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

Off-topic Spoken Response Detection with Word Embeddings

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

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

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

Abstract

In this study, we developed an automated off-topic response detection system as a supplementary module for an automated proficiency scoring system for non-native English speakers' spontaneous speech. Given a spoken response, the system first generates an automated transcription using an ASR system trained on non-native speech, and then generates a set of features to assess similarity to the question. In contrast to previous studies which required a large set of training responses for each question, the proposed system only requires the question text, thus increasing the practical impact of the system, since new questions can be added to a test dynamically. However, questions are typically short and the traditional approach based on exact word matching does not perform well. In order to address this issue, a set of features based on neural embeddings and a convolutional neural network (CNN) were used. A system based on the combination of all features achieved an accuracy of 87% on a balanced dataset, which was substantially higher than the accuracy of a baseline system using question-based vector space models (49%). Additionally, this system almost reached the accuracy of vector space based model using a large set of responses to test questions (93%).

1. Introduction

This study aims to develop an off-topic detection system as a part of an automated oral proficiency scoring system. The automated scoring system was designed to score spoken responses to a test of English speaking proficiency. When students are fatigued, unmotivated, distracted, they may not respond seriously. For instance, students may recite their response to a previous question (referred to as *off-topic* responses hereafter). Such responses often have sub-optimal characteristics which make it difficult for the automated scoring system to provide a valid score. In order to address this issue, the automated scoring system can employ a "filtering model" (hereafter, FM) to filter out off-topic responses. By filtering out such problematic responses, the remaining responses can be scored by the automated scoring system without concerns about scoring errors resulting from problematic responses.

Filtering off-topic responses is concerned with issues related to topicality. However, these issues are found at different ranks on [1]'s hierarchy of five similarity levels (unrelated, on the general topic, on the specific topic, same facts, and copied). In particular, off-topic responses belong to the *unrelated* group. In this study, we focus on off-topic responses and develop an automated FM which detects off-topic responses by utilizing semantic similarity measures. Especially, we use only the question text and do not use sample responses for test questions.

With the introduction of the FM, the overall architecture of our automated scoring system will be as follows. For a given spoken response, the system performs speech recognition and speech processing. Given the ASR output and the speech signal, it computes a set of linguistic features assessing pronunciation, prosody, vocabulary, and grammar skills. In addition, document similarity features are generated based on word hypotheses and content models. The FM then uses the similarity features to filter out off-topic responses. Finally, the remaining responses are scored by the automated scoring model. In this study, we will only focus on the FM part of the overall architecture.

2. Relevant studies

Previous studies, such as [2, 3, 4], focused on scoring of highly restricted speech (e.g., read-aloud) and detected off-topic responses using features derived from the automated speech recognition (ASR) system. This approach achieved good performance for restricted speech, but it is not appropriate for tasks that elicit unconstrained, spontaneous speech.

[5] applied document similarity features to detect gaming responses for an English speaking proficiency test that elicits spontaneous speech from non-native speakers. They developed a set of similarity features between a test response and a large number of question-specific responses (sample responses provided to the same question as the test response) using VSM (vector space model) and word overlaps. These features were used in identifying gaming responses with topic problems (e.g., question repetition and off-topic responses) and showed promising performance.

Approaches like those above require a sizable amount of response data for each question, and collecting question-specific data is an expensive and difficult task. To address this issue, [6] developed a system for detecting off-topic essays without the need for question-specific responses; the system was based on similarity features between the question text and the test response. The performance of this system was lower than the benchmark system trained on question-specific responses, but it achieved a substantial improvement over a majority-based baseline. [7] further improved this system by expanding question texts to include synonyms, inflected forms, and distributionally similar words to the question content. The performance of [7] showed a substantial improvement for questions consisting of only a small amount of text.

More recently, various approaches based on deep-neural networks (DNN) and word-embeddings trained on large corpora have showed promising performance in document similarity detection (e.g., [8, 9, 10]). In contrast to traditional similarity features, which are limited to a reliance on exact word matching (e.g., content vector analysis), these new approaches have the advantage of capturing topically relevant words that

http://dx.doi.org/10.21437/Interspeech.2017-388

are not identical. [11] and [12] applied this approach to the task of off-topic detection in spoken responses and essays, respectively, and achieved substantial improvements over systems using only word-matching. Based on the success of these previous studies, we will apply various DNN-based approaches and word-embeddings for off-topic spoken response detection in the context of automated speech scoring. Notably, there are large differences in length between the input pairs (i.e., the question text and a spoken response), and also among the test responses. In order to address this issue, we also explore methods that are efficient in handling differences in input length.

3. Data

We used a collection of spoken responses from an assessment of English 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 experience or background knowledge. The question texts were short and composed of fewer than four sentences. The number of words in each question text ranged from 17 to 48, and the average and standard deviation of word length were 33.3 and 8.7, respectively.

