Detecting Incorrectly-Segmented Utterances for Posteriori Restoration of Turn-Taking and ASR Results

Naoki Hotta¹, Kazunori Komatani¹, Satoshi Sato¹, Mikio Nakano²

¹Graduate School of Engineering, Nagoya University, Japan

²Honda Research Institute Japan, Co., Ltd., Japan

{n.hotta, komatani, ssato}@nuee.nagoya-u.ac.jp, nakano@jp.honda-ri.com

Abstract

Appropriate turn-taking is important in spoken dialogue systems as well as generating correct responses. We have developed a method that performs a posteriori restoration of incorrectly segmented utterances caused by erroneous voice activity detection (VAD), which result in automatic speech recognition (ASR) errors and inappropriate turn-taking. A crucial part of the method is to classify whether the restoration is required or not. We cast it as a binary classification problem detecting originally single utterances from pairs of utterance fragments. Various features are used representing timing, prosody, and ASR result information to improve its accuracy. Furthermore, two kinds of feature selection are performed to obtain effective and domain-independent features. The experimental results showed that the proposed method outperformed a baseline with manually-selected features by 4.8% and 3.9% in cross-domain evaluations with two domains. More detailed analysis revealed that the dominant and domain-independent features were utterance intervals and results from the Gaussian mixture model (GMM). **Index Terms:** spoken dialogue system, VAD error, turn taking, a posteriori restoration

1. Introduction

Appropriate turn-taking as well as generating correct responses is imperative in spoken dialogue systems. Turn-taking generally denotes that two people are talking alternatively. From this viewpoint, spoken dialogue systems should also not start speaking while the user is speaking [1].

However, sometimes a spoken dialogue system will mistakenly start speaking while the user is still speaking. A simple example is outlined in Fig. 1, where the system interrupts a user who pauses in the middle of uttering “What are the best restaurants in Singapore?” Here, a voice activity detection (VAD) error occurs: the user utterance is divided into two fragments by the short pause in the middle and the system accordingly starts responding to the first fragment. This phenomenon, called the *incorrect segmentation* of user utterances, causes two problems:

- the system starts speaking while a user is still speaking, and
- automatic speech recognition (ASR) tends to fail for the wrong VAD results.

The ASR results become always incorrect when word fragments are not in the system’s dictionary.

We have previously developed a method of solving these two problems [2]. For the former problem, we added rules on the MMDAgent toolkit [3] to terminate the system utterance

Figure 1: Example of inappropriate turn-taking

when an incorrect segmentation is detected. For the latter, we integrate utterance fragments and perform ASR again.

The crucial part of this method is to classify whether or not the restoration is required. In this work, we improve the classification accuracy of originally single utterances from pairs of utterance fragments. We cast this as a binary classification problem and perform decision tree learning with various features. The features are extracted from pairs of utterance fragments and represent timing, prosody, and ASR result information. To ensure use across various domains, the features should not be dependent on any specific domain. We thus perform two kinds of feature selection to obtain effective and domain-independent features for improving classification accuracy.

2. A Posteriori Restoration for VAD Errors

VAD errors occur often, especially when users make short pauses within utterances due to breathing or thinking about what to say next. Such short pauses can cause incorrect segmentation.

A VAD module generally detects silences on the basis of the amplitude of the target speech signals and zero crossing rates [4]. User utterances are regarded as having ended when the duration of silence exceeds a threshold. This threshold needs to be set smaller to ensure that the system can respond quickly enough. Responses with latency make users think their utterance has been rejected, and they may repeat it again. This should be avoided from the viewpoint of the user interface. When the threshold is set smaller, it becomes more difficult to determine whether the user has actually finished an utterance or intends to continue it. That is, there is a trade-off between latency and the false cut-in rate [5].

We have adopted an approach of a posteriori restoration [2]. Two steps are involved in the restoration process.

1. Classify whether a pair of utterance fragments resulted from an incorrect segmentation or not.
2. Integrate the utterance fragments if the above classification is required.

