ELSEVIER

Speech Communication 38 (2002) 183–199

SPEECH COMMUNICATION

www.elsevier.com/locate/specom

Speech act modeling in a spoken dialog system using a fuzzy fragment-class Markov model

Chung-Hsien Wu *, Gwo-Lang Yan, Chien-Liang Lin

Department of Computer Science and Information Engineering, National Cheng Kung University, 1 Ta-Hsueh Road, Tainan, Taiwan, ROC

Received 11 May 2000; received in revised form 5 June 2001; accepted 6 June 2001

Abstract

In a spoken dialog system, it is an important problem for the computer to identify the speech act (SA) from a user's utterance due to the variability of spoken language. In this paper, a corpus-based fuzzy fragment-class Markov model (FFCMM) is proposed to model the syntactic characteristics of a speech act and used to choose the speech act candidates. A speech act verification process, that estimates the conditional probability of a speech act given a sequence of fragments, is used to verify the speech act candidate. Most main design procedures are statistical- and corpus-based to reduce manual work. In order to evaluate the proposed method, a spoken dialog system for air travel information service (ATIS) is investigated. The experiments were carried out using a test database from 25 speakers (15 male and 10 female). There are 480 dialogs, containing 3038 sentences in the test database. The experimental results show that the speech act identification rate can be improved by 10.5% using the FFCMM and speech act verification with a rejection rate of 6% compared to a baseline system.

Keywords: Spoken language; Speech act; Fuzzy fragment-class Markov model; Verification

1. Introduction

In recent years, the domain of spoken dialog has been broadly researched. Many application systems such as air travel information services (ATIS), weather forecast systems, automatic call managers, and railway ticket reservations have been presented (Meng et al., 1996; Bennacef and Lamel, 1996; Seide and Kellner, 1997; Lee et al., 1997; Chiang et al., 1998; Wang et al., 1997). However, there are still many problems, which make the dialog system unnatural. The main problem is that it is not easy to get enough knowledge to represent the real meaning of a sentence. In spoken language, utterances contain a speech act (SA) (Allen, 1994) and descriptive information (Saeki et al., 1996). These two features are generally used to determine the intention of an input utterance.

Some approaches that use a large set of rules to explain the syntactic and semantic possibilities for spoken sentences suffer from a lack of robustness when faced with the wide variety of spoken sentences that people really use. Generally, these systems have low capability to identify the exact speech act from the large number of sentence

* Corresponding author. Tel.: +886-6-2757-575; fax: +886-6-2747-075.

0167-6393/02/$ - see front matter
PII: S0167-6393(01)00052-8

candidates generated from a speech recognizer based on an incomplete set of rules. The derivation of syntactic and semantic rules is labor intensive, time consuming and tedious. Furthermore, because many of the various spoken sentences have different syntactic and semantic characteristics, it is difficult to collect appropriate and complete rules to describe the syntactic and semantic diversity. In keyword-based approaches (Saeki et al., 1996), keywords are collected for a specific speech act to identify the intention from a sentence. This approach lacks contextual information and the syntactic structure of a sentence. In some other approaches using statistical methods, the bigram or trigram probability models (Harksoo Kim et al., 1999) are generally used to find the appropriate speech act. These kinds of approaches face the problem that local syntactic characteristics (bigram/trigram) cannot describe the speech act very well without considering the syntactic and semantic structure of the entire sentence. Another approach using a statistical representation of task-specific semantic knowledge, such as HMM (Pieraccini et al., 1992), extracts the words and their association to the conceptual structure of the task. The proposed HMM functions under the assumption that the acoustic model of a word is independent of the concept it expresses. HMM's states represent concept relations and its observation probabilities constitute state-local language models in the form of bigrams of words. This approach may face the ambiguity that the same word, expressing different concepts, has the same acoustic representation in real language. Still the state-local language models of the observation probabilities in the form of bigrams cannot model the long distance characteristics of a sentence well.

In our approach, a statistical speech act model, called a fuzzy fragment-class Markov model (FFCMM), is proposed to model the statistical syntactic characteristics of a speech act to assist in the analysis of spoken language. The FFCMM's states represent the fragment classes (FCs) clustered using a fuzzy C-means algorithm (Zimmermann, 1991). The state transition probability of the FFCMM models the syntactic transition of FC in a specific speech act. In other words, the FFCMM models not only the FC's concept, but also the syntactic transition of the FCs in a speech act. Finally, the speech act verification estimates the verification score using the conditional probability derived from the relationship between speech acts and fragments. In addition, a fragment extraction method and a speech act modeling method are proposed to construct the speech act model semi-automatically from a training corpus.

For performance evaluation, a spoken dialog system for an ATIS was investigated. The architecture of this system is shown in Fig. 1. There are four main components in this system. In the speech recognition component, the input speech is recognized into possible fragment sequences according to a fragment dictionary and a fragment bigram language model. The semantic analysis component analyzes the fragment sequences by identifying the speech act using 30 FFCMMs and outputs the probable speech act candidates. For each speech act candidate, a verification score is used to accept/reject the speech act. The accepted speech act with the highest score is chosen as the output to the dialog component. In the dialog component, the dialog manager accepts the results from the semantic analysis component to generate the response sentence based on a dialog scenario defined in (Wu et al., 1998). In the response component, the text-to-speech (TTS) (Wu and Chen, 1999) module generates the speech response to the user according to the response sentence from the dialog component.

This paper is organized as follows. In Section 2, we describe how to analyze and cluster the fragments. In Section 3, the definition and construction of the FFCMM is depicted. Speech act verification is provided in Section 4. The experimental results are shown in Section 5. Finally, a brief conclusion is presented in Section 6.

2. Fragment analysis

2.1. Corpus collection

Corpus collection is a key issue for the corpus-based approach. In order to collect the corpus of spoken language in human–machine interaction,

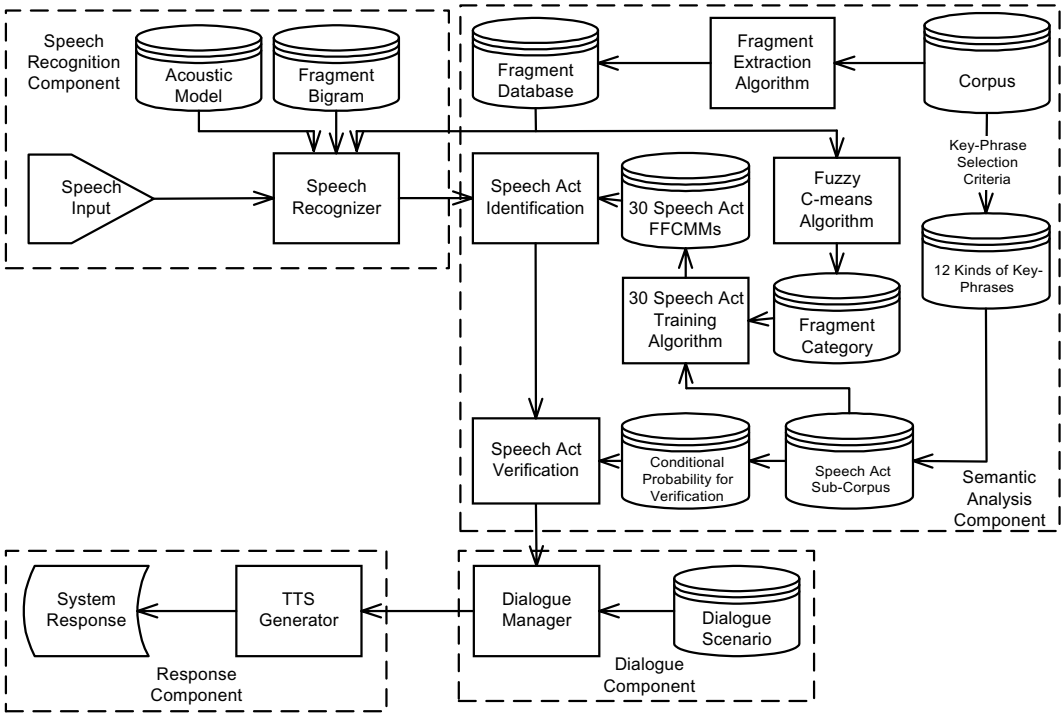

Fig. 1. Dialog system architecture.

the corpus collection was developed in two steps. In the first step, the initial corpus was transcribed from recordings from an actual dialog environment. The corpus approximated 4250 spontaneous utterances collected in fluent spoken dialog and real human interactions. These data were transcribed and used to create an initial version of the ATIS system. In the second step, the database was augmented in a wizard environment with subjects brought into the lab and given scenarios to solve. We collected over 2200 utterances from this prototype system. The second corpus was collected in interactions between humans and the machine. In a practical application of human–machine interaction, the second corpus is helpful for improving the system performance because it contains the various habits and behaviors that occurred when people faced a computer service system.

