Dialogue Act Detection in Error-Prone Spoken Dialogue Systems
Using Partial Sentence Tree and Latent Dialogue Act Matrix

Wei-Bin Liang, Chung-Hsien Wu, Yu-Cheng Hsiao
Department of Computer Science and Information Engineering,
National Cheng Kung University, Tainan, Taiwan
(lliangnet, chunghsienwu, ychsiao9)@gmail.com

Abstract
In a goal-oriented spoken dialogue system, the major aim of spoken language understanding is to detect the dialogue acts (DAs) embedded in a speaker’s utterance. However, error-prone speech recognition often degrades the performance of the SLU component. In this work, a DA detection approach using partial sentence trees (PSTs) and a latent dialogue act matrix (LDAM) is presented for spoken language understanding. For each input utterance with speech recognition errors, several partial sentences derived from the recognized sentence can be obtained to construct a PST. The relationship between the GRs and the DAs is modeled by an LDAM. Finally, the DA with the highest probability estimated from the speech recognition likelihood, the LDAM and the historical information is determined as the detected DA. In evaluation, compared to the semantic slot-based method which achieved 48.1% dialogue act detection accuracy, the proposed approach can achieve 84.3% accuracy, with 35.2% improvement in accuracy.

Index Terms: Dialogue act, partial sentence tree, latent dialogue act matrix

1. Introduction
Dialogue acts often provide a useful evidence for characterizing dialogue behaviors in human-computer and human-human dialogues [1]. In the last decade, a variety of practical goal-oriented spoken dialogue systems (SDS) have been developed [2][3]. While DA detection can be achieved using several methods, including semantic role labeling [4], finite state-based [5], and semantic slot-based approaches [2][6], in this work, a DA detection approach using partial sentence trees (PSTs) [7] and a latent dialogue act matrix (LDAM) is proposed to alleviate the speech recognition error problem for robust dialogue act detection. In the proposed system, the dialogue key phrases, which are the main components to represent a dialogue act, should be manually defined first and the other words/phrases can be regarded as complementary and optional. For each recognized sentence from the speaker’s utterance, a partial sentence tree is constructed; each partial sentence should contain at least one dialogue key phrase. A set of sentence grammar rules (GRs) is generated for each partial sentence in the PST using the Stanford parser. The relationship between the GRs and the DAs is modeled by an LDAM.

2. Dialogue Act Detection
In this work on dialogue act detection, given the utterance $U$ and the dialogue historical information $DA_h$, the most likely $DA$ can be determined as follows.

$$DA^* = \underset{DA \in \Omega_{DA}}{\arg \max} P(DA | U, DA_h)$$

(1)

where $DA^*$ denote the detected DA; $\Omega_{DA}$ represents the set of the DAs; $DA_h$ is the $q$-th possible DA, and $DA_h$ represents the historical information. Although several word sequences can be decoded from the utterance $U$, we only adopt the best ASR output $\hat{W} = [\hat{w}_1, \ldots, \hat{w}_M]$ over all possible output $\Omega_k$ for DA detection. Therefore, Eq.(1) is re-written as Eq. (2) and further expanded to Eq. (3):

$$DA^* = \underset{DA \in \Omega_{DA}}{\arg \max} \sum_{\hat{W} \in \Omega_k} P(\hat{W} | U, DA_h)$$

(2)

$$= \underset{DA \in \Omega_{DA}, \hat{W} \in \Omega_k}{\arg \max} P(DA | \hat{W}, U, DA_h)$$

(3)

Generally, since $\hat{W}$ and $U$ convey the same information and the ASR output is independent of $\Omega_k$, $DA$ detection can be obtained from Eq. (4). Besides, the term $P(\hat{W} | U)$ in Eq. (4) can be rewritten according to the Bayes’ rule into Eq. (5):

$$DA^* = \underset{DA \in \Omega_{DA}, \hat{W} \in \Omega_k}{\arg \max} \left[ \frac{P(\hat{W} | DA_h)P(DA_h)}{P(U)} \times \frac{P(\hat{W} | DA_h)P(DA_h)}{P(U)} \right]$$

