Scoring Spoken Responses Based on Content Accuracy

Fei Huang
CS Dept. Temple Univ.
Philadelphia, PA, 19122
tub58431@temple.edu

Lei Chen
Educational Testing Service (ETS)
Princeton, NJ, 08541
lchen@ets.org

Jana Sukkarieh
ETS
JSukkarieh@ets.org

Abstract

Accuracy of content have not been fully utilized in the previous studies on automated speaking assessment. Compared to writing tests, responses in speaking tests are noisy (due to recognition errors), full of incomplete sentences, and short. To handle these challenges for doing content-scoring in speaking tests, we propose two new methods based on information extraction (IE) and machine learning. Compared to using an ordinary content-scoring method based on vector analysis, which is widely used for scoring written essays, our proposed methods provided content features with higher correlations to human holistic scores.

1 Introduction

In recent years, there is an increasing interest of using speech processing and natural language processing (NLP) technologies to automatically score speaking tests (Eskenazi, 2009). A set of features related to speech delivery, such as fluency, pronunciation, and intonation, has been utilized in these studies. However, accuracy of an answer's content to the question being asked, important factors to be considered during the scoring process, have not been fully utilized. In this paper, we will report our initial efforts exploring content scoring in an automated speaking assessment task. To start, we will briefly describe the speaking test questions in our research.

In the test we used for evaluation, there were two types of questions. The first type, *survey*, requires a test-taker to provide answers specific to one or several key points in a survey question without any background reading/listening related to the topic of the survey. Typical questions could be "*how frequently do you go shopping?*" or "*what kind of products did you purchase recently?*" In contrast, the second type, *opinion*, requires a test-taker to speak as long as 60 seconds to present his or her opinions about some topic. An example of such questions could be, "*Do you agree with the statement that online shopping will be dominant in future or not?*" Compared to the essays in writing tests, these spoken responses could just be incomplete sentences. For example, for the *survey* questions, test-takers could just say several words. For the questions described above, some test-takers may just use phrases like "once a week" or "books". In addition, given short responding durations, the number of words in test-takers' responses is limited. Furthermore, since scoring speech responses requires speech recognition, more noisy inputs are expected. To tackle these challenges, we propose two novel content scoring methods in this paper.

The remainder of the paper is organized as follows: Section 2 reviews the related previous research efforts; Section 3 proposes the two content-scoring methods we designed for two types of questions described above; Section 4 reports the experimental results of applying the proposed methods; finally, Section 5 concludes our reported research and describes our plans for future research.

2 Related Work

For writing tests, previous content scoring investigations can be divided into the following three groups. The first group relies on obtaining and matching patterns associated with the correct answers (Leacock and Chodorow, 2003; Sukkarieh and Blackmore, 2009).

The second group of methods, also mostly used

The 7th Workshop on the Innovative Use of NLP for Building Educational Applications, pages 122–126,
Montréal, Canada, June 3-8, 2012. ©2012 Association for Computational Linguistics

for content-scoring, is to rely on a variety of text similarity measurements to compare a response with either pre-defined correct answers or a group of responses rated with a high score (Mohler and Mihalcea, 2009). Compared to the first group, such methods can bypass a labor intensive pattern-building step. A widely used approach to measuring text similarity between two text strings is to convert each text string into a word vector and then use the angle between these two vectors as a similarity metric. For example, Content Vector Analysis (CVA) has been successfully utilized to detect off-topic essays (Higgins et al., 2006) and to provide content-related features for essay scoring (Attali and Burstein, 2004). For this group of methods, measuring the semantics similarity between two terms is a key question. A number of metrics have been proposed, including metrics (Courley and Mihalcea, 2005) derived from WordNet, a semantics knowledge database (Fellbaum, 1998), and metrics related to terms' co-occurrence in corpora or on the Web (Turney, 2001).