These two corpora collected from the above procedure were in a speech format. These corpora were then orthographically transcribed and tagged into 71 and 30 KB text corpus separately (101 KB totally). There were many kinds of sentence patterns such as greetings, acknowledgements, inquiries, describing dates and time. These paradigms were all transcribed and tagged as clearly as possible so that these marks and symbols preserve the grammar of the spoken language. The characteristics of these two corpora are listed in Table 1. Generally, the average length of the sentences and the bigram perplexity, calculated according to 206 fragments extracted by the fragment extraction algorithm described in Section 2.2, in the human–machine corpus was smaller than that in the human–human corpus. This is because natural spoken language would carry more information and be more complex in a sentence spoken by a user. If the user found that he/she was talking with a computer, he/she would speak shortly and simply (Riccardi and Gorin, 2000). This effect also affected the average turns so that the computer ran more turns than an operator in ATIS. We collected more than 6500 sentences as the training corpus. The distribution of the number of dialogs in ATIS is

Table 1
The characteristic of the collected corpus

	No. of sentences	Memory size	Average sentence length	Average no. of turns	Bigram perplexity
Human–human corpus	4246	71 KB	7.3	4.3	8.1
Human–machine corpus	2287	30 KB	5.6	6.3	6.8
Total corpus	6533	101 KB	6.7	5.0	7.6

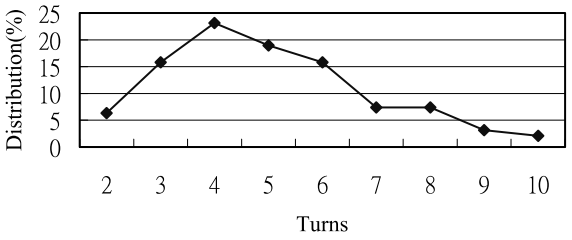

Fig. 2. Distribution of the number of turns.

shown in Fig. 2. Most of the dialogs contain 3–6 turns.

2.2. Fragment extraction algorithm

In a spoken sentence, the phrases or words are not the same as in a written sentence. They are generally a combination of some of the words or characters defined in the lexicon. A fragment is defined as a combination of some words or characters that generally appear together in a specific domain. In this study we adopted a fragment extraction algorithm (Lai and Wu, 2000) to extract fragments from a task-specific corpus. In this algorithm, for each word sequence $p = w_1 w_2 \dots w_n$ in a corpus, a likelihood ratio $r_i(p)$, which is quite related to the standard mutual information measure is defined as

$$r_i(p) = \frac{f(p)}{\min_i\{f(w_1 w_2 \dots w_i), f(w_{i+1} w_{i+2} \dots w_n)\}}, \quad 1 \leqslant i < n, \tag{1}$$

where $f(p)$ is the appearance frequency of p in the corpus, $f(w_1 w_2 \dots w_i)$ is the appearance frequency of the first i words in p, and $f(w_{i+1} w_{i+2} \dots w_n)$ is the appearance frequency of the last $n - i$ words in p. The likelihood ratio $r_i(p)$ is an important cue and represents the relationship between $w_1 w_2 \dots w_i$ and $w_{i+1} w_{i+2} \dots w_n$. Now, we define a fragment using the likelihood ratio as follows:

Given a word sequence $p = w_1 w_2 \dots w_n$ extracted from the corpus, if the following conditions hold simultaneously, we call p a fragment:

1. $n > 1$,
2. $f(p) \geqslant c$, and
3. $\exists i$, $r_i(p) \geqslant 1 - \varepsilon$ or $\exists i$, $r_i(p) \cdot f(p) \geqslant d$,

where c is constant to limit the appearance frequency of fragment p, ε is a tolerance limit and approximates zero, and d is a lower boundary.

The first two conditions are trivial. The third condition is related to the standard techniques of entropy minimization. It has two parts separated by the operation "or". The first part means that if we can find a position in which p can be separated into two-word sequences and if $r_i(p)$ approximates one, then p is a fragment. In the second part, if p has either a high appearance frequency or high likelihood ratio, then p is defined as a fragment. In our approach, we followed the methods proposed in (Lai and Wu, 2000) to choose the values of the three parameters c, ε and d as 30, 0.1 and 25, respectively. These values achieved the best performance of precision rate and recall rate for fragment extraction according to the experimental results described in (Lai and Wu, 2000). In total, 206 fragments were extracted according to the above algorithm to construct a task-specific fragment dictionary. The 206 fragments contain fourteen city names, seven airlines and other frequently used words in ATIS. Their frequency distribution is shown in Fig. 3. There are still some additional fragments that are not in the fragment dictionary. They are defined as out-of-vocabulary words and can be treated as garbage in this domain.

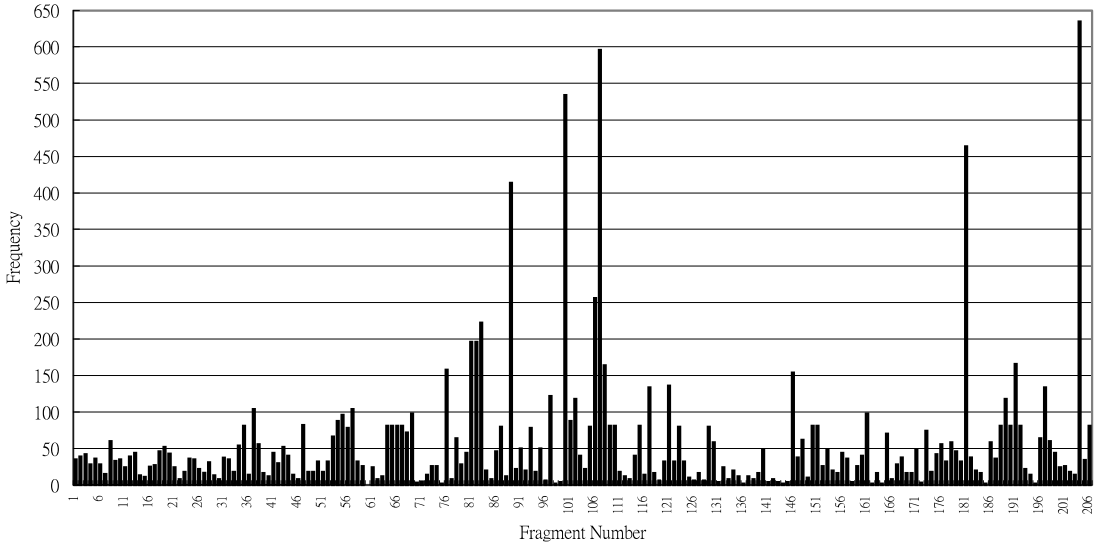

Fig. 3. Fragment frequency distribution.

