UNSUPERVISED STATE CLUSTERING FOR STOCHASTIC DIALOG MANAGEMENT

Fabrice Lefèvre and Renato de Mori

LIA - University of Avignon - BP 1228 84911 AVIGNON FRANCE
{fabrice.lefevre, renato.demori}@univ-avignon.fr

ABSTRACT

Following recent studies in stochastic dialog management, this paper introduces an unsupervised approach aiming at reducing the cost and complexity for the setup of a probabilistic POMDP-based dialog manager. The proposed method is based on a first decoding step deriving semantic basic constituents from user utterances. These isolated units and some relevant context features (as previous system actions, previous user utterances...) are combined to form vectors representing the on-going dialog states. After a clustering step, each partition of this space is intented to represent a particular dialog state. Then any new utterance can be classified according to these automatic states and the belief state can be updated before the POMDP-based dialog manager can take a decision on the best next action to perform.

The proposed approach is applied to the French MEDIA task (tourist information and hotel booking). The MEDIA 10k-utterance training corpus is semantically rich (over 80 basic concepts) and is segmentally annotated in terms of basic concepts. Before user trials can be carried out, some insights on the method effectiveness are obtained by analysis of the convergence of the POMDP models.

1. INTRODUCTION

Spoken Language Understanding (SLU) is the interpretation of signs conveyed by a speech signal. This is a difficult task because meaning is mixed with other information like speaker identity and environment. If SLU is used in a spoken dialog system, interpretation is performed on successive dialog turns in which components of semantic structures and dialog acts are detected together with their relations.

For this purpose, a modular approach to SLU is proposed. It consists in hypothesizing, in a first stage, basic semantic constituents which can be roles and values, verb cases and relations. In successive stages, constituents and relations are composed into semantic structures which can be represented by frames.

A frame is a computational model for representing semantic entities and their properties. It is based on a data structure containing slots and fillers. Computer interpretation uses computer semantic to represent objects of a domain by frame instances hypothesized from speech signal. Frame instantiation may be triggered by evidence or by expectation. Evidence can be partial and may be confirmed by tests. Once a frame instance is created, then slots are filled in different dialog turns. SLU and dialog manager (DM) progressively build frame instances that represent beliefs about user intentions. The goal is to satisfy a user request when it is enough specified to allow a consequent system action to be performed.

In order to achieve this type of goals in typical telephone service applications, the machine can perform three types of actions: send a message to the user (communicative action), prune the set of dialog beliefs (control action), perform an internal action like the access to a database (system action). The choice of the action to perform at a given dialog turn is part of the dialog strategy which can be conceived using different approaches.

Recently, it has been proposed to consider detailed instantiations and actions as belonging to a master space and to introduce a summary space in which summary actions are performed [9]. This makes it possible to conceive dialog strategies in the framework of a partially observable Markov decision process (POMDP) which is based on policies obtained by an optimality criterion.

With this strategy, a dialog is supposed to be in different possible states at a given dialog turn with a given probability for each state. The action to be performed in that turn is decided based on a criterion for obtaining the maximum reward on the entire dialog. The approach is applicable only if the number of possible states is small. In reality, a dialog state should be considered for any possible binding of the logic variables in the SLU and dialog knowledge.

In order to apply the POMDP approach, detailed dialog states in the master space should be clustered into general dialog states in summary space. It is advantageous to perform dialog state clustering with an automatic, possibly unsupervised procedure. Some techniques like Principal Component Analysis (PCA) have been proposed for this purpose [1]. States in summary space, obtained with a clustering technique, should be consistent with the semantic of the meaning representation formalism, which, in our case, is a frame language.

This work was partially supported by the 6th Framework Research Programme of the European Community, Project LUNA IST contract 33549 and PASCAL network of excellence.

A reasonable assumption, which is considered in this paper, is that progress of semantic composition is described by states in the state master space, while actions related to slot filling priorities and modalities, such as corrections and requests for confirmation, are summary actions resulting from decisions based on POMDP. The conjecture analyzed and evaluated in this paper is that summary actions can be determined only on the basis of instances of semantic constituents, without considering the details of semantic composition. States in summary space are obtained automatically using semantic constituents and other semantic features. Summary state transition and observation probabilities as well as the other POMDP parameters are obtained directly from an annotated training corpus. In this work, the rich dialog corpus MEDIA is used. MEDIA contains complex dialogs involving tens of semantic constituents, complex semantic structures to be composed as a dialog progresses and a fairly large number of turns per dialog.