A dataset comprised of 60,000 responses was used for the training and evaluation of off-topic FMs (hereafter, FM set). First, 20 questions covering diverse topics were selected, and 30,000 responses were elicited using them (hereafter, on-topic responses). Since we did not have a large set of authentic off-topic responses collected from actual administrations of the test, we used students' responses elicited from different questions. For this purpose, 50 questions that did not overlap with on-topic questions were selected, and 30,000 responses to those questions were selected. The FM set was further partitioned into 20 folds, and each fold included responses to one on-topic question and randomly selected off-topic responses. Each fold was balanced in that it consisted of 1,500 on-topic and 1,500 off-topic responses.

The responses contained 106.0 words on average, but there was substantial variation in length among the responses, with word counts ranging from 1 to 218. Responses were rated by trained human raters using a 4-point scoring scale, where 1 indicated low speaking proficiency and 4 indicated high speaking proficiency. The raters gave a score of 0 when test takers did not show any intention to directly respond to the question. The majority of the zero score responses were blank responses. Finally, the raters also labeled responses as TD (technical difficulty) when responses contained technical issues that were substantial enough to make it impossible to provide a valid score by a human rater (e.g., background noise or audio distortion). These TD and 0 responses were excluded from the dataset since the current study is focused on the detection of topicality issues. The speakers, question information, and the average proficiency score for FM set are presented in Table 1.

4. Method

4.1. Overview

In this study, we used various features that assess document similarity for off-topic response detection. As a baseline system, we trained a $tf{\cdot}idf$ weighted vector space model using only question texts and a large set of responses that do not include any question-specific responses used in the evaluation set. Next, we developed a set of features based on word-embeddings and neural networks. Finally, we trained response-based VSMs as a benchmark system using the question-specific-response dataset.

4.2. Question-based VSMs

We trained a VSM for each question separately, since the topic of each question was unique. The question text was converted into a single vector, and tf was trained only using this vector. We collected 125,000 responses elicited from 319 questions and used it to calculate an idf. The dataset covered a wide range of questions except the questions used in FM set.

4.3. Weighted embeddings

Following [12]'s approach, we created word-embedding-based features using a publicly available word embedding vectors trained on the Google News corpus by [13]. It contains 300-dimensional vectors for 3 million unique words and phrases. The following two features were generated:

- averaged word embeddings: We created a vector for each question by mapping each word in the question text to a corresponding word embedding vector and averaging them. Next, we created a vector for a test response using the same process. Finally, we calculated the cosine similarity between the question vector and the response vector.
- idf weighted word embeddings: we calculated an idf weighted word embedding feature by scaling each word embedding vector by the corresponding idf weight and averaging the scaled vectors. We calculated the cosine similarity between these weighted vectors.

4.4. Word Mover's Distance (WMDist)

Word mover's distance (WMDist; [8]) is a distance measure between two documents based on word-embeddings. When the embeddings of each word are represented as a vector, the distance between two words can be measured using the Euclidean distance between the two corresponding word vectors in the embedding space. WMDist represents the sum of the minimum values among the Euclidean word distances between words in the two compared documents (a source and a target). For each word in a source, WMDist algorithm first selects a word with a minimum Euclidean distance from the target then sums up their distances. This minimization problem is a special case of Earth Mover's Distance ([14, 15]) (hence the name), for which efficient algorithms are available. [8] report that WMDist outperformed other distance measures on document retrieval tasks, and that the embeddings trained on the Google News corpus consistently performed well across a variety of contexts. WMDist does not rely on exact word matching, and therefore, at least theoretically, can be more robust against the potential inflation of document similarity when a long document with a large number of content words is compared to a short document. WMDist was, therefore, well-suited as a distance measure between responses and question texts, which often differ in length.

For this experiment, we used the same word embeddings used in weighted embedding features as the input for the WMDist calculation. We first deleted stop words in the responses and the questions. We then calculated the WMDist between responses and questions using the WMDist implementation in the gensim package[16]. Each response was compared

Table 1: *Number of responses and distribution of test questions in the on-topic and off-topic FM sets*

Description	# Responses	# Speakers	# Questions	Average Score
On-topic	30,000	15,000	20	2.75
Off-topic	30,000	15,000	50	2.72

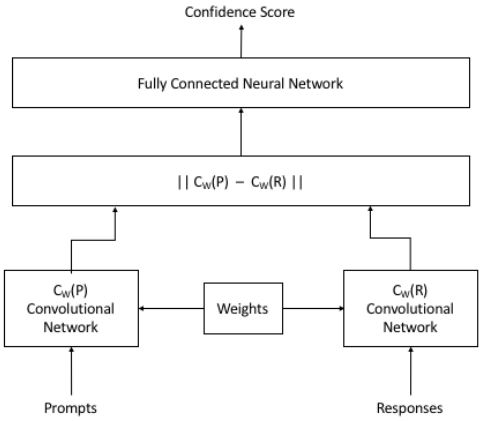

Figure 1: *Diagram of Siamese Convolutional Neural Network.*

against its question text, resulting in one WMDist value for every response. For each word in a question, WMDist algorithm selected a word with a minimum Euclidean distance from the response and summed up their distances.

