A Human-Computer Collaboration Approach to Improve Accuracy of an Automated English Scoring System

Jee Eun Kim
Hankuk University of Foreign Studies
Seoul, Korea
jeeeunk@hufs.ac.kr

Kong Joo Lee
Chungnam National University
Daejeon, Korea
kjoolee@cnu.ac.kr

Abstract

This paper explores an issue of redundant errors reported while automatically scoring English learners' sentences. We use a human-computer collaboration approach to eliminate redundant errors. The first step is to automatically select candidate redundant errors using PMI and RFC. Since those errors are detected with different IDs although they represent the same error, the candidacy cannot be confirmed automatically. The errors are then handed over to human experts to determine the candidacy. The final candidates are provided to the system and trained with a decision tree. With those redundant errors eliminated, the system accuracy has been improved.

1 Introduction

An automated English scoring system analyzes a student sentence and provides a score and feedback to students. The performance of a system is evaluated based on the accuracy of the score and the relevance of the feedback.

The system described in this paper scores English sentences composed by Korean students learning English. A detailed explanation of the system is given in (Kim et al., 2007). The scores are calculated from three different phases including word, syntax and mapping, each of which is designed to assign 0~2 points. Three scores are added up to generate the final score. A spelling error, a plural form error, and a confusable word error are considered as typical word errors. A subject verb agreement error, a word order error and relative clause error are typical examples of syntactic errors. Even when a student sentence is perfectly correct in lexical and syntactic level, it may fail to convey what is meant by the question. Such sentences are evaluated as grammatical, but cannot be a correct answer for the question. In this case, the errors can only be recognized by comparing a student sentence with its correct answers. The differences between a student answer and one of the answers can be considered as mapping errors.

These three phases are independent from one another since they use different processing method, and refer different information. Independency of three phases causes some problems.

(Ex1) *Correct answer*: The earth is bigger than the moon. *Student answer*: The earth is small than the Moon.	
Err1: MODIFIER_COMP_ERR|4-7|	syntactic
Err2: LEXICAL_ERROR|4|	mapping

(Ex1) is an example of error reports provided to a student. The following two lines in (Ex1) show the error information detected from the student answer by the system. Err1 in (Ex1) reports a comparative form error of an adjective 'small', which covers the 4 ~ 7th words of the student sentence. Err2 indicates that the 4th word 'small' of the student sentence is different from that of the answer sentence. The difference was identified by comparing the student sentence and the answer sentence. Err1 was detected at the syntactic phase whereas Err2 was at the mapping phase. These two errors points to the same word, but have been reported as different errors.

(Ex2) *Correct answer*: She is too weak to carry the bag. *Student answer*: She is too weak to carry the her bag.	
Err1: EXTRA_DET_ERR|7-9|	syntactic
Err2: UNNECESSARY_NODE_ERR|8|(her)	mapping

Similarly, Err1 in (Ex2) reports an incorrect use of an article at the 7~9th words. The syntactic analysis recognizes that 'the' and 'her' cannot occur consecutively, but it is not capable of determine which one to eliminate. Err2, on the other hand, pinpoints 'her' as an incorrectly used word by comparing the student sentence and the answer sentence.

(Ex1) and (Ex2) have presented the errors which are detected at different processing phases,

Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications, pages 80–83,
Los Angeles, California, June 2010.

but represent the same error. Since these redundant errors are a hindering factor to calculate accurate scores, one of the errors has to be removed. The proposed system deals with 70 error types; 16 for word, 46 for syntax, and 14 for mapping. In this paper, we have adopted a human-computer collaboration approach by which linguistic experts assist the system to decide which one of the redundant errors should be removed.

2 Redundant Errors

The system-detected errors are reported in the following format:

Error_ ID | Error_ Position | Error_Correction_Info

Each error report is composed of three fields which are separated by '|'. The first field contains error identification. The second includes the numbers indicating where the error is detected in a student input sentence. For example, if the field has number "5-7", it can be interpreted as the input sentence has an error covering from the 5th word to 7th word. Since syntactic errors are usually detected at a phrasal level, the position of an error covers more than one word. The third field may or may not be filled with a value, depending on the type of an error. When it has a value, it is mostly a suggestion, i.e. a corrected string which is formed by comparing a student sentence with its corresponding correct answer.