The paper is organized as follows. The next section presents the MEDIA corpus. Section 3 describes the basic semantic constituent decoding followed by a description of the proposed unsupervised dialog state clustering in Section 5. A short review of POMDP-based dialog manager is given in Section 4. The last section reports some experiments on the MEDIA corpus.

2. THE MEDIA CORPUS

The MEDIA corpus [2] has been recorded using a wizard of Oz (WOZ) system simulating a telephone server for tourist information and hotel booking. Eight scenario categories were defined with different levels of complexity. An example of dialog is given in Figure 2. The corpus accounts 1257 dialogs from 250 speakers and contains about 70 hours of dialogs. The MEDIA training corpus is conceptually rich with more than 80 basic concepts manually transcribed and annotated.

An example of a semantic representation of a user utterance is given in Figure 3. A semantic segment is represented by a triplet which contains: the mode (+, -, ? and ˜ meaning `optional` as in *with a shower if possible*), the role name and the normalized value. In order to disambiguate sentences such as *not in Paris... in Nancy*, modes are assigned in a per segment basis. In the reference annotations, only the user messages are annotated.

The semantic dictionary includes 83 basic roles and 19 specifiers, representing the main objects of the task (`hotel`, `room...`). Relations and compositions are not annotated. However, combinations of roles and specifiers make it possible to express relationships between semantic segments. In Figure 3, the attribute `payment-amount-integer-room` is derived from the combination of the `payment-amount-integer` attribute with the specifier `room`. This annotation is comparable to a labelling produced by a semantic shallow parsing. The hierarchical representation of a query can be retrieved from its flat annotation [2]. The combinations of basic roles and specifiers result in 1121 potential attributes. A total of 144 distinct attributes appears in the training corpus with about 2.2k different normalized values.

3. SLU DECODING FOR OBTAINING SEMANTIC CONSTITUENTS

Two types of basic constituents are used in this work: dialog acts (DA) corresponding to POMDP actions and roles.

Dialog acts

A DA annotation has been performed on the WOZ turns of the corpus. This annotation is rule-based. The rules have been manually corrected and adapted. Being human based, the WOZ interactions may not always be easily categorized as a unique dialog act. For instance, most database queries are combined with a final confirmation question (see example in Figure 2). When such DA combination has many occurrences, compound DA have been retained (see Figure 4) and used in the annotation. A total of 15 DAs are considered: 10 basic acts and 5 compound acts defined in Figure 4.

Roles

Semantic roles are tagged by conceptual decoding. The result is a sequence of concepts representing the meaning of a utterance with the assumption that there is a correspondence between tags and word chunks [3].

Stochastic SLU modeling has been recently used with interesting results and parsimonious development cost [4, 5, 6]. In [7], the development of a 2+1-level SLU system based on dynamic Bayesian networks (DBN) has been presented. The system described in this paper has a first stage of DBN-based decoding producing a sequence of role/value hypotheses. Two further steps are then executed to perform classification based on Conditional Random Fields to hypothesize modes and references described by specifiers. The concept error rate (CER) is about 26% on the MEDIA test set with an automatic extraction of concept sequences from annotated sentences.

4. POMDP DIALOG MANAGER REVIEW

Foundations for using POMDP is applications such as dialogue management can be found in [8, 9, 10]. At each dialog turn, the dialog manager has a belief represented by a probability distribution over its states. The action to be performed depends on the belief rather than on a specific state. Probabilities of the system being in each state at the next turn depends on the previous belief and the action performed.

For the i-th state at the k-th dialog turn, the probability $p_k(s_i)$ of being in state s_i at turn k is:

$$p_k(s_i) = f(a_{k-1}, b_{k-1}) \quad (1)$$

The probability $p_k(s_i)$ is a function of the belief b_{k-1} and the action a_{k-1} at turn $k-1$.