2.3. Fragment class

The idea of FC is similar to the grammar fragment proposed in (Wright et al., 1997) which places fragments together when they are different as strings but similar semantically. An FC is a category with fragments that are similar in syntactic characteristics. The syntactic characteristic of a fragment is observed by the position that the fragment appears in a sentence. Although other fragments in a sentence might affect the syntactic characteristic of a fragment, in this study, we considered the preceding and succeeding fragments only. For each fragment, the relation to its preceding and succeeding fragments is used as the feature to cluster the fragments into FCs. This feature, called a bi-directional bigram vector, is defined as follows:

For a fragment *Frag*, the *i*th dimension of the bi-directional bigram vector represents the frequency of the fragment Fr_i just preceding (or succeeding for the opposite direction) the fragment Frag in the training corpus. Consequently, the preceding bigram frequency can be denoted as $f(\mathrm{Fr}_i, \mathrm{Frag})$ and the succeeding bigram frequency can be denoted as $f(\mathrm{Frag}, \mathrm{Fr}_i)$. The bi-directional bigram vector for the fragment Frag can be expressed as

$$\begin{aligned}\mathrm{BBV}(\mathrm{Frag}) = [&f(\mathrm{Fr}_1, \mathrm{Frag}), f(\mathrm{Fr}_2, \mathrm{Frag}), \ldots,\\ &f(\mathrm{Fr}_d, \mathrm{Frag}), f(\mathrm{Frag}, \mathrm{Fr}_1), f(\mathrm{Frag}, \mathrm{Fr}_2), \ldots,\\ &f(\mathrm{Frag}, \mathrm{Fr}_d)],\end{aligned} \tag{2}$$

where d is the total number of fragments. The BBV's concept is the same as the syntactic association proposed by Arai et al. (1999). It is useful for syntactic clustering. In our approach, we extracted 206 fragments using the fragment extraction algorithm (Lai and Wu, 2000). The total dimension is 412 for the bi-directional bigram vector.

In the clustering process, a class can be seen as a part-of-speech (POS) that contains the fragments with the same syntactic representation and syntactic usage in that language. We assumed that every fragment belongs to one or more classes and a fuzzy membership function is used to represent the degree a fragment belonged to a class. The idea of a fragment belonging to many classes (Jelinek et al., 1990) has been described in many papers in recent years. These papers equalized the weights of the fragment belonging to different classes. It is not reasonable from the viewpoint of statistics. Basically, a fragment should have different syntactic representations with different weights. A seldom-used syntactic representation should have a lower

weight. For example, the Chinese fragment "查詢" has two syntactic representations which are verb type and noun type. Conventionally, the verb type of "查詢 (inquire)" is more frequently used than the noun type "查詢 (inquiry)". In order to model this characteristic, the fuzzy C-means algorithm (Zimmermann, 1991; Tran et al., 1998a,b) was adopted for fragment clustering. The fuzzy C-means algorithm is one of the best-known algorithms for the clustering problem and minimizes the overall average distortion. It provides a good method to calculate different weights denoted as membership for their importance to each class. This is different from other vector quantization methods like the k-means algorithm. The fuzzy objective function Z_m (Zimmermann, 1991) can be defined by the least-square function as

$$Z_m(U,\mu;\mathrm{Frag}) = \sum_{t=1}^{T}\sum_{i=1}^{I} u_{it}^m d_{it}^2. \tag{3}$$

The entry u_{it} in the membership matrix U is the fuzzy membership of the BBV of the fragment Frag_t with respect to the class vector μ_i, $i = 1,\ldots,I$ and satisfies

$$0 \leqslant u_{it} \leqslant 1 \quad \text{and} \quad \sum_{i=1}^{I} u_{it} = 1, \tag{4}$$

where $m(>1)$ is the exponential weight, and d_{it} is the measure of distance defined in the following:

$$\begin{aligned} d_{it} &= \cos(0) - \cos(\theta_{\mathrm{Frag}_t,\mu_i}) \\ &= 1 - \left(\frac{\mathrm{BBV}(\mathrm{Frag}_t)\cdot\mu_i}{|\mathrm{BBV}(\mathrm{Frag}_t)|\times|\mu_i|}\right), \end{aligned} \tag{5}$$

where $|\cdot|$ refers to the Euclidean norm and $\times$ refers to multiplication. The distance d_{it} is defined based on the cosine value of the angle between the two vectors $\mathrm{BBV}(\mathrm{Frag}_t)$ and μ_i with the absolute cost to $\cos(0)(=1)$, which is the best condition between two vectors.

The basic idea in the fuzzy C-means algorithm is to minimize Z_m over the membership matrix U and class vectors μ_i, $i = 1,\ldots,I$. The update functions (Zimmermann, 1991) to minimize Z_m are as follows:

$$u_{it} = \frac{\left(1/d_{it}^2\right)^{1/(m-1)}}{\sum_{i=1}^{I}\left(1/d_{it}^2\right)^{1/(m-1)}}, \tag{6}$$

$$\mu_i = \frac{\sum_{t=1}^{T} u_{it}^m \mathrm{BBV}(\mathrm{Frag}_t)}{\sum_{t=1}^{T} u_{it}^m}. \tag{7}$$

This algorithm is described in the following:

Fuzzy C-means algorithm:

Step 1. Initialize I to 1 and the initial class c_1 to $\mathrm{BBV}(\mathrm{Frag}_1)$. Select a radius R and m $(m > 1)$.

Step 2. For each input $\mathrm{BBV}(\mathrm{Frag}_t)$ $(t = 1, 2,\ldots,T)$, sequentially calculate the following equation:

If $\cos(0) - \cos(\theta_{\mathrm{Frag}_t,c_i}) \leqslant R \quad (i = 1,2,\ldots,I)$, then set $\mathrm{BBV}(\mathrm{Frag}_t) \in c_i$, else create a new class c_{I+1} to $\mathrm{BBV}(\mathrm{Frag}_t)$ and set $I = I + 1$.

Step 3. Compute distance d_{it} using Eq. (5).

Step 4. Update matrix U using Eq. (6) and calculate the new class vectors using Eq. (7).

Step 5. Stop if the decrease in the value of the fuzzy objective function Z_m at the current iteration relative to the value of Z_m at the previous iteration is below a chosen threshold. Otherwise, go to Step 2.

In our system, R was chosen according to the system performance. The experimental result is described in Section 5.3. Using the fuzzy C-means algorithm, the fragments with similar syntactic structures are grouped into the same class and have similar membership functions. Fig. 4 shows the membership functions for the three fragments – "上午(morning)", "下午(afternoon)" and "晚上(evening)" in the same class. In total, we have 38 classes, each representing one FC, using the bi-directional bigram vector.

2.4. Perplexity of FC bigram model

Perplexity is an important metric to evaluate language models. In recent years, many clustering methods have been proposed to minimize the

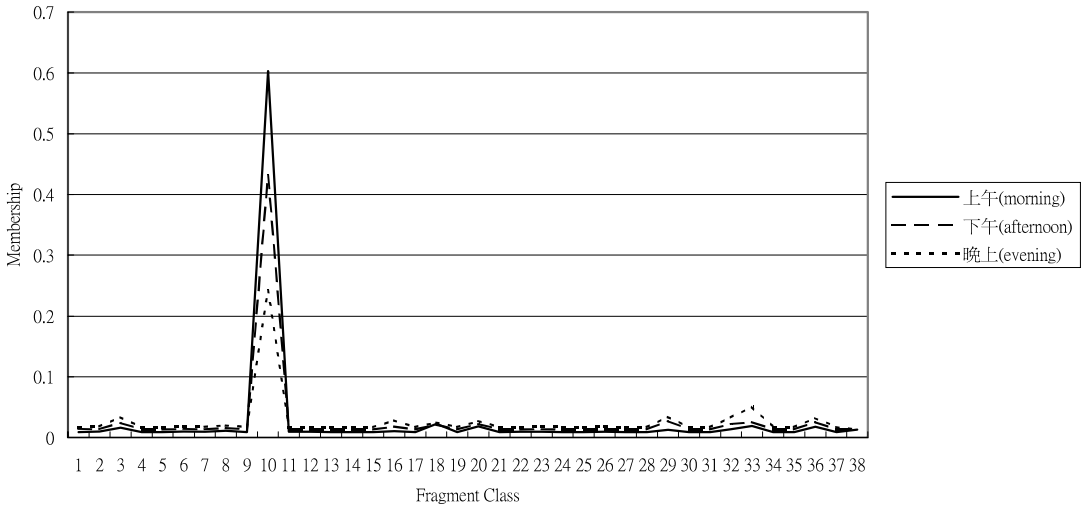

Fig. 4. Membership values of the three fragments – "上午 (morning)", "下午 (afternoon)" and "晚上 (evening)" in the same class.

perplexity (Martin et al., 1998). Some of these approaches assumed that every word could belong to only one class. However, in many cases, each word can belong to several classes (Jelinek et al., 1990) due to POS or the position that the word appears in a sentence. Besides this, the concept of a word belonging to many classes is widely used in many other clustering methods. However, the weights for these clustered classes are generally assigned as equal. This offsets the benefit of assigning a single word to many classes. Therefore, in the proposed approach, we assumed that every fragment can be assigned to several classes and the membership denotes the degree that the fragment belongs to a class. We redefined the perplexity according to the original definition of perplexity, the average word branching factor of the language model.