4.5. Siamese-CNN

Successful applications of Siamese networks to the task of image matching [17] has led to the application of the algorithm to similar problems in natural language processing. It has been applied to the detection of similar sentences and has shown competitive performance [9, 10, 18]. Therefore, we also applied a Siamese network to our off-topic detection task. We assumed that the similarity distance between an on-topic response and its question would be closer than the similarity distance between an off-topic response and its question.

Our siamese network consisted of three main components as shown in Figure 1: two convolutional networks with shared weights, a layer for similarity calculation, and a fully connected neural network. The twin convolutional networks shared parameters although the networks received two distinct inputs (i.e., test questions and responses). We followed [19] in implementing the convolutional networks. An advantage of the shared parameters is that questions and responses are mapped to the same feature space. In the layer for similarity calculation, element-wise absolute differences between output vectors of the twin CNNs were calculated. The absolute differences were passed into the fully connected neural network. The last layer of the fully connected neural network was a sigmoid layer for binary classes. We borrowed this idea from [20], in order to learn the semantic differences from reduced feature vectors of CNNs during training. We used the FM set to train the network since it requires both on-topic and off-topic responses. Using the 20 folds described in Section 3, we trained the network using 19 folds and generated features on the remaining fold.

4.6. Response-based VSMs (response-VSM)

We trained tf models for the 20 questions contained in the FM on-topic set. For this purpose, we collected an additional 20,000 responses (1,000 responses per question) and trained distinct tf models for each question. Assuming that the responses with the highest proficiency scores contain the most diverse and appropriate words related to the topic, we only selected responses with a score of 4. We obtained the ASR-based transcriptions of the responses, and all responses to the same question were converted into a single vector. In this study, the term was a word unigram and the document was the response. We used the same idf weights that were trained for Question-based VSMs.

5. Experiments

A gender independent acoustic model (AM) was built with 800 hours of speech extracted from the same English oral proficiency test, using the Kaldi toolkit [21]. The AM training dataset contained 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 language model trained using the same dataset used for AM training. This ASR system achieved a Word Error Rate of 23% on the 600 held-out responses. Detailed information about the ASR system is provided in [22]. Word hypotheses were generated for each response in the dataset described in Table 1, and the set of features described in Section 4 were generated based on these ASR transcriptions.

Decision tree models were trained to predict binary values (on-topic and off-topic) using the FM set and the J48 algorithm (WEKA implementation of C4.5) in the WEKA machine learning toolkit [23]. We conducted 20-fold cross-validation using the FM set partitioned into 20 folds as described in 3. The model was trained on 19 sets and evaluated on the remaining set. There was no overlap in on-topic questions between the train and evaluation sets. In this way, the FM was trained without using responses for questions used in the evaluation set; this scenario thus shows the system performance that would be expected when both content models and FMs are not updated for new test questions that are introduced to the assessment. We report the average of the 20 folds in the result section.

In order to compare the impact of various similarity features on the quality of off-topic response detection, we first trained four different models by including only one set of similarity features: question-based VSM feature (baseline), Weighted embedding, WMDist, Siamese-CNN. These features were based only on question texts without the question-specific response set. Hereafter, we will refer to them as question-based features. Finally, we trained a new model combining all question-based features. The results are presented in Table 2.

As a benchmark comparison, we trained a model using the response-based VSM features using the question-specific response set. These results are presented in the last row of Table 2.

6. Results

We report the performance of models in terms of accuracy, precision, recall, and F-score for detecting off-topic responses. In this study, the accuracy of the majority class baseline (classifying all responses as on-topic responses) was 50% since the proportion of on-topic and off-topic responses was balanced.

Table 2: *Performance of FMs in off-topic detection*

Feature Set	**Acc.**	**Pre.**	**Rec.**	**F-score**
Question-VSMs	0.489	0.518	0.861	0.606
Weighted Embeddings	0.724	0.743	0.885	0.775
WMDist	0.818	0.838	0.855	0.822
Siamese-CNN	0.818	0.844	0.800	0.808
All	0.869	0.875	0.894	0.874
Response-VSMs	0.932	0.928	0.939	0.932

The results in Table 2 show that the word-embedding and DNN-based models outperformed the question VSM-based model. Both the accuracy and F-score of the question VSM-based model were low and the accuracy was close to the majority baseline. Among the neural embedding-based models, both the Siamese-CNN-based model and the WMDist-based model achieved good performance with accuracies of 82%. The combination of all features resulted in further improvement, with both accuracy and F-score around 0.87.