Associated to each state are estimations of the user's input act [11], the intended user goal and a representation of the dialog history. A dialog policy is used to specify a system speech act. The dialog ends when the user's goal is satisfied or the dialog fails.

The most suitable system action in a dialog turn is selected following an optimal dialog strategy. Reinforcement learning [12] offers methods for selecting optimal actions in an environment described by a Markov Decision Process (MDP). A discussion on MDP in DM can be found in [8].

The dialog policy π chooses a system action, based on the actual dialog state:

$$a_k = \pi(s_k) \tag{2}$$

State transitions are characterized by a probability distribution:

$$p(s_k|s_{k-1}, a_{k-1}) \tag{3}$$

The combination of action and state at turn k determines the immediate reward:

$$r_k = R(s_{k-1}, a_{k-1}, s_k) \tag{4}$$

where R is a (sparse) tabulated function designed manually by an expert. The goal is to find a policy which maximizes the total future reward:

$$R_k = \sum_{\tau=k+1}^{K} r_\tau \tag{5}$$

All solution methods of reinforcement learning use a value function for any state:

$$V_\pi(s) = E_\pi(r_k|s_k = s) \tag{6}$$

An optimal policy is obtained with a practically more useful function:

$$Q^\pi(s, a) = E_\pi(r_k|s_k = s, a_{k-1} = a) \tag{7}$$

The optimal policy at state s can be determined as follows:

$$\pi'(s) = \arg\max_a Q^\pi(s, a) \tag{8}$$

Methods for finding an optimal policy are proposed in [12]. Unfortunately, MDP is not applicable in practice because complete knowledge of the system state is unrealistic, and the user's belief cannot be observed.

Thus a dialog system must base its strategy on incomplete data. A suitable strategy can be obtained with POMDPs. A POMDP is equivalent to an MDP with a continuous state space. With POMDP, dialog states are summarized to produce system beliefs. Algorithms for POMDP can be found in [13, 14]. Comprehensive introductions to the use of POMDP in DM can be found in [9, 10, 15].

As dialog states cannot be directly observed, relations between observations and states are summarized in the observation probability distribution:

$$p(o_k|a_{k-1}, s_k) \tag{9}$$

The observation probabilities, combined with state transition probabilities, are used for computing the bayesian update of the belief state. The algorithm iteratively computes the best value for V, w.r.t. the selected actions providing an optimal policy solution. Unfortunately these algorithms have a high computational complexity. Sub-optimal variants of these complex algorithms have been proposed. Even these variants can hardly handle more than few tens of observations, states and actions. This requires the summarization of dialogue states in a small number of descriptors as proposed in [16, 17].

5. CLUSTERING OF DIALOG STATES

If the main purpose of a dialogue is to interpret the messages of a user and infer her/his intentions, it is possible to define a dialogue state for the k-th dialog turn based on system actions and semantic hypotheses generated up to that turn. In practice, if an annotated corpus is available, a state is characterized using features derivable from the annotations. In the MEDIA corpus, annotations are available for basic semantic constituents, modes and specifiers. In order to take into account system actions, the annotation has been integrated with a human labeling of system prompts with dialog acts listed in Figure 2. The dialog state at turn k is thus characterized by the set of annotated features asserted up to the the turn including possible specifiers and modalities. Composition of constituents into semantic structures and possible inferences are not taken into account at the moment. Using this type of information would imply the availability of corpora annotated with predicates and arguments or roles and values. These annotations should be performed and validated by human experts or obtained by an automatic system using formal knowledge compiled by humans. In both cases, a costly effort is required and was not made for the MEDIA corpus.

Using a set of features consisting of semantic constituents and some other contextual information (previous user basic concepts, previous user action, previous system action), clusters are derived with an unsupervised procedure to represent states in summary space.

A light supervision is introduced only in the way the features are selected and combined. These operations are based on the following considerations:

- Combination is based only on semantic constituents and former dialog acts.
- Constituent information can be over-specified and redundant for the purpose of designing DM strategies. Thus, constituents used by similar semantic functions are merged into a subset. For example, `loc-town` and `loc-near` are considered as components of a `localization` function. Most values are discarded. Noticeable exceptions are values of dialogic constituents which may be useful to distinguish among different interpretations. Examples are `response`, `command-task`...