For the fragment sequence uv, the traditional class bigram is defined as

$$P(C_j|C_i) = \frac{\sum_{u,v} N(u \in C_i,\ v \in C_j)}{\sum_u N(u \in C_i)}, \tag{8}$$

where $N(\cdot)$ represents the number of occurrences in the training corpus of the event in parentheses. Since the fuzzy C-means algorithm assigns a fragment to one or more classes, we defined a joint membership function for the two concatenated fragments uv in the following equation:

$$JM(u \in C_i,\ v \in C_j) = U_{C_i}(u) * U_{C_j}(v), \tag{9}$$

where $U_{C_i}(u)$ is the membership of u belonging to class C_i, assuming that the two events are independent. According to the joint membership, the FC bigram can be rewritten as

$$P(C_j|C_i) = \frac{\sum_{u,v} N(u,v) * JM(u \in C_i, v \in C_j)}{\sum_u N(u) * U_{C_i}(u)}. \tag{10}$$

The conditional probability $P(u \in C_i|v \in C_j)$ can be derived from the following equation:

$$P(v \in C_j|u \in C_i) = P(v|C_j) * P(C_j|C_i), \tag{11}$$

where

$$p(v|C_j) = \frac{N(v) * U_{c_j}(v)}{\sum_\phi N(\phi) * U_{c_j}(\phi)}. \tag{12}$$

The entropy of FC, H_p, using the FC bigram models can be denoted as follows:

$$
\begin{aligned}
H_p &= -\frac{1}{Q}\sum_{n=1}^{Q}\log P(W_n \in C_j | W_{n-1} \in C_i) \\
&= -\frac{1}{Q}\sum_{u,v}\sum_{i,j} N(u \in C_i,\ v \in C_j) \\
&\qquad * \log P(v \in C_j | u \in C_i) \\
&= -\frac{1}{Q}\sum_{u,v}\sum_{i,j} N(u \in C_i,\ v \in C_j) \\
&\qquad \times \log(P(v|C_j) * P(C_j|C_i)) \\
&= -\frac{1}{Q}\sum_{u,v}\sum_{i,j} N(u,v) * JM(u \in C_i,\ v \in C_j) \\
&\qquad \times \log(P(v|C_j) * P(C_j|C_i)).
\end{aligned} \tag{13}
$$

The perplexity of the FC bigram model can be calculated as

$$\text{Perplexity of FC} = 2^{H_p}.$$

3. FFCMM for a speech act

3.1. Speech act analysis

In a spoken dialog system, a human interacts with the computer agent using a speech act. Each speech act can be decomposed into one or more meaningful fragment combinations. For example, the fragments "謝謝 (thanks)" and "再見 (goodbye)" are two important meaningful fragments for the speech act – "Ending" in ATIS. The idea of the relationship between fragments and speech acts will assist in tagging the speech acts in the text sentences in the training corpus. However, not all fragments are important in determining a speech act. In our approach, some key-phrases are chosen from fragments and treated as the necessary components of a speech act. The idea (Saeki et al., 1996) of using a combination of important key-phrases to detect a speech act brings certain assumptions. It assumes that each speech act in a sentence is just the occurrence of key-phrases and the corresponding positions of these key-phrases do not affect the meaning of a sentence. These assumptions all go against the structure of a sentence and lead to some ambiguities and misunderstanding of the original meaning. For example, the sentence "The girl picks up a book at the desk". The key-phrases are "the girl", "a book", "at the desk". These key-phrases may cause an ambiguity and generate another sentence "The girl at the desk picks up a book". In addition, the selection of important key-phrases is also a thorny problem that directly affects the speech act identification rate. Although this method is not suitable for determining an exact speech act, it can be adopted as a pre-process to assist with the analysis and tagging of the corpus.

The determination of speech acts can be generalized into two steps. They are briefly described below:

1. *Key-phrase selection*: In our approach, we defined three criteria for empirically selecting key-phrases. Any fragment that conforms to one of the following criterion will be selected. The first criterion was defined because the semantic slots are the most important information for a speech act. It contains several kinds of semantic phrases or several FCs that are necessary to complete a speech act. The fragments used to fill the semantic slots were chosen first as the key-phrases. Second, the fragments in greeting and ending dialogs were also included. Finally, we choose key-phrases according to the frequency of the fragments in the training corpus. In total, we selected 67 fragments and manually clustered them into 12 kinds of key-phrases for speech act definition. The 12 kinds of key-phrases included "Date Type I", "Date Type II", "Place", "Time Type I", "Time Type II", "Airline", "About Fragment", "Ending", "Greeting", "Inquiry", "Affirmative" and "Negative".
2. *Speech act definition*: According to the above analysis, the idea (Saeki et al., 1996) of using a combination of important key-phrases to detect a speech act was adopted as a pre-process to assist in the analysis and tagging of the corpus. Each sentence in the training corpus can be decomposed into a combination of the key-phrases and then assigned to a specific speech act. The training sentences for a specific speech act were manually checked and selected in order to avoid duplicating speech acts and

reduce the number of tagging errors. Similar speech acts were merged and meaningless speech acts were discarded. In our approach, 30 main speech acts were chosen and defined semi-automatically using selected key-phrases based on the sentence analysis in the training corpus.

3.2. Definition of speech act FFCMM (Wu et al., 1998)

In general, grammar can be described as a sequence of POSs. The connection between two POSs produces several kinds of meanings. These meanings will evolve into a speech act through the interaction of POSs in the sequence. After the observation and analysis of speech acts, their characteristics can be modeled into a kind of Markov model that is effective at describing the transition relationship between two states.

In our approach, the above idea was adopted to model and identify a speech act. The speech act FFCMM is proposed to model the sequence of FCs. Each FC represents one state of speech act FFCMM. The transition mechanism in the FFCMMs is the same as that in a standard Markov model. That is, each state can transit to any other state. For each state, a state or FC membership is estimated and assigned to be 1 and is therefore discarded in a standard Markov model. In this approach, the frequency and the membership of the fragment in the FC are used to calculate the membership of the fuzzy fragment class in the FFCMM to show how much the syntactic information in the fragment belongs to this FC. The speech act FFCMM is defined as follows:

- N: the number of states in the model, each representing one FC. We labeled the individual states as $\{1, 2, \ldots, N\}$ and denoted the state at time ℓ as S_ℓ.
- M: the number of distinct observation symbols per state. The observation symbols correspond to the input fragments in the speech act being modeled. We denoted the individual symbols as
$$Y = \{y_1, y_2, \ldots, y_M\}. \quad (14)$$
- The state-transition probability from state i to state j is represented by
$$a_{ij} = P(S_{\ell+1} = j \,|\, S_\ell = i), \quad 1 \leqslant i, j \leqslant N. \quad (15)$$
The allowable transition paths are trained using the training corpus.
- The membership of the fuzzy fragment class corresponding to the observation symbol at state j is defined as
$$b_j(z) = \frac{N_{\mathrm{SA}_r}(y_z) \times u_{jz}}{\sum_{y \in C_j} N_{\mathrm{SA}_r}(y) \times u_{jz}}, \quad 1 \leqslant z \leqslant M, \quad (16)$$
where $N_{\mathrm{SA}_r}(y_z)$ is the frequency of y_z in a specific speech act SA_r and u_{jz} is the membership of y_z in the class C_j.
- The initial state is
$$\pi_i = P[S_\ell = i], \quad 1 \leqslant i \leqslant N. \quad (17)$$

3.3. Construction of speech act FFCMM

The construction of the speech act FFCMM can be divided into three parts. They are fragment extraction, fragment clustering and training of FFCMM. They are briefly explained below:

1. *Fragment extraction*: The main work in this step is to collect fragments for a specific task from the corpus. The fragment extraction method (Lai and Wu, 2000) was adopted to determine fragments based on the training corpus. We chose 206 fragments to form a task-specific dictionary.
2. *Fragment clustering*: This step clusters the fragments into FC. The fuzzy C-means algorithm was adopted to cluster the fragments based on the bi-directional bigram vector.
3. *FFCMM training*: The training corpus was tagged with 30 speech acts. Each FFCMM was trained using the subcorpus corresponding to a speech act. The training algorithm for the FFCMM was the maximum likelihood training algorithm.

3.4. Speech act identification

In this study, in order to evaluate the proposed methods, a basic recognition system integrating the acoustic and bigram language models were constructed to choose the most probable fragment sequence. For the speech recognizer in the

recognition component, given a speech utterance U, the score for each possible fragment sequence using the basic recognition system is described in the following equation:

$$\mathrm{BS}(\mathrm{FS}_k; U) = (1-\alpha)\log \mathrm{AP}(\mathrm{FS}_k; U) + \alpha \sum_{\ell} \log P(\mathrm{Fr}_{\ell}^{k} \,|\, \mathrm{Fr}_{\ell-1}^{k}), \tag{18}$$

where FS_k is the kth fragment sequence. $\mathrm{AP}(\mathrm{FS}_k; U)$ is the acoustic probability for FS_k. Fr_{ℓ}^{k} is the ℓth fragment in the sequence FS_k. $P(\mathrm{Fr}_{\ell}^{k} \,|\, \mathrm{Fr}_{\ell-1}^{k})$ is the fragment bigram probability. α is the weight between 0 and 1.

In the following, the FFCMM is proposed and combined to identify a speech act. For a speech utterance U, the corresponding possible fragment sequences are mapped to their corresponding FC sequence. The Viterbi algorithm was employed to find the most probable FC sequence, described as follows:

$$P_h(\mathrm{FC}_k|U) = \max_{1 \leqslant i \leqslant N} \delta_L^{k,h}(i), \tag{19}$$

$$\delta_{\ell}^{k,h}(i) = \max_{S_1 S_2 \ldots S_{\ell-1}} P[S_1 S_2 \ldots S_{\ell} = i, o_1 o_2 \ldots o_{\ell} | \lambda_h], \tag{20}$$

where $P_h(\mathrm{FC}_k \,|\, U)$ represents the probability corresponding to the kth FC sequence, FC_k, via the hth speech act FFCMM. N is the number of states. $\delta_{\ell}^{k,h}(i)$ is the highest probability along a single path, for the ℓth input phrase, which accounts for the first ℓ observations $O = [o_1 o_2 \ldots o_{\ell}]$ and ends at state i. For example, the fragments in the sentence "我要訂下午二點的飛機 (I want to book the flight departing at two o'clock this afternoon)" can be segmented into "我要訂 (I want to book)", "下午 (this afternoon)", "二點 (two o'clock)", "的 (de)" and "飛機 (flight)" denoted in Fig. 5(a). The corresponding FC are "Action", "Time", "Time", "Filler" and "Flight", denoted in Fig. 5(b), respectively. The state transition in the

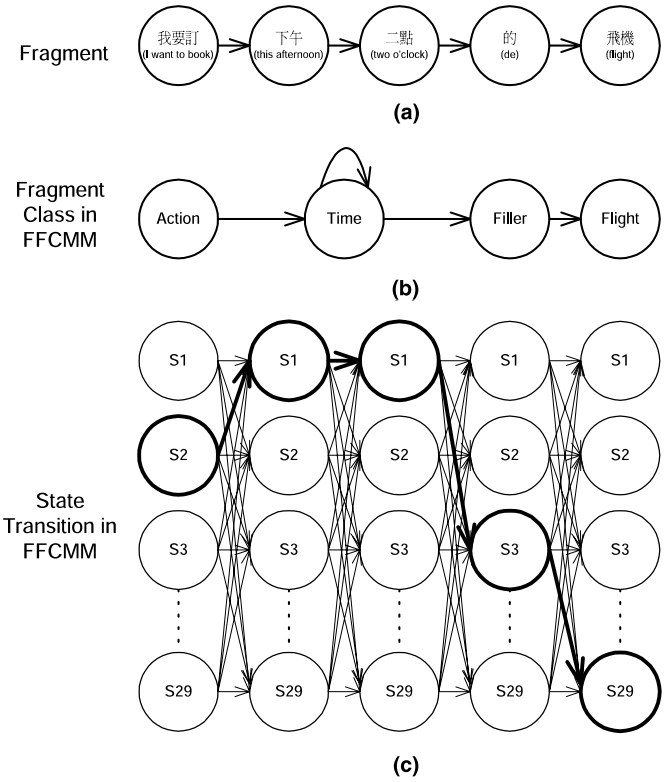

Fig. 5. Example of a speech act: (a) the fragments of the sentence, (b) the corresponding FCs, and (c) the state transition in the FFCMM.

SEMANTIC_SLOTS [
Action : 我要訂 (I want to book)
Date : (NULL)
Time : 下午 (this afternoon), 二點 (two o'clock)
Airline_Company : (NULL)
Location [
Depart_Place : (NULL)
Destination : (NULL)
]
]

Fig. 6. Semantic slots corresponding to the inquiry "我要訂下午兩點的飛機 (I want to book the flight departing at two o'clock this afternoon)".

FFCMM is shown in Fig. 5(c). In this case, the speech act FFCMM with the highest probability among all of the speech act FFCMMs is the "Booking" FFCMM. The semantic slots are then filled with respect to the inquiry, as shown in Fig. 6.

The score estimated from the FFCMM that considers the syntactic structure of the fragment sequence FS_k is then combined with the score from a basic recognition system that provides the acoustic score for the input speech to determine the most probable fragment sequences. This combination is described in the following equation:

$$\begin{aligned}&\mathrm{BS\_FHMM}_{h^*}(\mathrm{FS}_k;U)\\&\quad=(1-\beta)\max_{1\leqslant h\leqslant H}[\log P_h(\mathrm{FC}_k\,|\,U)]+\beta\mathrm{BS}(\mathrm{FS}_k;U),\end{aligned}\tag{21}$$

where β is the weight between 0 and 1. H is the number of speech act FFCMMs and h^* is the FFCMM with the highest score.

4. Speech act verification

Utterances in spoken language contain not only speech acts but also descriptive information (Saeki et al., 1996). In other words, the descriptive information is also an important part that represents the meaning of a sentence. In spoken language, the meaning of a sentence is generated not only by the words, but also by the interactions of the word sequences, especially the words filled in the semantic slots. In a word, there are some words that play a filler role in a sentence and can be ignored as garbage. But other words that can be filled in the semantic slots, are expressed meaningfully, especially the hidden semantic information expressed by the interactions between these words. These relationships must be carefully considered just as the real semantic meaning included in a sentence. For example, in the Chinese sentence "我要訂到台北的飛機 (I want to book the flight to Taipei)", the main semantic slots are "我要訂 (I want to book)" and "台北 (Taipei)". Trivially, these two main semantic slots imply the meaning of the original sentence after the inference. Therefore, the semantic slots and their interactions can properly model the descriptive information of a sentence.