Finally, we compared the model based on the combination of all question-based features (*All*) with the response VSM-based model, for which both accuracy and F-score were around 0.93. While *All* question-based feature model did not achieve a better performance than the response-based VSM model, the results are still encouraging since it did not make use of any question-specific responses and achieved an accuracy of 87%.

In order to use models in actual operational assessments, it is important that the models show consistent performance throughout all questions. 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. In this table, we only used our best performing system (*All*) and the benchmarking system (*Response-based VSMs*).

Table 3: *Question-specific accuracy of FMs*

	M	SD	Min	Max
All	0.869	0.081	0.660	0.940
Response-based VSMs	0.932	0.043	0.820	0.990

The accuracy of the proposed system (*All*) varied substantially across different questions. The standard deviation (0.081) was almost twice larger than the standard deviation of the response-based VSMs (0.043). This is a critical challenge for the system to be deployed in the operational assessment, and we will further need to investigate how to reduce this performance fluctuation across different questions. We will further investigate this issue in our future study.

7. Conclusions

In this study, an off-topic response detection system was developed for an automated speech proficiency scoring system. In order to provide a system that can be scaled efficiently in an operational assessment, the model was trained without any test taker responses for new questions. The combination of similarity features based on VSMs, neural-embeddings, and CNN resulted in a high-performing detection module. However, due to lack of a large set of authentic off-topic responses, we instead used a large set of responses to different questions. In further examination of this topic, we need to collect authentic off-topic responses and use them for system development. In order to draw more confident conclusions it is especially important to have authentic data for system evaluation. Moreover, we included the same amount of off-topic responses as on-topic responses, but the percentage of these responses is likely to be substantially lower in the authentic test situation. Therefore, we need to develop an automated system that can achieve high performance in a sparse distribution of off-topic responses. We will further investigate these questions in future study.

8. References

[1] D. Metzler, Y. Bernstein, W. B. Croft, A. Moffat, and J. Zobel, "Similarity measures for tracking information flow," in *Proceedings of the 14th ACM international conference on Information and knowledge management*. ACM, 2005, pp. 517–524.

[2] J. van Doremalen, H. Strik, and C. Cucchiarini, "Utterance verification in language learning applications," in *Proceedings of the SLaTE*, 2009.

[3] W.-K. Lo, A. M. Harrison, and H. Meng, "Statistical phone duration modeling to filter for intact utterances in a computer-assisted pronunciation training system," in *Proceedings of Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, 2010, pp. 5238–5241.

[4] J. Cheng and J. Shen, "Off-topic detection in automated speech assessment applications," in *Proceedings of InterSpeech*, 2011, pp. 1597–1600.

[5] 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.

[6] 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.

[7] 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.

[8] M. Kusner, Y. Sun, N. Kolkin, and K. Q. Weinberger, "From word embeddings to document distances," in *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, 2015, pp. 957–966.

[9] 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

[10] P. Neculoiu, M. Versteegh, and M. Rotaru, "Learning text similarity with siamese recurrent networks," in *Proceedings of the 1st Workshop on Representation Learning for NLP*. Berlin, Germany: Association for Computational Linguistics, August 2016, pp. 148–157. [Online]. Available: http://anthology.aclweb.org/W16-1617

[11] 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.

[12] 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.

[13] 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.

[14] G. Monge, "Mémoire sur le calcul intégral des équations aux différences partielles," *Histoire de l'Académie Royale des Sciences*, pp. 118–192, 1784.

[15] Y. Rubner, C. Tomasi, and L. J. Guibas, "A metric for distributions with applications to image databases," in *Proceedings of the Sixth International Conference on Computer Vision*, ser. ICCV '98. Washington, DC, USA: IEEE Computer Society, 1998, pp. 59–. [Online]. Available: http://dl.acm.org/citation.cfm?id=938978.939133

[16] R. Rehurek and P. Sojka, "Software framework for topic modelling with large corpora," in *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Citeseer, 2010.

[17] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML Deep Learning Workshop*, 2015. [Online]. Available: https://sites.google.com/site/deeplearning2015/37.pdf?attredirects=0

[18] 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.

[19] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, October 2014, pp. 1746–1751.

[20] T. Rama, "Siamese convolutional networks for cognate identification," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan: The COLING 2016 Organizing Committee, December 2016, pp. 1018–1027.

[21] 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.

[22] 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.

[23] 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.