- Context-sensitive information is useful. For example, in case of a negative modality, an already activated concept is canceled.

Due to the nature of the feature vectors, algorithms for unsupervised clustering, suitable for sparse and (mostly) binary vectors have been considered (e.g. [18]). Different types of classifiers for mapping feature vectors into clusters were considered and compared. The unsupervised clustering algorithm and the classifiers that were used will be briefly described with experimental results in the next section.

After computing features for the training data, two sets of probabilities, defined in Section 4, have to be estimated to define the POMDP parameters:

$$p(s_k|s_{k-1}, a_{k-1}) \quad (10)$$

$$p(o_k|a_{k-1}, s_k) \simeq p(o_k|s_k) \quad (11)$$

Using sequences of clusters obtained with the training data, state transition probabilities in summary space are estimated from the training data.

Observation probabilities are estimated by comparing clustering and classification results. As data sparseness is often an issue for POMDP training, estimates for the observation probability distributions requires the use of suitable models for state sequences in summary space. Each state in summary space is identified by a cluster index and these indices are considered as words of language models. Factored language models (FLM) have been chosen for sequences of cluster indices along with generalized parallel backoff (GPB) [19]. FLMs are an extension of standard LMs where the prediction is based upon a set of features (and not only on previous occurrences of the predicted variable). GPB extends the standard backoff procedures to the case where heterogeneous feature types are considered and no obvious temporal order exists. The FLM parameters are estimated using training data for both transition and observation probabilities. With these probabilities the others POMDP parameters are obtained.

6. EXPERIMENTS

To evaluate the proposed method, a POMDP-based DM for the MEDIA task was created. A set of 14k utterances were used for training. A set of 3k new utterances was used for testing. The clustering process is performed simultaneously on both sets so as to obtain references for the test test. A classical EM clustering algorithm is used [1].

Three conditions were considered for classifying the test data. In the first condition, the reference semantic annotation is used to build feature vectors. In the second condition, vectors are built using the concept sequences decoded from manual transcription of the users utterances. In the third condition, concept sequences are decoded using the 1-best hypothesis generated by an Automatic Speech Recognition (ASR) system. Two classifiers for assigning utterances to clusters were compared: an Instance-Based classifier [18] (a version of the k-nearest neighbors, well suited to binary sparse vectors) and a Support Vector Machine (SVM) classifier [20]. Two versions of the Instance-Based classifiers with respectively one and twenty neighbor vectors were considered.

[1] The WEKA toolkit, http://www.cs.waikato.ac.nz/ml/weka/, is used in the experiments reported here.

Classifier	Train (#14282)	Test (#3524)		
		REF	SLU	ASR+SLU
WER	0.0	0.0	0.0	33.5
CER	0.0	0.0	21.3	43.4
1-NN (IB1)	0.0	8.4	26.3	30.1
20-NN (IB20)	16.0	16.2	30.0	33.4
SVM	3.9	3.9	23.5	29.2

Table 1. Classification error rates on the training and test sets with 1-NN, 20-NN and SVM considering 3 conditions for concept decoding (reference (*REF*), from manual transcription (*SLU*) or from ASR (*ASR+SLU*)).

Table 1 shows word error rates (WER) and concept error rates (CER) obtained with the classifiers on interpretations from transcriptions (indicated as SLU) and from ASR results (indicated as ASR+SLU). The word error rate on the test set is 33.5% [21], the concept error rates are 21.3% on perfect transcriptions and 43.4% using ASR results (here no specifiers and only 2 modes are considered). The results shown in Table 1 are unsurprinsingly good since test vectors are obtained in the same conditions as training vectors (column "Test/REF" in Table 1). However an important loss is observed when the classification is performed after concept decoding (between 15 and 20%) and an additional loss appears when decoding is based on ASR results (between 3 and 6%).

The POMDP parameters were estimated from the whole MEDIA data set. Figure 1 shows the evolution of average rewards for simulated sequences of states as function of POMDP training iterations represented by training time (obtained on a standard PC). Two parameter sets were tested for the POMDP definition, with different observation probability distribution. In the first case, observations were built from concept sequences decoded from manual annotations (indicated as SLU) exhibiting a 23.5% classification error, the second from decoding on ASR hypotheses (indicated as ASR+SLU) exhibiting a 29.2% classification error.