(4)

In the detection mechanism, the probability $P(\hat{W})$ is the same for all DAs and $P(DA_h)$ is the prior probability which has also the same value for all DAs. Thus, the $DA$ is determined according to the following equation:

$$DA^* = \underset{DA \in \Omega_{DA}, \hat{W} \in \Omega_k}{\arg \max} P(\hat{W} | DA_h)P(DA_h)$$

(6)

where $P(\hat{W} | DA_h)$ represents the probability of $\hat{W}$ classified to the $q$-th DA. This value can be estimated using the proposed LDAM and will be described in the following section. The second term $P(DA_h | DA_l)$ is the conditional probability of the $q$-th DA given the dialogue historical information, which can be calculated from the training corpus. Finally, $P(\hat{W} | U)$ provides the likelihood of the recognition output $\hat{W}$ from the ASR given the utterance $U$. 

Copyright © 2010 ISCA 3038 26 – 30 September 2010, Makuhari, Chiba, Japan
Figure 1 illustrates the training and test phases of the proposed approach. In the training phase, manual transcription and ASR output of the input speech utterance are both used to construct a PST for a sentence. In the PST, each path traversing from the root node to a leaf node, represents a partial sentence (PS). Next, the dialogue key phrases of each PS are substituted by their corresponding name entity class and then the Stanford parser (S-Parser) [7] is employed to parse each partial sentence to obtain a set of GRs. The relationship between GRs and DAs is modeled by a matrix LDAM. Finally, a probabilistic model in Eq. (6) is employed to detect the optimal dialogue act given the input speech utterance.

### 2.1. Partial Sentence Tree

One of the major factors for performance degradation in an SDS is due to speech recognition errors. Fortunately, a sentence can convey the latent meaning even some words are lost. Therefore, it is reasonable that a sentence is composed of at least one key phrase (KP) which can represent a DA and the other phrases are regarded as the optional phrases (OPs). So, we can define a partial sentence as a sequence of OPs and KPs:

$$PS=OP_1,\ldots,OP_{NB},KP, OP_{NB+1},\ldots,OP_{NB+N}$$

where $NB$ and $NA$ denote the numbers of OPs before and after the KP respectively. Then, a $PS$ can be a fragmented sentence in which the KP is preserved and the OPs in the ASR output verified as unreliable will be rejected and therefore omitted. The remaining word sequence forms a PS. Thereafter, a PST [8] is constructed from all PSs. Figure 2 is an example of the PST with four PSs. In this example, the whole sentence is “Where is the Anping Fort?” with the phrases “Where” and “Anping Fort” being the KPs. Using the PST for eliminating the speech recognition error problem generally increases the computation cost. Thus, a simple statistical measure, $z$-score [9], is employed to verify the confidence of the recognized words.

The $z$-score of word $w_m$ is defined as:

$$z\text{-score} = \frac{\text{score}(w_m) - \mu_{wm}}{\sigma_{wm}}$$

If the $z$-score of word $w_m$ is below a threshold, this word will be rejected and replaced by a substitute term, “Filler”. In addition, a succession of the rejected words is replaced by only one substitute term because a sequence of substitute terms contains only one “Filler” for further process.

Following PST construction, S-Parser is utilized to convert each PS to the corresponding grammar rules for GR generation. Assuming that totally there are $L$ sentence GRs obtained from the training corpus, an $L \times 1$ vector $R$ is constructed to record which GR the $PS$ will use. For example, a vector $R=[0,1,1]$ represents $L=3$ and the second and third rule are used in this sentence.

