IMPROVING NONNATIVE SPEECH UNDERSTANDING USING CONTEXT AND N-BEST MEANING FUSION

Yushi Xu, Stephanie Seneff

Spoken Language Systems Group, MIT Computer Science and Artificial Intelligence Laboratory, USA

ABSTRACT

Speech understanding of nonnative language learners' speech is a challenging problem. In this paper, we investigate the use of dialogue context cues to help improve concept error rate (CER) of nonnative speech in a language learning system. Given that the student's task is known, we show that incorporating the game scores to help select the best hypothesis improves the CER. We also introduce a novel N-best fusion method to create a single final hypothesis on the meaning level. The experimental results show that the fusion methods can further improve the CER.

Index Terms— N-Best Fusion, Spoken Dialogue Systems, Computer-Aided Language Learning

1. INTRODUCTION

Word error rate (WER) has been used as the most common metric for evaluating speech recognition performance. In this metric, every word in the utterance is considered as equally important. However, for larger speech-based systems such as spoken dialogue systems, some of the errors in the word sequence are much more important than others. Errors in many of the stop words are unimportant for language understanding, while errors in the content words are very likely to cause misunderstanding. Thus, for a spoken dialogue system, instead of optimizing the word error rate, it is more useful to optimize the concept error rate (CER), capturing language understanding performance.

To calculate CER involves more than the speech recognizer. One common method used in spoken systems is that the language understanding component takes the most confident output of the speech recognizer, accepts and processes it if the confidence score is above a certain threshold, and rejects it if it is below the threshold. The drawback of this method is that only the top recognizer's hypothesis is used, and the information in the rest of the N-best list is lost.

Better approaches pass the entire N-best hypothesis list to the subsequent modules, and defer the decision until later modules have produced useful cues. In [1], the decision is made by the parser. The top hypothesis on the N-best list that yields a full parse is selected as the best hypothesis. In [2], features obtained from the dialogue manager and the domain knowledge are used in addition to the acoustic features, to classify the hypotheses into *accept*, *clarify*, *reject* or *ignore*. In [3], a statistical user simulator measures the likelihood that the user would say each hypothesis in the current context. [4] adopted a similar approach by using features derived from the recognition score, distributional aspects of the N-best list, and the system's response. Instead of classifying the recognition hypotheses, the classification was optimized based on the system's response.

In this paper, we examine the use of dialogue context cues for improving CER in a language learning system. We also present novel N-best meaning fusion methods to take into account the entire N-best list.

The rest of the paper is organized as follows. In Section 2, we describe the background of this work, explaining the system and the data we have collected. In Section 3, we elucidate the usefulness of context cues by experiments in N-best selection. In Section 4, the N-best meaning fusion methods are introduced and validated with experiments. Section 5 concludes the paper with some future work.

2. BACKGROUND

2.1. Dialogue Game for Second Language Learning

This paper utilizes data that were collected in the context of a dialogue game designed for Mandarin learning [5]. In the framework, the student is asked to interact with the system to book flights that satisfy a given scenario similar to the following paragraph.

> *"You are currently in New York. You plan to travel to Chicago on the first Monday of October, and come back three weeks later. You prefer morning flights."*

The scenario is generated from several abstract templates, and shown in a natural paragraph in English using a number of different wordings. The dialogue is then carried out fully in Mandarin. A parser parses the hypotheses from the speech recognizer, and produces meaning representations. Before being sent to the dialogue manager, the meaning representations are further compressed into key-value forms using language generation techniques. Figure 1 shows an example of the key-value representations.

During the dialogue, the system gives real-time feedback to encourage the student, or to point out mistakes the student has made. The student collects points for the utterances he has spoken, and advances to a higher level when enough points have been accumulated. Points are given according to the following aspects: sentence wellness, context wellness, dialogue progress, independence and scenario difficulty.

```
source: “NYC”
departure_date: {  month: “OCT”
                   day_number: 3  }
```

Figure 1. An example key-value representation of the utterance "flight from New York on October 3."

Scenarios in the higher levels are more complex and have more constraints. The dialogues are conducted against a simulated flight database, which generates different flights for each session. The scenarios are made sure to be satisfiable via a “self update” feature. Whenever no flight can be found, the system changes some constraints or removes some constraints from the scenario, and prompts the student to continue the dialogue with the new constraints.