2.1 Definition of Redundant Errors

(Condition 1) The errors should share an error position.
(Condition 2) The errors should be detected from different error process phases.
(Condition 3) The errors should represent linguistically the same phenomenon.

(Condition 1) implies that the two errors must deal with one or more common words. The position is indicated on the student sentence. However, there are some exceptions in displaying the position. An example of the exception is 'OBLIGATORY_NODE_MISSING_ERR' and 'OPTIONAL_NODE_MISSING_ERR' which are mapping errors. Since these errors are detected when a certain word is missing from a student input but included in the answer, the position is indicated on the answer sentence. Err5 and Err6 from (Ex3) represent the case. Error position '(7)' and '(8)' [1] means that the 7th and 8th word of the answer sentence, 'to' and 'our' are missing, respectively. When an error position points to an answer sentence not a student sentence, the error cannot be checked with whether it includes the words shared with the errors whose positions indicate the student sentence. In this case, the error is assumed to have shared words with all the other errors; Err5 and Err6 are considered containing shared words with Err 1~4 in (Ex3).

(Ex3) *Correct answer*: She is a teacher who came to our school last week. *Student answer*: She is a teacher who come school last week.	
Err1: CONFUSABLE_WORD_ERR|9|week	word
Err2: SUBJ_VERB_AGR_ERR|3-7|	syntactic
Err3: VERB_SUBCAT_ERR|6-7|	syntactic
Err4: TENSE_UNMATCHED_ERR|6|came[past]	mapping
Err5: OPTIONAL_NODE_MISSING_ERR|(7)|to	mapping
Err6: OPTIONAL_NODE_MISSING_ERR|(8)|our	mapping

Err1 and Err2 from (Ex3) cannot be redundant errors since they do not share an error position and accordingly do not satisfy Condition 1. Err2 and Err3 share error positions 6~7, but they are not also considered as redundant errors since both of them were detected at the same process phase, the syntactic phase. Err2 and Err4 satisfy both Condition 1 and 2, but fail to meet Condition 3. Err2 represents the subject-predicate agreement error whereas Err4 points out a tense error. In comparison, Err3 and Err5 are legitimate candidates of "redundant errors" since they satisfy all the conditions. They share error positions, but were detected from different error process phases, the syntactic phase and the mapping phase, respectively. They also deal with the same linguistic phenomenon that a verb "*come*" does not have a transitive sense but requires a prepositional phrase led by "*to*".

2.2 Detection of Redundant Errors

Two errors need to satisfy all the conditions mentioned in section 2.1 in order to be classified as redundant errors. The system's detecting process began with scoring 14,892 student answers. From the scoring result, the candidates which met Condition 1 and 2 were selected. In the following subsections, we have described how to determine the final redundant errors using the system in collaboration with human's efforts.

[1] Error positions in answer sentences are marked with a number surrounded by a pair of parenthesis.

2.2.1 Selection of the Candidates

The system selected candidate errors which satisfied Condition 1 and 2 among the student sentences. For example, Table 1 presents 8 candidates extracted from (Ex3).

1	CONFUSABLE_WORD_ERR|9|week OPTIONAL_NODE_MISSING_ERR|(7)|to
2	CONFUSABLE_WORD_ERR|9|week OPTIONAL_NODE_MISSING_ERR|(8)|our
3	SUBJ_VERB_AGR_ERR|3-7| TENSE_UNMATCHED_ERR|6|came[past]
4	SUBJ_VERB_AGR_ERR|3-7| OPTIONAL_NODE_MISSING_ERR|(7)|to
5	SUBJ_VERB_AGR_ERR|3-7| OPTIONAL_NODE_MISSING_ERR|(8)|our
6	VERB_SUBCAT_ERR|6-7| TENSE_UNMATCHED_ERR|6|came[past]
7	VERB_SUBCAT_ERR|6-7| OPTIONAL_NODE_MISSING_ERR|(7)|to
8	VERB_SUBCAT_ERR|6-7| OPTIONAL_NODE_MISSING_ERR|(8)|our

Table 1 **Candidate pairs of errors extracted from (Ex3).**

As a result of the selection process, the total of 150,419 candidate pairs was selected from 14,892 scoring results of the student sentences.