The third group of methods treats content scoring as a Text Categorization (TC) task, which treats the responses being scored on different score levels as different categories. Therefore, a large amount of previous TC research, such as the many machine learning approaches proposed for the TC task, can be utilized. For example, Furnkranz et al. (1998) compared the performance of applying two machine learning methods on a web-page categorization task and found that the Repeated Incremental Pruning to Produce Error Reduction algorithm (RIPPER) (Cohen, 1995) shows an advantage concerning the feature sparsity issue.

3 Methodology

As described in Section 1, for the two types of questions considered, the number of words appearing in a response is quite limited given the short response time. Therefore, compared to written essays, when applying the content-scoring methods based on vector analysis, e.g., CVA, feature sparsity becomes a major factor negatively influencing the performance of these methods. Furthermore, there are more challenges when applying vector analysis on *survey* questions because test-takers could just use words/phrases rather than completed sentences. Also, some *survey* questions could have a very large range of correct answers. For example, if a question is about the name of a book, millions of book titles could be potential answers. Therefore, a simple phrase-matching solution cannot work.

3.1 Semi-Automatic Information Extraction

For *survey* responses, the answers should be related to the key points mentioned in the questions. For example, for the question, "*What kind of TV programs do you like to watch?*", possible correct answers should be related to TV programs. Moreover, it should be the instances of specific TV programs, like news, comedy, talk shows, etc. Note that the acceptable answers may be infinite, so it is not realistic to enumerate all possible answers. Therefore, we proposed a method to extract the potential answer candidates and then measure their semantic similarities to the answer keys that could be determined manually. In particular, the answer keys were determined by the first author based on her analysis of the test prompts. For example, for the question "*What kind of books do you like to read?*", two answer keys, "book" and "reading" were selected. After a further analysis of the questions, we found that most of the *survey* questions are about "when" "where" and "what", and the answers in the responses were usually nouns or noun phrases. Therefore, we decided to extract the noun phrases from each response and use them as potential candidates.

We use two semantic similarity metrics (SSMs) to evaluate how each candidate relates to an answer key, including PMI-IR (Turney, 2001) and a word-to-word similarity metric from WordNet (Courley and Mihalcea, 2005). The PMI-IR is a measure based on web query analysis using Pointwise Mutual Information (PMI) and Information Retrieval (IR). For an answer candidate (c) and an answer key (k), their PMI-IR is computed as:

$$SSM_{\text{PMI-IR}}(c,k) = \frac{hits(c\text{NEAR}k)}{hits(c)}$$

where the $hits(x)$ function obtains the count of term x returned by a web search engine and **NEAR** is a query operator for proximity search, searching the pages on which both k and c appear within a specified distance. Among many WordNet (WN) based SSMs summarized in Courley and Mihalcea (2005), we found that the Wu-Palmer metric proposed by Wu and Palmer (1994) worked the best in our pilot study. This metric is a score denoting how similar two word senses are, based on the depth of the two word senses in the taxonomy and their Least Common Subsumer [1] (LCS):

$$SSM_{\text{WN}}(c,k) = \frac{2 * depth(LCS)}{depth(c) + depth(k)}$$

For each answer key, we calculated two sets of SSMs ($SSM_{\text{PMI-IR}}$ and SSM_{WN}, respectively) from all candidates. Then, we selected the largest $SSM_{\text{PMI-IR}}$ and SSM_{WN} as the final SSMs for this particular answer key. For each test question, using the corresponding responses in the training set, we built a linear regression model between these SSMs for all answer keys and the human judged scores. The learned regression model was applied to the responses to this particular testing question in the testing set to convert a set of SSMs to predictions of human scores. The predicted scores were then used as a content feature. Since answer keys were determined manually, we refer to this method as semi-automatic information extraction (Semi-IE).