In this section, the Bayesian probabilistic inference (Patterson, 1990), which provides a suitable inference method, was adopted as a post process to model the descriptive information of a speech act using the idea described above. The fragments are employed to verify the probable speech act from the top M candidate speech acts. Given some fragments as evidences E, the speech act H can be verified from the combination of fragments in a sentence. According to the Bayes' theory, the probability of hypothesis H given evidence E can be described as

$$P(H\,|\,E)=\frac{P(H)\times P(E\,|\,H)}{P(E)}.\tag{22}$$

For an arbitrary number of hypotheses H_i ($i=1,\ldots,K$), which are mutually exclusive and exhaustive, we suppose hypothesis H_i ($i=1,\ldots,K$) partition the universe. Eq. (22) can be generalized as

$$P(H_i\,|\,E)=\frac{P(H_i)\times P(E\,|\,H_i)}{\sum_{k=1}^{K}P(H_k)\times P(E\,|\,H_k)},\tag{23}$$

where K is the number of hypotheses.

To accommodate multiple evidence sources E_1, $E_2,\ldots,E_n$, Eq. (23) further generalized to obtain

$$P(H_i|E_1E_2,\ldots,E_n) = \frac{P(H_i)\times P(E_1E_2,\ldots,E_n|H_i)}{\sum_{k=1}^{K} P(E_1E_2,\ldots,E_n|H_k)\times P(H_k)}. \quad (24)$$

Eq. (24) describes the idea of modeling fragments and their interaction for a speech act. In other words, this is a suitable semantic score estimation for fragments and their hidden semantic information corresponding to a speech act.

The conditional probability $P(E_1E_2,\ldots,E_n|H_k)$ is very difficult to calculate from the training corpus because of the sparse data problem. Eq. (25) can approximate the conditional probability when assuming that each piece of evidence is statistically independent.

$$P(E_1E_2,\ldots,E_n|H_k) = P(E_1|H_k)\times P(E_2|H_k)\times\cdots\times P(E_n|H_k). \quad (25)$$

Therefore, given the fragment sequence E_1, $E_2,\ldots,E_n$, the probability of speech act H_i, $P(H_i|E_1E_2\ldots E_n)$, is rewritten as

$$P(H_i|E_1E_2,\ldots,E_n) = \frac{P(H_i)\times P(E_1|H_i)\times P(E_2|H_i)\times\cdots\times P(E_n|H_i)}{\sum_{k=1}^{K} P(E_1|H_k)\times P(E_2|H_k)\times\cdots\times P(E_n|H_k)\times P(H_k)}. \quad (26)$$

This probability is then integrated into our proposed system for speech act verification. In order to avoid the misrecognition and sparse data problems, we discard E_j if $P(E_j|H_k)\leqslant V$, where V is a chosen threshold. V is chosen as a small value ($=0.001$). That means we will discard fragment E_j if E_j rarely appears in speech act H_k in the training corpus.

In the verification process, using the Bayesian probabilistic inference, the fragments are used as the evidence E_j and the speech act is treated as a hypothesis H_k. The verification score for the kth fragment sequence is defined in Eq. (27),

$$\text{Verification\_Score}(\text{FS}_k|H_{h^*};U) = \log(P(H_{h^*}|\text{Fr}_1^k\text{Fr}_2^k\ldots\text{Fr}_\ell^k)), \quad (27)$$

where Fr_ℓ^k is the ℓth fragment in the sequence FS_k. The speech act with the highest $\text{BS\_FHMM}_{h^*}(\text{FS}_k;U)$ is regarded as the final output on the condition that $\text{Verification\_Score}(\text{FS}_k|H_{h^*};U)$ is above a chosen threshold T. Conversely, some speech act candidates with scores below the threshold are rejected in the verification process.

5. Experiments

In order to evaluate the proposed method, a spoken dialog system for an ATIS was investigated. There were 206 fragments extracted from the corpus and they were clustered into 38 FCs. The training text corpus with a memory size of 101 KB is described in Section 2.1. Twelve kinds of key-phrases were employed to divide the corpus into 30 sub-corpora, each corresponding to one speech act. The distribution of this 30 speech act sub-corpora is shown in Fig. 7. The FFCMMs were then constructed and trained using the corresponding sub-corpus to model the speech acts. The system was implemented on an IBM personal computer with a Dialogic/ESC telephone interface card. The experiments were carried out using a test database from 25 speakers (15 male and 10 female). There are 480 dialogs, which contain 3038 sentences.

5.1. Experiment on the class bigram perplexity

Perplexity, often called the average word branching factor of a language model, is a considerable parameter for evaluating word clustering performance. Eq. (13) gives the estimation of the class perplexity for the language model. The result of the perplexity for the language model using the fuzzy C-means algorithm is shown in Table 2. The traditional class bigram perplexity is provided in Table 1 for comparison. In our fuzzy C-means class bigram model, the perplexity becomes large compared to the traditional class bigram perplexity. This result is not strange because the uncertainty of the fuzzy C-means class bigram model becomes higher when the fragment belongs to many classes with a fuzzy membership between 0 and 1. The other characteristic in Table 2 is that the perplexity does not monotonically decrease when the class number becomes large. In our

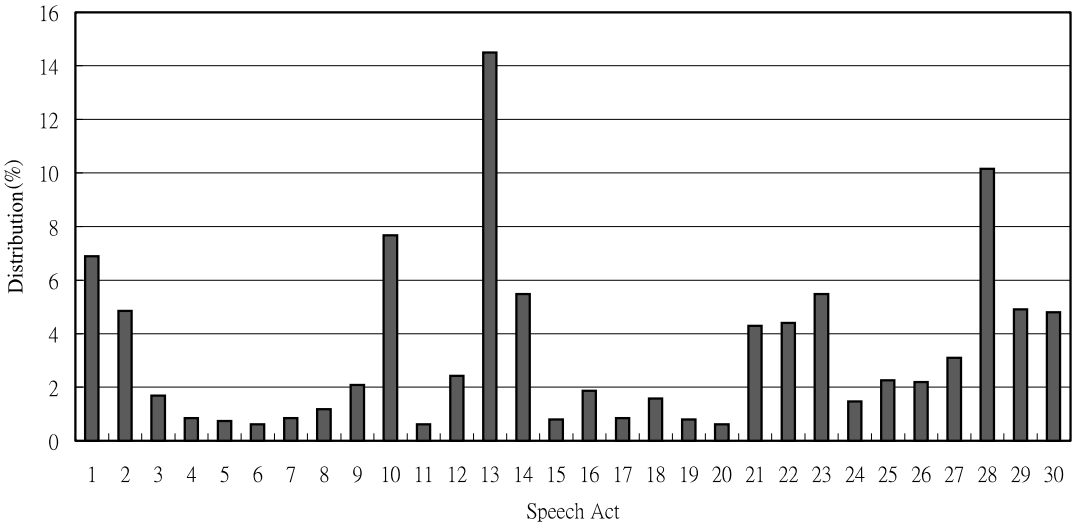

Fig. 7. Speech act distribution.

Table 2
Class bigram perplexity for different class numbers

Radius (R)	1.0	0.9	0.8	0.7	0.6
Class number	21	24	33	38	46
Fuzzy C-means class bigram perplexity	73.09	97.54	78.01	69.02	53.48
Traditional class bigram perplexity	17.32	15.44	13.64	12.64	11.27

opinion, one reason is because the unsupervized fuzzy C-means algorithm minimizes the overall average distortion. This criterion does not imply the monotonic decrease in perplexity when the class number becomes large. The other reason is because a fragment belongs to many classes with a fuzzy membership between 0 and 1. This is a soft decision not a hard decision (0 or 1).