2.2.2 Filtering Candidate Errors

The candidates extracted through the process mentioned in 2.2.1 were classified based on their error identifications only, without considering error position and error correction information. 150,419 pairs of the errors were assorted into 657 types. The frequency of each type of the candidates was then calculated. These candidate errors were filtered by applying PMI (Pointwise Mutual Information) and RFC (Relative Frequency Count) (Su et al., 1994).

$$PMI(E_1, E_2) = \log \frac{P(E_1, E_2)}{P(E_1)P(E_2)} \quad (1)$$

$$RFC(E_1, E_2) = \frac{freq(E_1, E_2)}{\overline{freq}} \quad (2)$$

PMI is represented by a number indicating how frequently two errors E_1 and E_2 occur simultaneously. RFC refers to relative frequency against average frequency of the total candidates. The filtering equation is as follows:

$$PMI(E_1, E_2) \times RFC(E_1, E_2) \geq k \quad (3)$$

Using this equation, the system filtered the candidates whose value was above the threshold *k*. For this experiment, 0.4 was assigned to *k* and 111 error types were selected.

2.2.3 Human Collaborated Filtering

Filtered 111 error types include 29,588 candidate errors; on the average 278 errors per type. These errors were then handed over to human experts[2] to confirm their candidacy. They checked Condition 3 against each candidate. The manually filtered result was categorized into three classes as shown in Table 2.

Class A: (number: 20)	(DET_NOUN_CV_ERR,DET_UNMATCHED_ERR) (EXTRA_DET_ERR, DET_UNMATCHED_ERR) (MODIFIER_COMP_ERR, FORM_UNMATCHED_ERR) (MISSPELLING_ERR, LEXICAL_ERR) …
Class B: (number: 47)	(SUBJ_VERB_AGR_ERR,TENSE_UNMATCHED_ERR) (AUX_MISSING_ERR, UNNECESSARY_NODE_ERR) (CONJ_MISSING_ERR, DET_UNMATCHED_ERR) …
Class C: (number: 44)	(VERB_FORM_ERR, ASPECT_UNMATCHED_ERR) (VERB_ING_FORM_ERR, TENSE_UNMATCHED_ERR) (EXTRA_PREP_ERR, UNNECESSARY_NODE_ERR) …

Table 2 **Classes of Human Collaborated Filtering.**

Class A satisfies Condition 1 and 2 and is confirmed as redundant errors. When a pair of errors is a member of Class A, one of the errors can be removed. Class B also meets Condition 1 and 2, but is eliminated from the candidacy because human experts have determined they did not deal with the same linguistic phenomenon. Each error of Class B has to be treated as unique. With respect to Class C, the errors cannot be determined its candidacy with the information available at this stage. Additional information is required to determine the redundancy.

2.2.4 Final Automated Filtering Using Decision Rules

In order to confirm the errors of Class C as redundant, additional information is necessary.

(Ex4) *Correct answer*: I don't know why she went there. *Student answer*: I don't know why she go to their.	
Err1: CONFUSABLE_WORD_ERR|8|there	word
Err2: SUBJ_VERB_AGR_ERR|6|went[3S]	syntactic
Err3: **EXTRA_PREP_ERR|6-8|**	syntactic
Err4: **UNNECESSARY_NODE_ERR|7|(to)**	mapping
Err5: TENSE_UNMATCHED_ERR|6|went[past]	mapping

(Ex5) *Correct answer*: Would you like to come? *Student answer*: you go to home?	
Err1: FIRST_WORD_CASE_ERR|1|	word
Err2: **EXTRA_PREP_ERR|3-4|**	syntactic
Err3:OBLIGATORY_NODE_MISSING_ERR|(1,3)| Would _ like	mapping
Err4: **UNNECESSARY_NODE_ERR|4|(home)**	mapping
Err5: LEXICAL_ERR|2|come	mapping

[2] They are English teachers who have a linguistic background and teaching experiences of 10 years or more.