3.2 Machine Learning Using Smoothed Inputs

For the *opinion* responses, inspired by Furnkranz et al. (1998), we decided to try sophisticated machine learning methods instead of the simple vector-distance computation used in CVA. Due to short response-time in the speaking test being considered, the ordinary vector analysis may face a problem that the obtained vectors are too short to be reliably used. In addition, using other non-CVA machine learning methods can enable us to try other types of linguistic features. To address the feature sparsity issue, a smoothing method, which converts word-based text features into features based on other entities with a much smaller vocabulary size, is used. We use a Hidden Markov Model (HMM) based smoothing method (Huang and Yates, 2009), which induces classes, corresponding to hidden states in the HMM model, from the observed word strings. This smoothing method can use contextual information of the word sequences due to the nature of HMM.

Then, we convert word-entity vectors to the vectors based on the induced classes. TF-IDF (term frequency and inverse document frequency) weighting is applied on the new class vectors. Finally, the processed class vectors are used as input features (smoothed) to a machine learning method. In this research, after comparing several widely used machine learning approaches, such as Naive Bayes, CART, etc., we decided to use RIPPER proposed by Cohen (1995), a rule induction method, similar to Furnkranz et al. (1998).

4 Experiments

Our experimental data was from a test for international workplace English. Six testing papers were used in our study and each individual test contains three *survey* questions (1, 2, and 3) and two *opinion* questions (4 and 5). Table 1 lists examples for these question types. From the real test, we collected spoken responses from a total of $1,838$ test-takers. $1,470$ test-takers were used for training and 368 were used for testing. Following scoring rubrics developed for this test by considering speakers' various language skill aspects, such as fluency, pronunciation, vocabulary, as well as content accuracy, the *survey* and *opinion* responses were scored by a group of experienced human raters by using a 3-point scale and a 5-point scale respectively. For the *survey* responses, the human judged scores were centered on 2; for the *opinion* responses, the human judged scores were centered on 3 and 4.

Qs.	Example
1	*How frequently do you go shopping?*
2	*What kinds of products do you buy often?*
3	*How should retailers improve their services?*
4	*Make a purchase decision based on the chart provided and justify your decision.*
5	*Do you agree with the statement that online shopping will be dominant in the future or not? Please justify your point.*

Table 1: Examples of the five kinds of questions investigated in the study

All of these non-native speech responses were manually transcribed. A state-of-the-art HMM Automatic Speech Recognition (ASR) system which was trained from a large set of non-native speech data was used. For each type of test question, acoustic and language model adaptations were applied to further lower the recognition error rate. Finally,

[1] Most specific ancestor node

a word error rate around 30% to 40% could be achieved on the held-out speech data. In our experiments, we used speech transcriptions in the model training stage and used ASR outputs in the testing stage. Note that we decided to use speech transcriptions, instead of noisy ASR outputs that match to the testing condition, to make sure that the learned content-scoring model are based on correct word entities related to content accuracy.

For the *survey* responses, we manually selected the key points from the testing questions. Then, using a Part-Of-Speech (POS) tagger and a sentence chunker implemented by using the OpenNLP [2] toolkit, we found all possible nouns and noun-phrases that could serve as answer candidates and applied the Semi-IE method described in Section 3.1. For *opinion* questions, based on Huang and Yates (2009), we used 80 hidden states and applied the method described in Section 3.2 for content scoring. We used JRip, a Java implementation of the RIPPER (Cohen, 1995) algorithm in the Weka (Hall et al., 2009) machine learning toolkit, in our experiments.

When measuring performance of content-related features, following many automated assessment studies (Attali and Burstein, 2004; Leacock and Chodorow, 2003; Sukkarieh and Blackmore, 2009), we used the Pearson correlation r between the content features and human scores as an evaluation metric. We compared the proposed methods with a baseline method, CVA. It works as follows: it first groups all the training responses by scores, then it calculates a TF vector from all the responses under a score level. Also, an IDF matrix is generated from all the training responses. After that, for each testing response, CVA first converts it into a TF-IDF vector and then calculates the cosine similarity between this vector with each score-level vector respectively and uses the largest cosine similarity as the content feature for that response. The experimental results, including content-features' correlations r to human scores from each proposed method and the correlation increases measured on CVA results, are shown in Table 2. First, we find that CVA, which is designed for scoring lengthy written essays, does not work well for the *survey* questions, especially on first two questions, which are mostly phrases (not completed sentences). By contrast, our proposed Semi-IE method can provide more informative content measurements, indicated by substantially increased r. Second, CVA works better on *opinion* questions than on *survey* questions. This is because that *opinion* questions can be treated as short spoken essays and therefore are closer to the data on which the CVA method was originally designed to work. However, even on such a well-performing CVA baseline, the HMM smoothing method allows the Ripper algorithm to outperform the CVA method in content-features' correlations to human scores. For example, on question 4, on which either a table or a chart has been provided to test-takers, the CVA achieves a r of 0.47. The proposed method can still improve the r by about 15%.

