be substantially smaller than the number in an actual operational test, in which there may be no limitation on the number of topics for off-topic responses. In the future, we will investigate the impact of off-topic responses elicited using questions not covered in the training dataset. We will create a new set of evaluation data that includes these unseen off-topic responses, and we will investigate the impact on both siamese-CNN and response-based VSMs. Also, we will explore how to determine the optimal way to create the training dataset in order to minimize the impact of unseen off-topic responses. In this study, we only examined Decision Tree algorithm while training filtering models. Another future work will be the examination of other machine learning algorithms in the model training.

6. References

[1] S.-Y. Yoon and S. Xie, "Similarity-based non-scorable response detection for automated speech scoring," in *Proceedings of the 9th Workshop on Innovative Use of NLP for Building Educational Applications*, 2014, p. 116.

[2] A. Metallinou and J. Cheng, "Syllable and language model based features for detecting non-scorable tests in spoken language proficiency assessment applications," in *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 89–98. [Online]. Available: http://www.aclweb.org/anthology/W14-1811

[3] A. Malinin, R. C. Van Dalen, Y. Wang, K. M. Knill, and M. J. Gales, "Off-topic response detection for spontaneous spoken english assessment," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016, pp. 1075–1084.

[4] M. Sanderson, "Duplicate detection in the reuters collection,"

Technical Report (TR-1997-5) of the Department of Computing Science at the University of Glasgow G12 8QQ, UK, 1997.

[5] T. C. Hoad and J. Zobel, "Methods for identifying versioned and plagiarized documents," *Journal of the American society for information science and technology*, vol. 54, no. 3, pp. 203–215, 2003.

[6] N. Shivakumar and H. Garcia-Molina, "Scam: A copy detection mechanism for digital documents," 1995.

[7] S. Brin, J. Davis, and H. Garcia-Molina, "Copy detection mechanisms for digital documents," in *ACM SIGMOD Record*, vol. 24, no. 2. ACM, 1995, pp. 398–409.

[8] K. Evanini and X. Wang, "Automatic detection of plagiarized spoken responses," in *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 22–27. [Online]. Available: http://www.aclweb.org/anthology/W14-1803

[9] D. Higgins, J. Burstein, and Y. Attali, "Identifying off-topic student essays without topic-specific training data," *Natural Language Engineering*, vol. 12, no. 02, pp. 145–159, 2006.

[10] A. Louis and D. Higgins, "Off-topic essay detection using short prompt texts," in *Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, 2010, pp. 92–95.

[11] B. Hu, Z. Lu, H. Li, and Q. Chen, "Convolutional neural network architectures for matching natural language sentences," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2042–2050.

[12] K. S. Tai, R. Socher, and C. D. Manning, "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)*. Beijing, China: Association for Computational Linguistics, July 2015, pp. 1556–1566. [Online]. Available: http://www.aclweb.org/anthology/P15-1150

[13] W. Yin and H. Schütze, "Convolutional neural network for paraphrase identification," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, May–June 2015, pp. 901–911. [Online]. Available: http://www.aclweb.org/anthology/N15-1091

[14] X. Zhu, P. Sobihani, and H. Guo, "Long short-term memory over recursive structures," in *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, D. Blei and F. Bach, Eds. JMLR Workshop and Conference Proceedings, 2015, pp. 1604–1612. [Online]. Available: http://jmlr.org/proceedings/papers/v37/zhub15.pdf

[15] J. Mueller and A. Thyagarajan, "Siamese recurrent architectures for learning sentence similarity," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, ser. AAAI'16. AAAI Press, 2016, pp. 2786–2792. [Online]. Available: http://dl.acm.org/citation.cfm?id=3016100.3016291

[16] H. He, K. Gimpel, and J. Lin, "Multi-perspective sentence similarity modeling with convolutional neural networks," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, September 2015, pp. 1576–1586. [Online]. Available: http://aclweb.org/anthology/D15-1181

[17] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a "siamese" time delay neural network," in *Advances in Neural Information Processing Systems 6*, J. D. Cowan, G. Tesauro, and J. Alspector, Eds. Morgan-Kaufmann, 1994, pp. 737–744. [Online]. Available: http://papers.nips.cc/paper/769-signature-verification-using-a-siamese-time-delay-neural-network.pdf

[18] M. Rei and R. Cummins, "Sentence similarity measures for fine-grained estimation of topical relevance in learner essays," in *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, 2016, pp. 283–288.

[19] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[20] J. Tao, S. Ghaffarzadegan, L. Chen, and K. Zechner, "Exploring deep learning architectures for automatically grading non-native spontaneous speech," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 6140–6144.

[21] S. Xie, K. Evanini, and K. Zechner, "Exploring content features for automated speech scoring," in *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2012, pp. 103–111.

[22] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[23] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014. [Online]. Available: http://dl.acm.org/citation.cfm?id=2627435.2670313

[24] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012. [Online]. Available: http://arxiv.org/abs/1212.5701

[25] F. Chollet, "Keras," https://github.com/fchollet/keras, 2015.

[26] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.