5.2. Experiment on the system for comparison

In order to evaluate the performance of our proposed approach, a baseline system using the keyword set and typical speech act patterns was established. The keyword set and typical speech act patterns in the baseline system were selected using the combination of 12 kinds of key-phrases in Section 3.1 with respect to the corresponding speech act. Thirty speech acts were chosen to compare our approach. The baseline system processes the fragment sequence with the highest score produced by Eq. (18) to identify the speech act using the keyword set and typical speech act patterns. In order to evaluate the capability of this system a correct fragment rate (FCR) was defined as

$$\mathrm{FCR} = \frac{N_{\mathrm{C}} - N_{\mathrm{D}} - N_{\mathrm{I}}}{N_{\mathrm{TF}}}, \tag{28}$$

where N_{C} is the correct number of fragments, N_{D} is the number of fragments deleted, N_{I} is the number of fragments inserted, and N_{TF} is the total number of the fragments in the test database. The FCR was defined according to the standard accuracy measurements in the ASR. Eq. (28) does not take the substitution error into account because the substitution error can be seen as a kind of insertion error. Consequently, the FCR was defined according to the number of deletion and insertion errors. The other standard evaluation criterion was the speech act correct rate (SACR) which is defined as

$$\mathrm{SACR} = \frac{N_{\mathrm{C}}(\mathrm{SA})}{N_{\mathrm{TS}}}, \tag{29}$$

where $N_{\mathrm{C}}(\mathrm{SA})$ is the correct number of speech acts and N_{TS} is the total number of test sentences. To evaluate the performance of the baseline system and to find an appropriate value for α, the 3038 sentences in the speech form were fed into the telephone speech recognizer to output the fragment sequences. The performance of the baseline system for different values of weighting coefficients α in Eq. (18) is shown in Fig. 8. The SACR achieved 80.4% when $\alpha = 0.8$ and the FCR was 76.5%. This is the best performance for the baseline system. The value of α and the basic recognition system were then used for further experiments.

5.3. Experimental performance for different R values

Before evaluating the performance of the FFCMM, the radius R that determines the class number must be chosen first. In our approach, R is chosen according to the SACR performance. The experiment was conducted using 3038 test sentences when $\alpha = 0.8$. The experimental results are shown in Table 3. SACR is listed as a function of the value of β for different values of R. The class number was chosen as 38 ($R = 0.7$) and this results in the best speech act identification rate when β was chosen as 0.2.

5.4. FFCMM experimental performance

For evaluating the system SACR and FCR, the output of the basic recognition system with the value $\alpha = 0.8$ was used directly as the input for our proposed system. The 3038 sentences in speech form were also fed into the basic recognition system to output the fragment sequences. The value of β must be determined in order to achieve the optimal combination of FFCMM and the basic recognition system. The experimental results shown in Fig. 9 give the SACR and FCR for different values of β. If $\beta = 0$, the system is dominated by the baseline system. When β was chosen

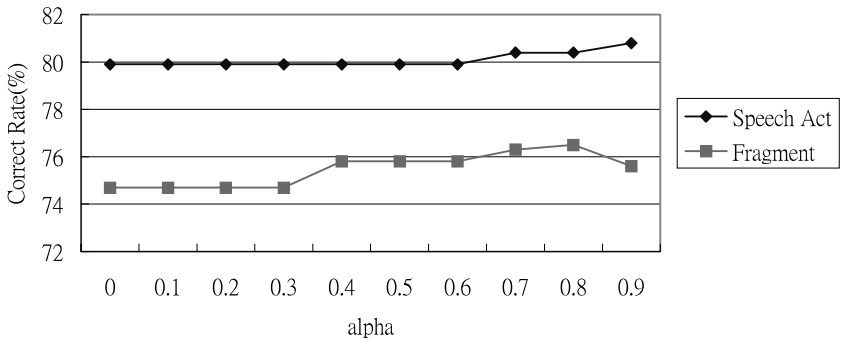

Fig. 8. Performance of the baseline system as a function of the values of α.

Table 3
SACR as a function of the values of β for different R

		Beta								
Radius R	Class number	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
1.0	21	85.7	85.7	85.7	84.2	83.8	81.7	75.9	63.4	52.2
0.9	24	86.5	86.2	86.2	84.7	84.2	82.9	80.4	71.4	54.0
0.8	33	87.1	87.5	86.8	85.1	84.8	82.9	81.3	74.6	58.0
0.7	38	87.1	**88.4**	86.8	85.6	85.1	83.5	81.6	80.2	65.4
0.6	46	87.5	87.9	86.2	85.6	85.1	83.1	80.6	77.2	61.2

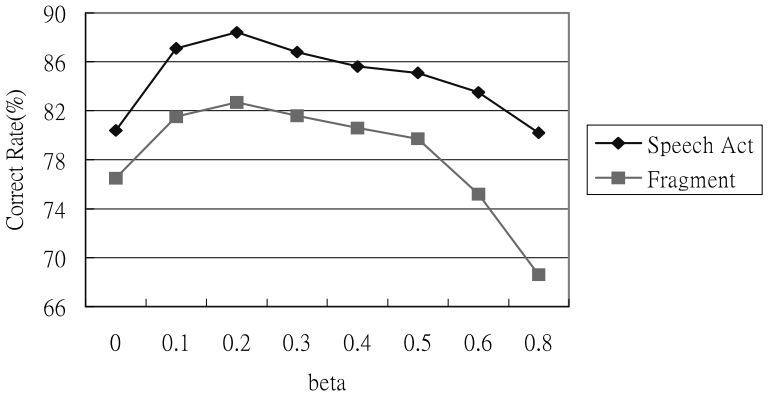

Fig. 9. SACR and FCR as a function of the values of β.

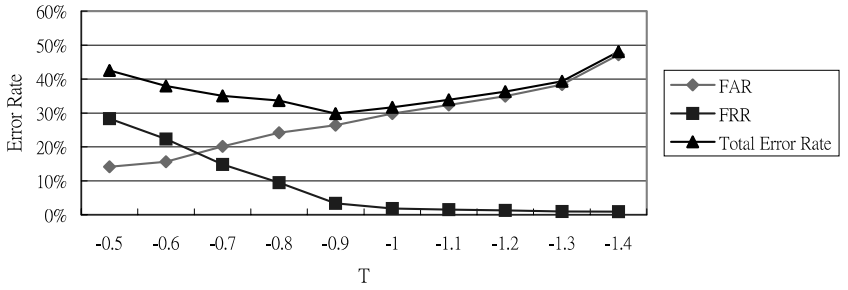

Fig. 10. Total error rate as a function of the threshold T.

as 0.2, the system achieved the best result in which SACR was 88.4% and FCR was 82.7%. The SACR for the system combining the FFCMM can be improved by 8% compared to the baseline system. This improvement was achieved because the proposed FFCMM takes the syntactic characteristics of a speech act into account. It not only models the grammar of a sentence, but also models the transition between two FCs. Intuitively, the information modeled by FFCMM is efficient for adjusting the bigram model because the bigram model provides only a prediction probability for the succeeding fragment according to the preceding fragment. The combination of the bigram model with the FFCMM is helpful to moderate the effect caused by speech recognition errors.

5.5. Experiment on the speech act verification model

In the speech act verification model, the verified score Verification_Score$(\mathrm{FS}_k \,|\, H_{h^*}; U)$ was used to make acceptance/rejection decisions. The experimental results shown in Fig. 10 give the false acceptance rate (FAR) and false rejection rate (FRR) for different values of the threshold T. When the threshold T was chosen as -0.9, the sentence rejection rate was about 6% and the total error rate reached the lowest value of 30% containing a 3.4% FRR and a 26.6% FAR. Taking rejection into consideration, the modified FCR is redefined as follows:

$$\mathrm{MFCR} = \frac{NA_{\mathrm{C}} - NA_{\mathrm{D}} - NA_{\mathrm{I}}}{NA_{\mathrm{TPS}}}, \tag{30}$$

where NA_{C} is the correct number of fragments, NA_{D} is the number of deleted fragments, NA_{I} is the number of inserted fragments, and NA_{TPS} is the total number of fragments in the accepted sentences. The modified SACR is

$$\mathrm{MSACR} = \frac{N_{\mathrm{C}}(\mathrm{SA}) - N_{\mathrm{FR}}}{N_{\mathrm{TS}} - N_{\mathrm{R}}}, \tag{31}$$

where N_{TS} is the total number of test sentences, N_R is the number of rejected sentences, $N_C(SA)$ is the correct number of SA, and N_{FR} is the number of falsely rejected sentences.