2.2. Data Collection

The data we use for this research were obtained from a data collection effort involving nine learners of Mandarin and three native speakers. The nonnative speakers were asked to self-rank their Mandarin proficiency. The average score for the speaking ability was 3.0 on a 5-point scale, where 5 stands for native-like.

Each subject completed 2 to 10 scenarios, which were randomly generated from several written templates that include different itinerary types and different constraints. Each subject started from level 1, and gradually advanced to higher levels according to their performance.

The acoustic models of the speech recognizer [6] were trained from native speech data. The recognizer outputs 10 best hypotheses for each utterance, ordered by decreasing total score from both the acoustic model and the language model. The language understanding component then produced the key-value representation for each hypothesis. All the hypotheses were sent to the dialogue and performance assessment components to obtain the game scores. After pruning out the empty utterances and out-of-domain utterances (*e.g.* user making fun of the system,) we obtained 148 native utterances and 509 nonnative utterances. All the utterances were transcribed. Reference key-value representations were also created by a human expert.

3. N-BEST SELECTION USING CONTEXT CUES

3.1. Methods

Our situation is unusual in that, although the speech is likely to be highly accented, the context is known much more fully than is typical for a standard information-access system. For a language learning system, the learning scope is pre-defined. Particularly, for this system, although the dialogue is fully natural, the exact scenario is known at the time of the conversation. Assuming the student is cooperative, *i.e.*, he tries to solve the scenario, the information in the scenario provides strong cues for the recognition.

To verify this, we designed an N-best selection experiment. The goal is to select one best hypothesis from the N-best list that the recognizer produces. Four selection methods incorporating cues from different stages of the processing are compared.

Top recognizer hypothesis (1-best). The top hypothesis on the N-best list is selected.

Top full parse (parse). The top hypothesis that produces a full parse is selected.

Best dialogue score (dialogue). Hypotheses that fail to produce a full parse are filtered out. For the remaining hypotheses, two scores from the performance assessment component are used as the dialogue score: the context score and the dialogue progress score.

The context score assesses the appropriateness of the current utterance given the previous system’s response. For wh-questions, a negative score is given if the student provides the wrong type of information. For verification questions, since students do not necessarily respond with an explicit “yes” or “no” (*e.g.* responding to the question “do you want to book this flight” with “are there any other flights”), no deduction is given. However, if there is an explicit “yes” or “no”, a positive score is assigned.

The dialogue progress score assesses how far the current dialogue state has advanced towards a successful conclusion. It is calculated using the key points extracted from the scenario. The key points include the type of the information that the student needs to convey to the system, as well as the status of the current itinerary. Every correctly achieved key point is worth one point, while an incorrect key point results in one point deduction. The dialogue progress score for each utterance is the difference between the overall dialogue progress after this turn and that of the previous turn.

The sum of the context score and the dialogue progress score is used as the dialogue score. The hypothesis that maximizes the dialogue score is selected

Combined score (combined). The dialogue score implicitly contains the parse score by excluding the hypotheses that cannot produce a full parse. The goal of the combined score is to further incorporate the acoustic score. We assign N-best rank scores to the hypotheses. The top three hypotheses receive three points, the next three receive two, and the rest receive one.

3.2. Experimental Results

The four methods were evaluated for both WER and CER. For CER calculation, since our key-value representation is hierarchical, we first perform a normalization procedure to flatten the representation. Figure 2 shows the example representation after normalization.

```
source: “NYC”
departure_date_month:  “OCT”
departure_date_day_number: 3
```

Figure 2. Key-value representation after flattening.

CER is calculated using the following equation.

$$CER = \frac{\#slots_{ins} + \#slots_{del} + \#slots_{sub}}{\#slots_{ref}}$$

A substitution is counted if both the reference and the hypothesis contain the same key but with different values. An insertion is counted if the hypothesis contains a key that does not appear in the reference. Likewise, a deletion is counted if the reference contains a key that does not show up in the hypothesis. The CER is a very strict metric, for the denominator is usually small, and to be considered as correct, both the key and the value need to be matched. For example, if the utterance "I want to go from Boston" is misrecognized as "I want to go to Boston", the WER is 0.16, but the CER is 2.0 (inserted a destination, deleted a source).

