LEARNING USER INTENTIONS IN SPOKEN DIALOGUE SYSTEMS

Hamid R. Chinaei, Brahim Chaib-draa

Computer Science and Software Engineering Department, Laval University, Quebec, Canada
hrchinae@damas.ift.ulaval.ca, chaib@damas.ift.ulaval.ca

Luc Lamontagne

Computer Science and Software Engineering Department, Laval University, Quebec, Canada
Luc.Lamontagne@ift.ulaval.ca

Keywords: Learning, Spoken Dialogue Systems

Abstract: A common problem in spoken dialogue systems is finding the intention of the user. This problem deals with obtaining one or several topics for each transcribed, possibly noisy, sentence of the user. In this work, we apply the recent unsupervised learning method, Hidden Topic Markov Models (HTMM), for finding the intention of the user in dialogues. This technique combines two methods of Latent Dirichlet Allocation (LDA) and Hidden Markov Model (HMM) in order to learn topics of documents. We show that HTMM can be also used for obtaining intentions for the noisy transcribed sentences of the user in spoken dialogue systems. We argue that in this way we can learn possible states in a speech domain which can be used in the design stage of its spoken dialogue system. Furthermore, we discuss that the learned model can be augmented and used in a POMDP (Partially Observable Markov Decision Process) dialogue manager of the spoken dialogue system.

1 INTRODUCTION

Spoken dialogue systems are systems which help users achieve their goals via speech communication. The dialogue manager of a spoken dialogue system should maintain an efficient and natural conversation with the user. The role of a dialogue manager is to interpret the user's dialogue accurately and decides what the best action is to effectively satisfy the user intention. So, the dialogue manager of a spoken dialogue system is an agent that may have a personality (Walker et al., 1997). Examples of dialogue agents are a flight agent assisting the caller to book a flight ticket, a wheelchair directed by her patient, etc. (Williams and Young, 2007; Doshi and Roy, 2007). However, these agents have some sources of uncertainly due to automatic speech recognition and natural language understanding.

Figure 1 shows the architecture of a Spoken Dialogue System (SDS). The Automatic Speech Recognition (ASR) component receives the user's utterance (which can be a sequence of sentences) in the form of speech signals, and converts it to a sequence of transcribed noisy words. The Natural Language Understanding (NLU) component receives the transcribed
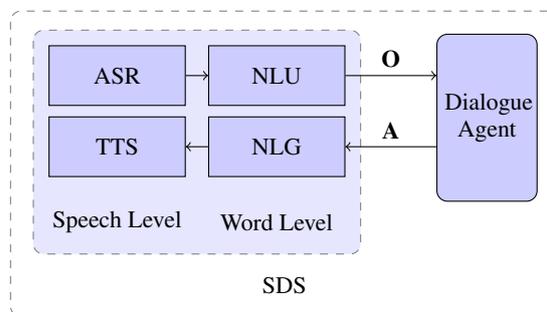

Figure 1: The architecture of a spoken dialogue system.

noisy words, and generates the possible intentions that the user could mean. The dialogue agent may receive the generated intentions with a confidence score as observation O since the output generated by Automatic Speech Recognition and Natural Language Understanding may consist of some uncertainty in the system. Based on observation O, the dialogue agent generates the action A, an input for Natural Language Generator (NLG) and Text-to-Speech (TTS) components.

Learning the intention of the user is crucial for design of a robust dialogue agent. Recent methods of design of dialogue agent rely on Markov Decision Process (MDP) framework. The basic assumption in MDPs is that the current state and action of the system determine the next state of the system (Markovian property). Partially Observable MDPs (or in short POMDPs) have been shown that are proper candidates for modeling dialogue agents (Williams et al., 2005; Doshi and Roy, 2007). POMDPs are used in the domains where in addition to the Markovian property, the environment is only partially observable for the agent; which it is the case in spoken dialogue systems.

Consider the following example taken from SACTI-II data set of dialogues (Weilhammer et al., 2004). SACTI stands for Simulated ASR-Channel: Tourist Information. Table 1 shows a sample dialogue in this corpus. The agent's observations are shown in braces. As the example shows, because of the speech recognition errors, each utterance of the user is corrupted. In POMDP framework, the user utterance can be seen as the agent's observation. And, one problem for the agent would be obtaining the user intention based on the user utterance, i.e. the agent partial observations. Without loss of generality, we can consider the user intention as the agent's state (Doshi and Roy, 2007). For instance, states could be: ask information about restaurants, hotels, bars, etc. The system observations could be the same as the states in the simplest case, and in more complex cases any word that can represent the states.