An outline of the proposed method is shown in Fig. 2. Here, a user utterance is segmented into a pair of utterance fragments, denoted hereafter as the first and second fragments. Given a pair

Figure 2: Overview of proposed method

Table 1: Target data

Domain	Restaurant	World Herit.
No. of dialogues	120	156
No. of VAD results	6615	6593
No. of target fragment pairs	255	354

of utterance fragments, the system determines whether the fragments should be interpreted by integrating them or separately. This is equal to classifying whether the fragment pair was originally a single utterance or not. If the fragments are deemed to be parts of an utterance, the system does not start speaking and performs ASR again after integrating the fragments in order to restore turn-taking and the ASR results, which are erroneous due to incorrect segmentation. If the fragments are deemed to be two utterances, the system responds normally; that is, it generates responses based on the ASR results for each fragment.

Another approach to dealing with such ASR errors can be to add shorter subwords corresponding to utterance fragments into the ASR dictionary, as Jan et al. [6] and Katsumaru et al. [7] have done. However, this would degrade ASR accuracy because too many subwords would be added into the ASR dictionary.

3. Analysis of Utterance Fragments

3.1. Target Data

We use dialogue data in two domains: restaurants and world heritage sites. Data were collected by our spoken dialogue systems that search databases of the two domains [8].

Our target is pairs of utterance fragments likely to require the restoration. Thus, we selected pairs of utterance fragments (VAD results) close in time, as in our previous study [2]. We specifically selected fragment pairs whose intervals are shorter than 2000 milliseconds and each fragment is longer than 800 milliseconds; the latter is to exclude short noises. We also manually exclude repairs in advance, as we think repairs are different phenomena from our target and should be detected by other features. We have actually found in a preliminary experiment that repairs can be automatically excluded with a precision of 70%–90% by using the overlap ratio of phoneme bigrams between the fragments, i.e., how many phonemes are commonly included.

Overall, we use 255 and 354 pairs of utterance fragments in the restaurant and world heritage domains, respectively. The details are listed in Table 1.

Figure 3: Examples of pairs that are originally a single utterance

Figure 4: Examples of pairs that are not single utterances

3.2. Target Labels for Detection

The system needs to classify whether the restoration is required or not. That is, when given a pair of fragments, the system determines whether the pair should be interpreted by integrating them or separately. Restoration is required when the fragments were originally a single utterance.

We manually annotated each fragment pair with labels indicating if it was originally a single utterance or not. Since the pairs were automatically obtained from VAD results, the data set contains various sounds that are not actually user utterances, such as coughs, wind noise, the system's synthesized voices, etc.

Figure 3 shows examples in which fragment pairs are originally single utterances. At the top is an example of a user wanting to say the lengthy keyword “Santa Maria delle Grazie”, which is a world heritage site in Italy. However, the user pauses slightly in the middle of the word and the utterance is thus segmented incorrectly. In this case, ASR always fails because such word fragments are not in the system's dictionary. At the bottom is a user saying “I'd like to know how much lunch costs”. These fragment pairs should be integrated.

Figure 4 shows examples in which fragment pairs are not single utterances. At the top is a fragment pair where the first fragment is filler. This pair does not have to be integrated because the first fragment has no content to be conveyed to the system. This is the same for a fragment pair including either noise or the system's synthesized voice. At the bottom, the user's intentions (dialogue acts) are different for each fragment. Those of the first and second fragments are to delete search conditions for stations and foods, respectively. These fragments should not be interpreted by integrating them.

After the manual annotation, the numbers of originally single utterances are 156 (61.2%) and 270 (76.3%) out of the 255 and 354 pairs in the restaurant and world heritage domains, respectively.