### 2.2. Dialogue act modeling and detection

#### 2.2.1. LDAM training

A paradigm of vector space method, named Latent Semantic Analysis (LSA), useful to extract and represent the meaning of words by statistical computations applied to a large corpus of texts, was proposed in [10]. In this work, the LSA is adopted to construct a GR-DA matrix which describes the occurrences of $L$ GRs in $Q$ DAs. Typically, the weightings of the elements of the matrix are log entropy [11]. However, one key problem is to determine the size of $Q$. Although the tasks (e.g. information of sightseeing spots and railway timetable) can be predefined, in order to increase the flexibility of a tour-guide SDS, the speakers can speak out different sentences even the speakers inquire the same information of a sightseeing spot. Therefore, determining the size of $Q$ through sentence type clustering is the goal of this work. Herein, the spectral clustering algorithm [12], one of the most popular modern clustering algorithms, is employed to determine the size of $Q$. The most common spectral clustering algorithm can be described as follows.

Given a set of $N$ data points $R_1,\ldots,R_{N}$, the $N \times N$ similarity matrix $S$ contains the elements $s_{ij}\geq 0$, each representing a measure of the similarity between two GR vectors $R_i$ and $R_j$. In this work, the similarity measure can be estimated using the Euclidean distance as:

$$S_{\text{Euclidean}}(R_i, R_j) = \left\| R_i - R_j \right\|^2$$

or the cosine distance measure:

$$S_{\text{cosine}}(R_i, R_j) = \frac{R_i \cdot R_j}{\| R_i \| \cdot \| R_j \|}$$

where $\| \cdot \|$ denotes the norm of vector $R$. Spectral clustering algorithm makes use of spectrum of the similarity matrix $S$ of the data to cluster the data. Sometimes such techniques are also used to perform dimensionality reduction for clustering in fewer dimensions. After constructing the matrix $S$, the element $s_{ij}$ will not be zero if only if the value of $s_{ij}$ is above the threshold. Then, if the matrix $S$ is regarded as a graph that the elements are the vertices and the similarity represents the weight of $s_{ij}$, the spectral algorithm partition the data points into $k$ sets ($S_1,\ldots,S_k$) based on the eigenvectors $u_1,\ldots,u_k$. 

![Figure 1: The framework for DA detection in spoken language understanding. The solid arrows show the function flow in the training phase and the dashed arrows are for the test phase.](image)
corresponding to the first \( k \) eigenvalues \( \lambda_1, \lambda_2, \ldots, \lambda_k \) of the Laplacian matrix \( A \) of the similarity matrix \( S \) represented as

\[
A = D - S 
\]

(11)

where \( A \) depicts the connections of vertices, \( I \) is the unit matrix, and degree matrix \( D \) is a diagonal matrix with the degrees \( d_1, \ldots, d_n \) on the diagonal which contains information about the number of data point connected to each data point. The degree can be defined as:

\[
D_k = \sum_i d_i 
\]

(12)

This partition may be done in various ways, such as by taking the eigenvectors corresponding to the \( k \) largest eigenvalues of the matrix \( A \) for some \( k \), and then invokes the \( k \)-means algorithm to cluster the data points by their respective \( k \) components in these eigenvectors. Hence, totally \( Q \) DAs for all tasks are obtained after clustering. The final \( L \times Q \) matrix \( LDAM \) is shown in figure 1. The value of the entry \( (l,q) \) in \( LDAM \) is estimated as:

\[
\phi_{l,q} = (1-e_l) \frac{f_{l,q}}{\sum_{q'=1}^{Q} f_{l,q'}} 
\]

(13)

where \( f_{l,q} \) represents the frequency of the \( l \)-th GR appearing in the \( q \)-th DA over all GRs in \( DA_q \), and \( e_l \) is the normalized entropy of \( DA_q \), computed as

\[
e_l = -\frac{1}{\log Q} \sum_{q'=1}^{Q} \frac{f_{l,q'}}{\sum_{q'=1}^{Q} f_{l,q'}} \log \frac{f_{l,q'}}{\sum_{q'=1}^{Q} f_{l,q'}} 
\]

(14)