EXTRA_PREP_ERR' and 'UNNECESSARY_NODE_ERR' were selected as a candidate from both (Ex4) and (Ex5) through the steps mentioned in section 2.2.1 ~ 2.2.3. The pair from (Ex4) is a redundant error, but the one from (Ex5) is a false alarm. (Ex4) points out a preposition '*to*' as an unnecessary element whereas (Ex5) indicates a noun '*home*' as incorrect.

To determine the finalist of redundant errors, we have adopted a decision tree. To train the decision tree, we have chosen a feature set for a pair of errors (E_1, E_2) as follows.

(1) The length of shared words in E1 and E2 divided by the length of a shorter sentence (*shared_length*)
(2) The length of non-shared words in E1 and E2 divided by the length of a shorter sentence. (*non_shared_length*)
(3) The Error_Correction_Info of E1 (*E1.Correction_Info*)
(4) The Error_Correction_Info of E2 (*E2.Correction_Info*)
(5) Edit distance value between correction string of E1 and E2 (*edit_distance*)
(6) Error Position of E1 (*E1.pos*)
(7) Error Position of E2 (*E2.pos*)
(8) Difference of Error positions of E1 and E2 (*diff_error_pos*)

12,178 pairs of errors for 44 types in Class C were used to train a decision tree. We used CART (Breiman et al., 1984) to extract decision rules. The followings show a part of the decision rules to eliminate redundant errors from Class C.

```
E1=CONJ_MISSING_ERR
E2=OPTIONAL_NODE_MISSING_ERR
If E2.Correction_Info='conj' and E2.pos=1 then redundant_error

E1=EXTRA_PREP_ERR, E2=UNNECESSARY_NODE_ERR
If E2.Correction_Info='prep' and E2.pos=1 then redundant_error

E1=VERB_SUBCAT_ERR,
E2=OPTIONAL_NODE_MISSING_ERR
If diff_error_pos <=3 and E2.Correction_Info={'prep' , 'adv'}
then redundant_error

E1=VERB_ING_FORM_ERR, E2=TENSE_UNMATCHED_ERR
If E2.Correction_Info='verb-ing' then redundant_error
…
```

The errors are removed according to a priority specified in the rules. The syntactic phase is assigned with the highest priority since syntactic errors have the most extensive coverage which is identified at a phrasal level. On the other hand, the lowest priority is given to the mapping phase because mapping errors are detected through a simple word-to-word comparison of a student input with the correct answer.

3 Evaluation

We evaluated the accuracy of determining redundant errors. Table 3 presents the results. The evaluation was performed on 200 sentences which were not included in the training data. Even though the redundancy of the pairs of errors in Class A and Class B are determined by the human expert, the accuracies of both classes did not reach 100% because the errors detected by the system were incorrect. The total accuracy including Class A, B, and C was 90.2%.

	Class A	Class B	Class C
Accuracy	94.1%	98.0%	82.3%

Table 3**: The accuracy**

The performance of our automated scoring system was measured using exact agreement (Attali and Burstein, 2006) of the final scores calculated by the system and human raters. The overall performance was improved by 2.6% after redundant errors were removed.

4 Conclusion

This paper has introduced a human collaborated filtering method to eliminate redundant errors reported during automated scoring. Since scoring processes are performed through three separate phases including word, syntax and mapping, some of the errors are redundantly reported with different IDs. In addition, it is almost impossible to predict every type of errors that could occur in student answers. Because of these issues, it is not easy for the system to automatically determine which errors are reported redundantly, or to estimate all the possible redundant errors. As a solution to these problems, we have adopted a human assisted approach. The performance has been improved after redundant errors were removed with the approach implemented in the system.

References

Jee Eun Kim, K. J. Lee and K. A. Jin. 2007. Building an Automated Scoring System for a Single Sentence. *Korea Information Processing Society* Vol.4, No.3 (in Korean).

Keh-Yih Su, Ming-Wen We and Jing-Shin Chang. 1994. A Corpus-based Approach to Automatic Compound Extraction, In *Proceedings of the ACL 94*.

Leo Breiman, Jerome Friedman, Charles J. Stone, and R.A. Olshen. 1984. *Classification and Regression Trees*. Monterey, Calif., U.S.A.: Wadsworth, Inc.

Yigal Attali and Jill Burstein. 2006. Automated Essay Scoring with e-rater® V.2.