Table 1 lists the WER and CER of the four N-best selection methods. Statistically significant improvements were obtained in terms of both WER and CER for both native and nonnative when the dialogue scores and the N-best rank scores were both incorporated. Especially for the nonnative CER, over 12% absolute improvement was gained. We also experimented with real scores from the recognizer (the acoustic score plus the language model score) for the combined method. The results showed no significant difference from using the N-best rank scores.

Table 1. WER and CER of the N-best selection methods. Bold indicates statistically significant improvement over 1-best method.

	Native WER	Nonnative WER	Native CER	Nonnative CER
1-best	15.33%	19.30%	42.50%	56.49%
parse	15.33%	19.02%	41.39%	56.28%
dialogue	14.02%	**17.25%**	**35.00%**	**45.00%**
combined	**13.87%**	**17.20%**	**33.33%**	**43.72%**

4. N-BEST FUSION

4.1. Oracle experiments

We have verified that using context cues to select the best hypothesis improves the CER. However, selecting a single best hypothesis ignores the information contained in the rest of the N-best list. Thus, we would like to explore methods to fuse the N-best list into a single hypothesis to contain the information most likely to be correct from all the hypotheses.

We explore the N-best fusion technique at the level of key-value representations, because we are more concerned with correctly understanding the user's meaning. Besides, with the key-value representations, the unimportant information, such as carrier words, has been discarded during language understanding, resulting in fewer distractions for the fusing process.

As a validation of the feasibility of this idea, we first examine the oracle performance of the N-best fusion idea. The oracle works as follows: for every key-value pair in the reference, if it exists in one of the N-best key-value representations, the algorithm adds it into the final fused result. Thus, the oracle algorithm will not produce any substitution or insertion errors.

Table 2 shows the CER of the oracle algorithm, in comparison with the N-best selection oracles that optimize the WER and CER respectively. WER is not calculated, since the fusion algorithm might produce a result that is not in the original N-best list, and it would be challenging to rebuild the utterance from a fused key-value representation.

Table 2. CER of different oracle algorithms.

	Native	Nonnative
Selection (WER)	27.78%	33.94%
Selection (CER)	16.94%	23.83%
Fusion	11.94%	16.91%

The fusion oracle substantially outperforms both selection oracles in terms of the CER, which is promising for exploring real fusion methods.

4.2. Heuristic Fusion

To fuse the N-best key-value representations into one, appropriate key-value pairs need to be selected from the N-best candidates. The keys and the values represent different types of information. The keys are usually derived from a syntactic structure, while the values usually correspond to content words. For example, the keys "source" and "destination" are derived from two prepositional phrases, whose values are the objects of the prepositional phrases. Thus, the keys should be more robust than the values, because the vocabulary to form the syntactic structures is much smaller than that of the content words, and usually is well covered in the language model. To take advantage of this property, we would like to separate the tasks of selecting the keys and selecting their values.

On the other hand, the dialogue scores obtained from the system are attributed to the key-value pairs, not the keys alone or the values alone. Thus, we score both the key-value pair as a whole, as well as the keys and values separately.

Key-value pair scoring. The key-value pairs are scored according to their contribution towards the dialogue scores. However, the dialogue score for each key-value pair is hard to obtain for two reasons. First, due to the uncertainty embedded into the system, it is very hard to reproduce exactly the same scenario and the same dialogue. Secondly, certain dialogue progress is credited toward a combination of multiple key-value pairs, rather than a single pair. Therefore, we compute the score for each key-value pair by calculating the correlation between its occurrence and the dialogue score of each N-best hypothesis. The detailed formulation is as follows, where C is the occurrence vector, and D is the dialogue score vector for the N-best hypothesis.

$$s(k, v) = \mathrm{corr}(\vec{C}, \vec{D})$$

$$c_i = \begin{cases} 1 & (k, v) \in hyp_i \\ -1 & (k, v') \in hyp_i, v' \neq v \\ 0 & \text{otherwise} \end{cases}$$

Key and value scoring. Each key (or value) is scored using the following equation, where w_i is the weight for each N-best hypothesis calculated using the dialogue score and the N-best rank score discussed in Section 3.1.

$$s(k) = \sum_i \delta(k,i) w_i$$

$$\delta(k,i) = \begin{cases} 1 & \text{k appears in hyp}_i \\ 0 & \text{otherwise} \end{cases}$$

A key-value pair (k,v) that satisfies either of the following two criteria is selected into the final fusion result.