2.2.2. Dialogue act detection

Even though the PSs are the fragmented ASR output, a PS generally still comprises the GRs and name entities. Therefore, the term \( P(\tilde{W}|DA_q) \) in Eq. (6) could be approximated by the \( i \)-th PS with the highest score and decomposed as

\[
P(\tilde{W}|DA_q) \approx \max_{PS \in PS'} P(PS | DA_q) 
\]

(15)

where \( P(R|DA_q) \) and \( P(NE|DA_q) \) are the probabilities of grammar rule \( R \) and name entity \( NE \) in the \( q \)-th dialog act, respectively. For the term \( P(R|DA_q) \), the cosine distance measure is employed to estimate the probability:

\[
P(R | DA_q) = \frac{R^T D_{Aq}}{|R| \sqrt{|D_{Aq}|}} 
\]

(16)

where \( D_{Aq} \) is the \( q \)-th column of the \( LDAM \). For the term \( P(NE | DA_q) \), \( NE \) is an \( N \times 1 \) vector and \( Z \) denotes the number of KPs in \( PS \). Assuming these KPs are statistically independent, the following probability could be approximated as:

\[
P(NE | DA_q) = \prod_{n=1}^{N} P(NE_{n,q}^* | DA_q) 
\]

(17)

where \( P(NE_{n,q}^* | DA_q) \) denotes the probability of the \( z \)-th KP appearing in the \( PS \) in dialogue act \( DA_q \).

2.3. Historical information for DA detection

Imperfect speech recognition may result in diverse DA of each turn in a dialogue and therefore severely damage the performance of an SDS. Herein, a strategy that concatenates historical DA information in the speaker’s previous utterances to help DA detection is proposed. Let \( DA_t \) denote the DA at time \( t \) and the historical DA information be represented as \( DA_{t-1} = \{DA_{t-1,1}, DA_{t-1,2}, \ldots, DA_{t-1,k} \} \). For simplicity, this study assumes that \( DA_t \) is only dependent on its previous correct one confirmed by the speakers. Hence, the historical information for DA detection is defined as:

\[
P(DA_t | DA_{t-k}) = P(DA_t | DA_{t-k}) 
\]

(18)

Finally, Eq. (6) is adopted to determine the DA according to the probabilities \( P(W|DA_t), P(DA_t|DA_{t-k}) \) and \( P(W|U) \).

3. Experiments and Discussion

For the evaluation of the proposed approach, one spoken dialog corpus for the travel information task was collected. This corpus collected the speech utterances from six male and two female subjects. The utterance collection system was basically an SDS that could interact with the speakers by asking/answering questions and recording the speakers’ utterances. Totally, 144 dialogues consisting of 1586 sentences \( (N=1586) \) were collected. After data analysis, 32 name entity classes (NEC), 796 GRs \( (Q=38) \) and 796 DAs \( (L=796) \) were included. Some NEC and DA labels are listed in Table 1. The average turn number of a dialogue is 8.6 and the average length of the speaker’s utterance is 2.3 seconds. Moreover, each speech utterance was manually annotated and the speech recognition results were obtained by an HTK-based speech recognition engine. The speaker-independent ASR recognition rate achieved 86.1% (with a lexicon of 297 words). The travel information for the SDS was collected from the travel databases available on the web (e.g. Wikipedia and Yahoo!Travel). The collected data covered four tasks, including one historic spot information system, two railway systems in Taiwan, and greeting words for human-robot interaction. Figure 3 is an example of GRs obtained from the S-Parser. Although S-Parser cannot always guarantee the parsing results of the patterns in a PST, most of PSs are consistent in GRs. All evaluations were performed using 5-fold cross-validation.