Averages are performed on the immediate rewards accumulated with 10k simulated dialogs using the POMDP probability distributions as user model. Figure 1 shows the convergence of the value iteration algorithm for the two types of data. Various values were tested for the number of belief points. The choice of 500 points appears to be a reasonable trade-off between performance and training time. The dialog states being unsupervised, assigning rewards to particular states is not easy. So, two additional dialog states are added: start and end. The simulated dialogs start on the start state and are interrupted when the end state is reached or after a maxi-

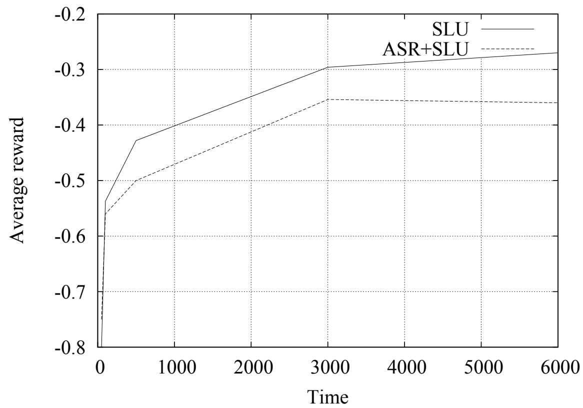

Fig. 1. Convergence of the average reward of the POMDP wrt to training time (sec).

mum number of turns (200). The two curves in Figure 1 show that, referring to the POMDP value iteration convergence, the proposed technique is only slightly sensitive to ASR and SLU decoding errors.

7. CONCLUSION

In this paper, an unsupervised automatic clustering of dialog states has been introduced and investigated. The proposed approach has allowed the development of a POMDP-based dialog manager on a challenging information retrieval task, embodied by the MEDIA corpus.

Preliminary experiments give some insights on the method feasibility and suggest future research work. The use of lattices of words and concepts (or n-best lists) fits perfectly in the proposed framework and is likely to improve the robustness of the system. Also, user trials will be needed for further evaluations of this approach.

8. REFERENCES

[1] N. Roy and G. Gordon. "Exponential family PCA for belief compression in POMDPs", NIPS, 15, 2003.

[2] H. Bonneau-Maynard et al, "Results of the French EVALDA-MEDIA evaluation campaign for literal understanding", LREC, 2006.

[3] E. Levin and R. Pieraccini, "Concept-based Spontaneous Speech Understanding System", ESCA Eurospeech, 1995.

[4] H. Bonneau-Maynard and F. Lefèvre, "A 2+1-level stochastic understanding model", IEEE ASRU, 2005.

[5] Y. He and S. Young, "Spoken Language Understanding using the Hidden Vector State Model", In Speech Communication, 48(3-4):262-275, 2005.

[6] G. Damnati, F. Bechet and R. De Mori, "Spoken Language Understanding Strategies On The France Telecom 3000 Voice Agency Corpus", IEEE ICASSP, 2007.

[7] F. Lefèvre, "Dynamic Bayesian Networks and Discriminative Classifiers for Multi-stage Semantic Interpretation", IEEE ICASSP, 2007.

[8] E. Levin, R. Pieraccini and W. Eckert, "Using MDP For Learning dialog Strategies", IEEE ICASSP, 1998.

[9] N. Roy, J. Pineau and S. Thrun, "Spoken Dialog Management Using Probabilistic Reasoning", ACL, 2000.

[10] S. Young et al, "The Hidden Information State Approach to Dialog Management", IEEE ICASSP, 2007.

[11] J. Allen and C. Perrault, "Analyzing Intention in Utterances", In Artificial Intelligence, 15:143178, 1980.

[12] R. Sutton and A. Barto, "Reinforcement Learning: An Introduction", In Adaptive Computation and Machine Learning", MIT Press, 1998.

[13] R. Brafman, "A Heuristic Variable Grid Solution Method for POMDPs", AAAI, 1997.

[14] B. Bonet, "An e-Optimal Grid-based Algorithm for POMDP", ICML, 2002.