(1) $s(k,v) > thres_{kv}$

(2) $s(k) > thres_k \sum_i w_i$, and for any other possible values v' of k, $s(v) \geq s(v')$

The selection is performed on the leaf key-value pairs. The most frequent parent, if any, is assigned to the pairs to re-create the hierarchical structure.

4.3. Fusion with SVM

We also experimented with fusion of the N-best key-value representation using an SVM classifier. Each key-value pair is classified into POSITIVE or NEGATIVE. If two pairs with the same key are classified into POSITIVE, the one with a higher score is retained.

The following features are used for classification: percentage of occurrence in the N-best list, index of first occurrence in the N-best list, correlation with the dialogue scores, sum of dialogue scores of the hypotheses it appears in, and sum of rank scores of the hypotheses it appears in.

4.4. Results

Table 3 shows the CER result of the two fusion methods in comparison with the N-best selection method. For heuristic fusion, we chose $thres_{kv}$=0.8 and $thres_k$=0.6. We used a linear kernel for the SVM experiments. Due to the small amount of data we have, the SVM fusion results were obtained via leave-one-speaker-out cross validation.

The heuristic fusion method gained statistically significant improvements on the nonnative data. For the native data, the CER was lowest using the SVM fusion, but the result was not statistically significant.

Table 3. CER of the fusion methods. Bold shows the statistically significant results over the selection method.

	Native	Nonnative
Selection (combined)	33.33%	43.72%
SVM Fusion	31.67%	44.04%
Heuristic Fusion	31.94%	**40.21%**
Manual Fusion	**27.22%**	**34.26%**

We also did a manual fusion experiment with an expert. The result showed that the human still performs a lot better than both methods. One difference we noticed is that the human takes into account the existence of other keys when deciding whether a particular key should be selected or not, which is not modeled by either of the methods.

5. CONCLUSIONS AND FUTURE WORK

We presented several experiments to improve speech understanding in the context of a Mandarin language learning dialogue system. We showed that incorporating the dialogue context cues to select a best hypothesis helps improve the CER. We also presented a heuristic method to fuse an N-best list into one hypothesis at the meaning level, as well as a fusion method using an SVM classifier. The experimental results showed that the heuristic fusion method further improved the CER statistically significantly compared to our best N-best selection method.

In both the N-best selection and N-best fusion experiments, the dialogue context cues helped a lot. However, the dialogue scores were obtained under the assumption that the student follows the given scenario. In real data, we did notice students carelessly providing wrong information, in which case the dialogue scores would favor incorrect recognition hypotheses which might be less incorrect with regards to the scenario. This creates a hard problem, which will require more analysis to solve.

In the future, we would like to explore more sophisticated fusion methods that model the probability of a key given other existing keys. Another interesting research topic is how to recover the word sequence given a fused key-value representation. This is especially important in a language learning system, since knowing the word sequence allows us to perform analysis such as pronunciation assessment.

6. ACKNOWLEDGMENT

This work is partly funded by Quanta Computer under the T-Party Initiative.

7. REFENRECES

[1] S. Seneff, "Response Planning and Generation in the Mercury Flight Reservation System," *Computer Speech and Language*, vol. 16, pp. 283-312, 2002.

[2] M. Gabsdil and O. Lemon, "Combining Acoustic and Pragmatic Features to Predict Recognition Performance in Spoken Dialogue Systems," in *Proc. ACL*, 2004.

[3] O. Lemon and I. Konstas, "User Simulations for context-sensitive speech recognition in Spoken Dialogue Systems," in *Proc. European Chapter of ACL*, Athens, Greece, 2009.

[4] A. Gruenstein, "Response-Based Confidence Annotation for Spoken Dialogue Systems," in *Proc. SIGDial*, Columbus, Ohio, USA, 2008.

[5] Y. Xu and S. Seneff, "A Generic Framework for Building Dialogue Games for Language Learning: Application in the Flight Domain," in *Submitted to SIGSLaTE*, 2011.

[6] J. Glass, "A probabilistic framework for segment-based speech recognition," *Computer Speech and Language*, vol. 17, no. 2-3, pp. 137-152, April/July 2003.

[7] S. Seneff, "TINA: A Natural Language System for Spoken Language Applications," *Computational Linguistics*, vol. 18, no. 1, pp. 61 - 86, March 1992.