Figure 4 shows the evaluation results of four cases under different ASR performances, including the semantic slot-based approach (baseline) and the other three cascaded cases. To control the influence of experimental conditions, the DAs are obtained from the proposed approach for evaluation. Moreover, the cosine distance is adopted in spectral clustering algorithm. The baseline labeled as “(1) baseline”, the name entities and DAs were directly used to build the LDAM without GR generation and induction. Contrast to the baseline, the first cascaded case labeled as “(2) LDAM+GR” is that the LDAM was constructed by GRs obtained from the S-Parser and the DAs without PST. The second cascaded case labeled as “(3) LDAM+GR+PST” was designed to compare with the second approach so that the PST was conducted in this evaluation. The last case labeled as “(4) Proposed” (LDAM+GR+PST+NEC) is the proposed approach. “Manual” means that manual transcriptions were used for the detection task directly. On the other hand, we randomly replaced different proportional number of phrases of a transcription to simulate the ASR performance. These simulation cases were labeled as “simulated-100%”, while the proposed approach achieved 84.3% recognition accuracy. Compared to the baseline, the proposed method obtained a significant improvement, especially in Traffic DA. Moreover, the baseline is a slot-based approach. In “simulated-40%, representing 40% recognition accuracy” and “simulated-60%”, the best result was obtained in Case (2) because almost half of the recognized phrases even the MPs were rejected and the parsing outputs were different from the expected GRs. If we preserve the KPs in both evaluations, Cases (3) and (4) are better than Case (2). Comparison between Cases (3) and (4),
NEC can contribute to the performance of DA detection, especially the scenic spot class comprising unknown words in S-Parser.

Figure 5 illustrate the comparison results among slot-based, N-gram of slots based on the a priori algorithm [13], LSA with log entropy weighting after k-means clustering for DA and the proposed approach. In fact, the N-gram approach considers the co-occurrence of slots (name entities) and improves the performance of the inquiry. The last two approaches were evaluated based on PST and GR. The results show that PST and GR contribute to the performance of the proposed approach for the condition of imperfect ASR.

4. Conclusions

In this work, a DA detection approach using PSTs, GRs and an LDAM is presented. Considering speech recognition errors, each recognized sentence can be expanded into several partial sentences to construct a PST. Each partial sentence in the PST is parsed into a set of GRs using the S-Parser. The relationship between the GRs and the DAs is modeled by an LDAM. The size of DA is determined by the spectral clustering algorithm using cosine distance as the similarity measure. Finally, the DA with the highest probability estimated from the recognition likelihood, the LDAM and the DA historical information is determined as the detected DA. In evaluation, the proposed approach can achieve 84.3% in accuracy, compared to the semantic slot-based method which achieved only 48.1% DA detection rate. Compared to the slot-based and N-gram method, the proposed approach can obtain the best performance of DA detection under the condition of imperfect ASR.

Figure 3: An example of GRs obtained from S-Parser

Table 1: Examples of NEC and DA labels

<table>
<thead>
<tr>
<th>NEC</th>
<th>MP</th>
<th>Type</th>
<th>DA Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>City</td>
<td>Tainan, Taipei, Kaohsiung etc.</td>
<td>Ask</td>
<td>System_Info, Spot, Station, etc.</td>
</tr>
<tr>
<td>Date</td>
<td>Today, Tomorrow, Yesterday etc.</td>
<td>Greeting</td>
<td>Hello, Goodbye, etc.</td>
</tr>
<tr>
<td>Time1</td>
<td>o’clock, Noon, to, Noon, past, etc.</td>
<td>Traffic</td>
<td>Bus, Start, TRA, Start, TSR, Start, etc.</td>
</tr>
<tr>
<td>Time2</td>
<td>Morning, Noon, Afternoon, etc.</td>
<td>Request</td>
<td>Address, Ticket, Phone, Open, Time, etc.</td>
</tr>
</tbody>
</table>

Figure 4: Average detection accuracy of four cascaded cases under different ASR accuracy. "ASR" is the real ASR performance.

Figure 5: The comparison of (a) slot-based, (b) N-gram word, (c) k-means + LSA with log entropy weighting and (d) the proposed approaches.

5. References