Thus, the problem would be estimating the intention of the user given the user utterance as the agent's observations. This can be seen as a typical problem in POMDPs, i.e. learning the observation model. In fact, capturing the intention of user is analogous to learning observation model in POMDPs and that the intention is analogues to the system's state in each turn of dialogue.

Blei and Moreno (2001) used aspect Hidden Markov Models for learning topics in texts. Their experimental result shows that their method is also applicable to noisy transcribed spoken dialogues. However, they assumed that the sequence of utterances is drawn from one topic and there is no notion of mixture of topics. Gruber et al. (2007) introduced Hidden Topic Markov Model (HTMM), in order to be able to introduce mixture of topics similar to PLSA (Probabilistic Latent Semantic Analysis) model (Hofmann, 1999). PLSA maps documents and words into a semantic space in which they can be compared even if they don't share any common words.

In this work, we observe that HTMM is a proper

Table 1: Sample dialogue from SACTI.

U1	Is there a good restaurant we can go to tonight
	[Is there a good restaurant week an hour tonight]
S1	Would you like an expensive restaurant
U2	No I think we'd like a medium priced restaurant
	[No I think late like uh museum price restaurant]
S2	Cheapest restaurant is eight pounds per person
U3	Can you tell me the name
	[Can you tell me the name]
S3	bochka
S4	b o c h k a
U4	Thank you can you show me on the map where it is
	[Thank you can you show me i'm there now where it is]
S5	It's here
U5	Thank you
	[Thank u]
U6	I would like to go to the museum first
	[I would like a hour there museum first]
	…

model for learning intentions behind user utterances at the word level (see Figure 1), which can be used in particular in POMDP framework. The rest of this paper is as follows. Section 2 describes the Hidden Topic Markov Models (Gruber et al., 2007), an unsupervised method for learning topics in documents. We explain the model with a focus on dialogues for the purpose of learning user intentions. This section also describes Expectation Maximization and forward backward algorithm for HTMM. In Section 3, we describe our experiment on SACTI dialogue corpus. In Section 4, we discuss our observations followed by conclusion and future directions on the project, Robotic Assistant for Persons with Disabilities[1] in Section 5.

2 HIDDEN TOPIC MARKOV MODELS FOR DIALOGUES

Hidden Topic Markov Models (HTMM) is a method which combines Hidden Markov Model (HMM) and Latent Dirichlet Analysis (LDA) for obtaining some topics for documents (Gruber et al., 2007). HMM is a framework for obtaining the hidden states based on some observation in Markovian domains such as part-of-speech tagging (Church, 1988). In LDA, similar to PLSA, the observations are explained by groups of latent variables. For instance, if we consider observations as words in a document, then the document is considered as bag of words with mixture of some topics, where topics are represented by the words with higher probabilities. In LDA as opposed to PLSA, the
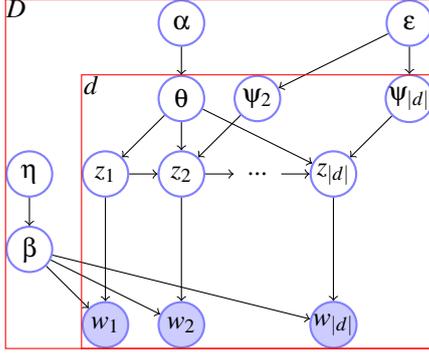
---

[1]http://damas.ift.ulaval.ca/projet.en.php

Figure 2: The HTMM model adapted from (Gruber et al., 2007), the shaded nodes are observations (w) used to capture intentions (z).