Question	r_{CVA}	$r_{Semi-IE}$	$r \Uparrow$
1	0.12	0.30	150%
2	0.15	0.27	80%
3	0.21	0.26	23.8%
Question	r_{CVA}	$r_{Ripper_{HMM}}$	$r \Uparrow$
4	0.47	0.54	14.89%
5	0.33	0.39	18.18%

Table 2: Comparisons of the proposed content-scoring methods with CVA on *survey* and *opinion* responses

5 Conclusions and Future Works

In this paper, we proposed two content-scoring methods for the two types of test questions in an automated speaking assessment task. For particular properties of these two question types, we utilized information extraction (IE) and machine learning technologies to better score them on content accuracy. In our experiments, we compared these two methods, Semi-IE and machine learning using smoothed inputs, with an ordinary word-based vector analysis method, CVA. The content features computed using the proposed methods show higher correlations to human scores than what was obtained by using the CVA method.

For the Semi-IE method, one direction of investigation will be how to find the expected answer keys automatically from testing questions. In addition, we will investigate better ways to integrate many se-

[2] `http://opennlp.sourceforge.net`

mantic similarly measurements (SSMs) into a single content feature. For the machine learning approach, inspired by Furnkranz et al. (1998), we will investigate how to use some linguistic features related to response structures rather than just TF-IDF weights.

References

Y. Attali and J. Burstein. 2004. Automated essay scoring with e-rater v.2.0. In *Presented at the Annual Meeting of the International Association for Educational Assessment*.

W. Cohen. 1995. Text categorization and relational learning. In *In Proceedings of the 12th International Conference on Machine Learning*.

C. Courley and R. Mihalcea. 2005. Measuring the semantic similarity of texts. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, pages 13–18.

M. Eskenazi. 2009. An overview of spoken language technology for education. *Speech Communication*, 51(10):832–844.

C. Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

J. Furnkranz, T. Mitchell, and E. Riloff. 1998. A case study in using linguistic phrases for text categorization on the WWW. In *Proceedings from the AAAI/ICML Workshop on Learning for Text Categorization*, page 512.

M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H Witten. 2009. The WEKA data mining software: An update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.

D. Higgins, J. Burstein, and Y. Attali. 2006. Identifying off-topic student essays without topic-specific training data. *Natural Language Engineering*, 12.

F. Huang and A. Yates. 2009. Distributional representations for handling sparsity in supervised sequence-labeling. In *Proceedings of ACL*.

C. Leacock and M. Chodorow. 2003. C-rater: Automated scoring of short-answer questions. *Computers and the Humanities*, 37(4):385–405.

M. Mohler and R. 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*, pages 567–575.

J. Z. Sukkarieh and J. Blackmore. 2009. c-rater: Automatic content scoring for short constructed responses. In *Paper presented at the Florida Artificial Intelligence Research Society (FLAIRS) Conference, Sanibel, FL.*

P. D. Turney. 2001. Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In *Procs. of the Twelfth European Conference on Machine Learning (ECML)*, pages 491–502, Freiburg, Germany.

Z. Wu and M. Palmer. 1994. Verbs semantics and lexical selection. In *Proceeding ACL '94 Proceedings of the 32nd annual meeting on Association for Computational Linguistics*.