4. Classification by Decision Trees

4.1. Features

We perform decision tree learning for this binary classification problem. This is because of interpretability of the obtained results and behaviors of the features. Use of other classifiers such

Table 2: Eighteen features used for decision tree learning

Features from ASR engine	Timing features	Prosodic features
(1) Average CM score of first fragment	(9) Interval between fragments	(14) Volume change in final part of first fragment
(2) CM score of last word of first fragment	(10) Duration of tail silence in first fragment	(15) Frequency gradient in first vowel of first fragment
(3) Language model (LM) score of first fragment	(11) Duration of head silence in second fragment	(16) Frequency range of first fragment
(4) Acoustic model (AM) score of first fragment	(12) Duration of first fragment	(17) Maximum loudness in first fragment
(5) Noise detection results by GMM	(13) Duration of final syllable of first fragment	(18) Maximum loudness in second fragment
(6) Overlap ratio of phoneme bigrams		
(7) Number of fillers in first fragment		
(8) Number of fillers in second fragment		

Bold: Effective features

as SVM is in our future work. The decision trees are built by J48 with its default parameter in machine learning software Weka¹.

In total, 18 features are used: eight from the ASR engine, five of timing, and five of prosody (Table 2). These are explained below with a focus on the five features, marked in bold, that were effective in our experiment. The numbering after each feature name corresponds to that in Table 2.

Features from ASR engine: (1)–(8)

We use a confidence measure (CM) score of the first fragment (1), which is obtained from ASR. The idea here is that an incorrectly-segmented utterance tends to have a low CM score, especially when a word is incorrectly segmented within it.

We also use noise detection results by a Gaussian mixture model (GMM) (5) constructed by Lee et al. [9]. This model classifies utterances into five classes: adults, children, laughter, coughing, and others. We set two values for this feature: “user utterance” if both fragments are classified as adults or children, because these two classes indicate normal utterances, and “noise” otherwise.

Timing features: (9)–(13)

We define an interval between fragments (9) as the time between the end of the first fragment and the start of the second. The idea here is that an originally-single utterance tends to have a shorter interval, as short pauses within utterances due to disfluency are shorter than intervals when a user’s utterances actually end. This tendency was confirmed in our previous study [2], where we found that fragment pairs with shorter intervals include more pairs that were originally a single utterance.

Prosodic features: (14)–(18)

The frequency range of the first fragment (16) is used for detecting noises with no harmonic structure. We also use the maximum loudness in the first fragment (17) to help detect the system’s synthesized voices, which are unintentionally mixed into the microphones and tend to be low loudness because the microphone was placed near the users. We use openSMILE² to obtain the prosodic features.

4.2. Two Kinds of Feature Selection

As stated earlier, the features used in the decision tree need to be effective also in other domains. We thus perform two kinds of feature selection:

1. Backward feature selection
2. Selection of domain-independent features

The backward feature selection aims to exclude features having a negative influence on classification [10]. First, we build a decision tree by removing a feature one by one and then

¹<http://www.cs.waikato.ac.nz/ml/weka/>

²<http://opensmile.sourceforge.net/>

compare its classification accuracy with the original one with all features. If the accuracy does not degrade without the feature, it is removed because it does not contribute to the accuracy.

To select features that are independent of domains, first, we build decision trees in both domains by ten-fold cross validation. If a feature is used in both the decision trees, that means it is effective in both domains and we regard it as not being dependent on either domain. We select such features as domain-independent ones.

5. Experimental Evaluation

To evaluate the classification accuracy, we performed cross-domain tests in addition to in-domain tests. The cross-domain test indicates that the decision tree is trained by one domain data and its accuracy is evaluated by the other domain data. This is to verify whether or not the obtained decision trees are dependent on any specific domain. All the in-domain tests were performed by ten-fold cross validation within one domain data. We performed four tests – two cross-domain tests and two in-domain tests – since we had two domains (restaurant and world heritage). Hereafter, “Cross” means results from the cross-domain test and “All” means total results from both the cross-domain and in-domain tests.

5.1. Results of Feature Selection

First, we clarify features that had a negative influence on decision trees by performing backward feature selection for all 18 features. Table 3 shows the change in the number of correct classification results when each feature was removed from all 18 features. The negative values in the table mean that the accuracy of the decision tree degraded when the corresponding feature was removed. From these results, we selected seven features ((1), (3), (5), (9), (12), (16), and (17)) that had negative values for the “All” condition in the table.