[15] J. Williams and S. Young, "Partially Observable Markov Decision Processes for Spoken Dialog Systems", In Computer, Speech and Language, 21:393-422, 2007.

[16] C. Boutilier and D. Poole, "Computing Optimal Policies for Partially Observable Decision Processes using Compact Representations", AAAI, 1996.

[17] J. Williams, P. Poupart and S. Young, "Factored POMDP for Dialog Management", Workshop on Knowledge and Reasoning in Practical Dialog Systems, IJCAI, 2005.

[18] D. Aha and D. Kibler, "Instance-based Learning Algorithms", In Machine Learning, 6:37-66, 1991.

[19] J. Bilmes and K. Kirchhoff, "Factored Language Models and Generalized Parallel Backoff", HLT-NAACL, 2003.

[20] S. Keerthi et al, "Improvements to Platt's SMO Algorithm for SVM Classifier Design", In Neural Computation, 13(3):637-649, 2001.

[21] C. Servan et al, "Conceptual Decoding from Word Lattices: Application to the Spoken Dialog Corpus MEDIA", ICSLP, 2006.

	dialog act	content (manual transcriptions)
WOZ	`opening`	welcome on the MEDIA server how may I help you
USER		i would like to book make a reservation for the forth week of march in paris 18th district in the montmartre district two hotel rooms for two couples one with a child at 75 euro maximum rate per room and per night in a two star hotel close to a theater i would like to make a reservation
WOZ	`explicitC`	you need a reservation in Paris Montmartre for the forth week of march
USER		yes yes
WOZ	`dbQueryfinalC`	hold on... i can propose you two hotels the bouquet de montmartre room rate is 50 euros le chat noir rate is 90 euros would you like to book in one of these hotels
USER		may be may be
WOZ	`finalC`	would you like to book in one of these hotels
USER		yes yes
WOZ	`contQ`	in which hotel
USER		in the hotel with a 50 euro room rate
WOZ	`explicitC`	you want to book le bouquet de montmartre
USER		yes yes
WOZ	`inform`	hold on i check vacancies this hotel is full would you like to modify you date or book another hotel
USER		yes book another hotel
WOZ	`dbQueryfinalC`	how about le chat noir the rate is 90 euros would you like to book this hotel
USER		yes yes
WOZ	`informC`	your reservation is done the charge for your stay is 1090 euros for 2 rooms 6 nights
USER		yes yes
WOZ	`informC`	would you like another information
USER		yes yes
WOZ	`openQ`	which information
USER		i would like the hotel to be near to a theater
WOZ	`inform`	the hotel le chat noir is closed to theatre du tertre would you like another information
USER		no no
WOZ	`closing`	thank you for using our service good bye

Fig. 2. MEDIA dialog example with WOZ DA annotation.

words	mode	attribute name	normalized value
donnez-moi	+	`null`	
le	?	`refLink-coRef`	`singular`
tarif	?	`object`	`payment-amount-room`
puisque	+	`connectProp`	`imply`
je voudrais	+	`null`	
une chambre	+	`number-room`	`1`
qui coûte	+	`object`	`payment-amount-room`
pas plus de	+	`comparative-payment`	`less than`
cinquante	+	`payment-amount-integer-room`	`50`
euros	+	`payment-unit`	`euro`

Fig. 3. Semantic attribute/value representation for the sentence *"give me the rate for I'd like a room charged not more than fifty euros"* in the MEDIA corpus.

dialog act	example
`opening`	Welcome on the MEDIA telephone service...
`constrained question`	What type of room do you want?
`open question`	Do you need some other details?
`explicit confirmation`	An hotel in Paris for the 1st of July, correct?
`information`	Your hotel has a swimming pool.
`information confirmation`	Your reservation is done. The price is...
`database query`	A moment please, I'm seeking your information.
`final confirmation`	Do you want to book this hotel?
`relax question`	No double available, would you accept a single?
`closing`	Good bye and thank you
Compounds: `dbquery+constqu, dbquery+constqu+finalc, dbquery+finalc, dbquery+openqu, explicitc+constqu`	

Fig. 4. WOZ dialog acts present in the MEDIA corpus.