The main purpose of verification is to reject the errors accepted by the FFCMM. These sentences are legal in the statistical grammar of a speech act. However, this does not imply that the sentences are meaningful in the specific speech act and should be rejected. The system achieved the best results with a 91.3% MSACR and an 85.1% MFCR after the verification process. The system performance using speech act verification was further improved compared to the combination of FFCMM and the basic recognition system. This improvement was because the descriptive information was used as an important part in representing the meaning of a speech act.

5.6. Experiment using SACR as a function of fragment sequence length

This experiment was conducted to evaluate the SACR for different fragment sequence lengths to observe the effect of FC transitions modeled by the FFCMM. In this experiment, the baseline system, the basic recognition system with FFCMM and the basic recognition system with FFCMM and speech act verification were evaluated separately and compared. The test database with 480 dialogs, which contain 3038 sentences, were fed into the dialog system in speech form. Each sentence was tagged with its corresponding correct speech act and different fragment sequence lengths. The experimental results are listed in Table 4.

From the comparison of the baseline system and the basic recognition system with FFCMM, the longer the fragment sequence length, the better the performance improvement. This result shows that the bigram model considers only the preceding word and is not suitable to express a long-term relation. The speech act identification method using the keyword set and typical speech act patterns in the baseline system was poor in identifying a speech act. Conversely, as the FFCMM is taken into consideration, the SACR can be significantly improved, especially for longer fragment sequence lengths. This is because the FFCMM models not only the FC of a fragment, but also the statistical syntactic characteristics of a speech act. This information is helpful in identifying a speech act and adjusting the sentence candidates from the basic recognition system to improve the SACR. However, the verification process does not produce an obvious improvement for longer sentences. This is because the verification is only performed based on some frequently used fragments due to the sparse data problem.

6. Conclusion

In this paper, a corpus-based FFCMM was proposed to identify the speech act in a spoken sentence. The FFCMM models not only the FCs of a fragment, but also the statistical syntactic characteristics of a speech act to improve the reliability of an identified speech act. The speech act verification utilizes the relationship between fragments and speech acts to infer the semantic meaning and to verify the identified speech acts. The fragment extraction method and FFCMM are statistically based. The experimental results show that the speech act identification rate can be improved by 10.5% using the FFCMM and the speech act verification compared to the baseline

Table 4
SACR as a function of the fragment sequence length for three systems

Fragment sequence length	2	3	4	5	6	7	8	9
Basic recognition system + keyword (baseline system)	80.6	81.6	83.5	80.9	79.8	77.2	75.4	74.1
Basic recognition system + FFCMM	88.3	87.8	88.6	87.9	88.7	88.5	90.3	87.5
Basic recognition system + FFCMM + verification ($T = -0.9$)	90.3	92.5	91.3	90.6	89.7	92.4	93.8	88.4

system. The system with FFCMM and speech act verification has a good capability to compensate for the typical errors that result from speech recognition systems and acquire accurate information from a dialog, especially for longer sentences.

Acknowledgements

The authors would like to thank the National Science Council, Republic of China, for its' financial support to this work, under contract No. NSC87-2622-E006-008.

References

Allen, J., 1994. Natural Language Understanding. Benjamin/Cummings, Menlo Park, CA, pp. 542, 554–557.

Arai, K., Wright, J.H., Riccardi, G., Gorin, A.L., 1999. Grammar Fragment acquisition using syntactic and semantic clustering. Speech Communication 27 (1), 43–62.

Bennacef, S., Lamel, L., 1996. Dialog in the RAILTEL telephone-based system. In: Proceedings of ICSLP'96, Vol. 1, pp. 550–553.

Chiang, T.H., Peng, C.M., Lin, Y.C., Wang, H.M., Chien, S.C., 1998. The design of a mandarin Chinese spoken dialog system. In: Proc. COTEC'98, Taipei, pp. E2-5.1–E2-5.7.

Harksoo Kim, Jeong-Mi Cho, Jungyun Seo, 1999. Fuzzy trigram model for speech act analysis of utterances in dialogs. In: Fuzzy Systems Conference Proceedings, 1999, FUZZ-IEEE '99, IEEE Internat., Vol. 2, 1999, pp. 598–602.

Jelinek, F., Mercer, R., Roukos, S., 1990, Classifying words for improved statistical language models. in: Proc. CASSP90, Vol. 1, pp. 621–624.

Lai, Y.S., Wu, C.H., 2000. Unknown word and phrase extraction using a phrase-like-unit-based likelihood ratio. Internat. J. Oriental Languages 13 (1), 83–95.

Lee, C.J., Huang, E.F. Chen, J.K., 1997. A multi-keyword spotter for the application of the TL phone directory assistant service. In: Proc. 1997 Workshop on Distributed System Technologies & Applications, pp. 197–202.

Martin, S., Liermann, J., Ney, H., 1998. Algorithms for bigram and trigram word clustering. Speech Communication 24 (1), 19–37.

Meng, H., Busayapongchai, S., Zue, V., 1996. WHEELS: A conversational system in the automobile classification domain. In: Proc. ICSLP'96, Vol. 1, pp. 542–545.

Patterson, D.W., 1990. In: Introduction to Artificial Intelligence & Expert System. Prentice-Hall, Englewood Cliffs, NJ, pp. 107–125.

Pieraccini, R., Tzoukermann, E., Gorelov, Z., Gauvain, J.-L., Levin, E., Lee, C.-H., Wilpon, J.G., 1992. A speech understanding system based on statistical representation of semantics. In: Proc. ICASSP92, Vol. 1, pp. 193–196.

Riccardi, G., Gorin, A.L., 2000. Stochastic language adaptation over time and state in natural spoken dialog systems. Speech Audio Process., IEEE Trans. Vol. 81, pp. 3–10.

Saeki, M., Matsumura, K., Shimoda, J., Kaiya, H., 1996. Structuring utterance records of requirements elicitation meetings based on speech act theory. Requirements engineering. In: Proc. ICRE 1996, pp. 21–30.

Seide, F., Kellner, A., 1997. Toward an automated directory information system. In: Proc. EuroSpeech'97, Vol. 3, pp. 1327–1330.

Tran, D., Le, T.V., Wagner, M., 1998a. Fuzzy Gaussian mixture models for speaker recognition. In: Proc. ICSLP98, Sydney, Australia.

Tran, D., Wagner, M., Le, T.V., 1998b. A proposed decision rule for speaker recognition based on fuzzy C-means clustering. In: Proc. ICSLP98, Sydney, Australia.

Wang, H.C., Wang, J.F., Liu, Y.N., 1997. A conversational agent for food ordering dialog based on venus dictate. In: Proc. ROCLING X Internat. Conf., pp. 325–334.

Wright, J.H., Gorin, A.L., Riccardi, G., 1997. Automatic acquisition of salient grammar fragments for call-type classification. In: Proc. Eurospeech97, Greece, September 1997, pp. 1419–1422.

Wu, C.H., Chen, J.H., 1999. Template-driven generation of prosodic information for Chinese concatenative synthesis. In: Proc. ICASSP'99, Phoenix, USA.

Wu, C.H., Yan, G.L., Lin, C.L., 1998. Spoken dialogue system using corpus-based hidden Markov model. In: Proc. ICSLP98, Sydney, Australia.

Zimmermann, H.J., 1991. Fuzzy Set Theory and its Applications. Kluwer Academic Publishers, Dordrecht, pp. 230–236.