Next, the results of selecting domain-independent features are shown in Table 4. The numbers in the table indicate how many times each feature was used in each of the 10 decision trees. They thus correspond to the importance of each feature in the domains. Five features, marked in bold in the table, appeared in both domains and were regarded as domain-independent. We used these five as the selection result.

5.2. Classification Accuracy of Decision Trees

We compared the classification accuracies for the following three conditions: a baseline, “without feature selection”, and “with feature selection”. The baseline only used the interval between fragments (9), which corresponds to a simple rule using optimal thresholds for the interval. The “without feature selec-

Table 5: Classification accuracies of decision trees

	Restaurant	W.H.	Restaurant→W.H.	W.H. → Restaurant
Baseline	215/255 (84.3%)	288/354 (81.4%)	285/354 (80.5%)	209/255 (82.0%)
Without feature selection	219/255 (85.9%)	291/354 (82.2%)	289/354 (81.6%)	214/255 (83.9%)
With feature selection	230/255 (90.2%)	305/354 (86.2%)	302/354 (85.3%)	219/255 (85.9%)

W.H. denotes the world heritage domain.

Table 3: Changes in the number of correct results when each feature was removed

Removed feature	Cross	All
(1) Average CM score of first frag.	-5	-6
(2) CM score of last word of first frag.	0	0
(3) LM score of first frag.	1	-1
(4) AM score of first frag.	-3	6
(5) Noise detection results by GMM	-12	-3
(6) Overlap ratio of phoneme bigrams	0	1
(7) Number of fillers in first frag.	0	5
(8) Number of fillers in second frag.	0	1
(9) Interval between frags.	-130	-175
(10) Duration of tail silence in first frag.	0	1
(11) Duration of head silence in second frag.	4	10
(12) Duration of first frag.	-8	-21
(13) Duration of final syllable of first frag.	13	17
(14) Volume change in final part of first frag.	0	1
(15) Frequency gradient in first vowel	0	5
(16) Frequency range of first frag.	-4	-4
(17) Maximum loudness in first frag.	-10	-10
(18) Maximum loudness in second frag.	4	9

Bold: Features improving accuracy

Table 4: Number of occurrences of each feature in decision trees

Features \ Domains	Rest.	W.H.
(1) Average CM score of first frag.	4	1
(3) LM score of first frag.	4	0
(5) Noise detection results by GMM	9	10
(9) Interval between frags.	10	10
(12) Duration of first frag.	5	0
(16) Frequency range of first frag.	4	1
(17) Maximum loudness in first frag.	8	9

Bold: Effective features in both domains

Rest. and W.H. denote the restaurant and world heritage domains.

tion” condition used all 18 features listed in Table 2. The “with feature selection” condition used the five features obtained by the feature selection process, i.e., (1), (5), (9), (16), and (17).

Table 5 summarizes the classification accuracies of decision trees. “Restaurant” and “W.H.” are the results of 10-fold cross validation in each domain. “Restaurant → W.H.” and “W.H. → Restaurant” are the results of the cross-domain tests. For example, the former shows the result when the decision tree was trained on the restaurant domain data and its accuracy was calculated on the world heritage domain data. Our main objective is to improve the classification accuracy in the cross-domain tests, which are shown in the right half of Table 5, because the obtained decision tree should be domain-independent.

Under all conditions, the accuracies of “without feature selection” were slightly higher than those of the baseline. This indicates that the incorporated features were helpful for the classification. Furthermore, the accuracies of “with feature selection” were also higher than those of “without feature selection”.

Table 6: Changes in the number of correct results when each feature was removed from final feature set

Removed features	Cross	All
(1) Average CM score of first frag.	-9	-12
(5) Noise detection results by GMM	-30	-49
(9) Interval between fragments	-52	-124
(16) Frequency range of first frag.	-9	-11
(17) Maximum loudness in first fragment	-3	-14

In condition “Restaurant → W.H.”, the difference was statistically significant ($p = 0.00079$) by McNemar test, but that was not in the other cross-domain condition ($p = 0.38$). The results demonstrate that the two kinds of feature selections could successfully select effective and domain-independent features.

5.3. Analysis for Obtained Features

We performed an additional backward feature selection for the final five features to confirm their effectiveness. Table 6 summarizes the result. The numbers in the table indicate the change in the number of correct classification results when each feature was removed. Here, no features had positive values, indicating that no features had a negative influence. The classification accuracies significantly decreased under both the “Cross” and “All” conditions when we removed the (5) and (9) features. This indicates that the noise detection results by GMM (5) and the interval between fragments (9) were important.

6. Conclusion

We classified whether or not a posteriori restoration is required in order to restore mistakenly segmented utterances caused by VAD errors. We formulated this as a binary classification problem that determines whether a fragment pair was originally a single utterance or not. We used decision tree learning with various features for which two kinds of feature selection were performed. Results demonstrated that the obtained decision trees did not depend on any specific domain. They also outperformed the baseline in terms of the classification accuracy.

There remains future work. The features for the classification should be enhanced especially for prosodic features. We will also verify whether and how much the improvement of the classification accuracy affects the ASR accuracy of user utterances. This method will be implemented into the spoken dialogue system we have been developing [2]. A user study is also planned to collect more evaluation data and to verify the effect of the proposed method on the overall performance of the system, i.e., the task success rate.

7. Acknowledgments

This work was partly supported by JST PRESTO and the Naito Science & Engineering Foundation.

8. References

- [1] J. Hirasawa, M. Nakano, T. Kawabata, and K. Akiyama, "Effects of system barge-in responses on user impressions," in *Proc. EUROSPEECH*, 1999, pp. 1391–1394.
- [2] K. Komatani, N. Hotta, and S. Sato, "Restoring incorrectly segmented keywords and turn-taking caused by short pauses," in *Proc. IWSDS*, 2014, pp. 27–38.
- [3] A. Lee, K. Oura, and K. Tokuda, "MMDAagent - a fully open-source toolkit for voice interaction systems," in *Proc. IEEE-ICASSP*, 2013, pp. 8382–8385.
- [4] A. Benyassine, E. Shlomot, H. Yu Su, D. Massaloux, C. Lamblin, and J.-P. Petit, "ITU-T recommendation G.729 Annex B: a silence compression scheme for use with G.729 optimized for V.70 digital simultaneous voice and data applications," *IEEE Communications Magazine*, vol. 35, no. 9, pp. 64–73, 1997.
- [5] A. Raux and M. Eskenazi, "Optimizing Endpointing Thresholds using Dialogue Features in a Spoken Dialogue System," in *Proc. SIGDIAL*, 2008, pp. 1–10.
- [6] E. Jan, L. M. B. Maison, and G. Zweig, "Automatic construction of unique signatures and confusable sets for natural language directory assistance applications," in *Proc. EUROSPEECH*, 2003, pp. 1249–1252.
- [7] M. Katsumaru, K. Komatani, T. Ogata, and H. G. Okuno, "Adjusting occurrence probabilities of automatically-generated abbreviated words in spoken dialogue systems," in *Proc. IEA/AIE*, 2009, pp. 481–490.
- [8] M. Nakano, S. Sato, K. Komatani, K. Matsukawa, K. Funakoshi, and H. G. Okuno, "A two-stage domain selection framework for extensible multi-domain spoken dialogue systems," in *Proc. SIGDIAL*, 2011, pp. 18–29.
- [9] A. Lee, K. Nakamura, R. Nisimura, H. Saruwatari, and K. Shikano, "Noise robust real world spoken dialogue system using gmm based rejection of unintended inputs," in *Proc. ICSLP*, 2004, pp. 173–176.
- [10] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.