Input: Set of transcribed dialogues D, N number of intentions
Output: Finding intentions for D

1 **foreach** *intention z in the set of N intentions* **do**
2   Draw $\beta_z \sim Dirichlet(\eta)$;
3 **end**
4 **foreach** *dialogue d in D* **do**
5   Draw $\theta \sim Dirichlet(\alpha)$;

6   **foreach** $i = 1 \ldots |d|$ **do**
7   **if** *beginning of a sentence* **then**
8           $\psi_i = bernoli(\epsilon)$
9   **else**
10          $\psi_i = 0$
11   **end**
12   **end**
13   **foreach** $i = 1 \ldots |d|$ **do**
14   **if** $\psi_i = 0$ **then**
15          $z_i = z_{i-1}$
16   **else**
17          $z_i = multinomial(\theta)$
18   **end**

19   Draw $w_i \sim multinomial(\beta_{z_i})$;
20   **end**
21 **end**

Algorithm 1: The HTMM generative algorithm adapted from (Gruber et al., 2007)

mixture of topics are generated from a Dirichlet prior mutual to all documents in the corpus. Since HTMM adds the Markovian property inherited in HMM to LDA, in HTMM the dependency between successive words is regarded, and no longer the document is seen as bag of words.

In HTMM model, latent topics are found using Latent Dirichlet Allocation. The topics for a document are generated using a multinomial distribution, defined by a vector θ. The vector θ is generated using the Dirichlet prior α. Words for all documents in the corpus are generated based on multinomial distribution, defined by a vector β. The vector β is generated using the Dirichlet prior η. Figure 2 shows that the dialogue d in a dialogue set D can be seen as a sequence of words (w) which are observations for some hidden topics (z). Since hidden topics are equivalent to user intentions in our work, from now on, we call hidden topics as user intentions. The vector β is a global vector that ties all the dialogues in a dialogue set D, and retains the probability of words given user intentions. The vector θ is a local vector for each dialogue d, and retains the probability of intentions in a dialogue.

Algorithm 1 shows the process of generating and updating the parameters. First, for all possible user intentions β is drawn using the Dirichlet prior η. Then, for each dialogue, θ is drawn using the Dirichlet prior α.

The parameter ψ_i is for adding the Markovian property in dialogues since successive sentences are more likely to include the same user intention. The assumption here is that a sentence represents only one user intention, so all the words in a sentence are representative for the same user intention. To formalize that, the algorithm assigns $\psi_i = 1$ for the first word of a sentence, and $\psi_i = 0$ for the rest. Then, the intention transition is possible just when $\psi = 1$. This is represented in the algorithm between lines 6 and 18.

HTMM uses Expectation Maximization (EM) and forward backward algorithm (Rabiner, 1990), the standard method for approximating the parameters in HMMs. It is because of the fact that conditioned on θ and β, HTMM is a special case of HMMs. In HTMM, the latent variables are user intentions z_i and ψ_i which determines if the intention for the word w_i is drawn from w_{i-1}, or a new intention will be generated. In the expectation step, for each user intention z, we need to find the expected count of intention transitions to intention z.

$$E(C_{d,z}) = \sum_{j=1}^{|d|} Pr(z_{d,j} = z, \psi_{d,j} = 1 | w_1, \ldots, w_{|d|})$$

where d is a dialogue in the corpus of dialogue D. Moreover, we need to find expected number of co-occurrence of a word w with an intention z.

$$E(C^{z,w}) = \sum_{i=1}^{|D|} \sum_{j=1}^{|d_i|} Pr(z_{i,j} = z, w_{i,j} = w | w_1, \ldots, w_{|d|})$$

In the Maximization step, the MAP (Maximum A Posteriori) for θ and β is computed using Lagrange multipliers:

$$\theta_{d,z} \propto E(C_{d,z}) + \alpha - 1$$
$$\beta_{z,w} \propto E(C^{z,w}) + \eta - 1$$

The random variable $\beta_{z,w}$ gives the probability of an observation w given the intention z.

The parameter ε denotes the dependency of the sentences on each other, i.e. how likely it is that two successive uttered sentence of the user have the same intention.

$$\varepsilon = \frac{\sum_{i=1}^{|D|}\sum_{j=1}^{|d|} Pr(\psi_{i,j}=1|w_1,\ldots,w_{|d|})}{\sum_{i=1}^{|D|} N_{i,sen}}$$

where $N_{i,sen}$ is the number of sentences in the dialogue i.

In this method, EM is used for finding MAP estimate in hieratical generative model similar to LDA. Griffiths and Steyvers (2004) argued that Gibbs sampling is preferable than EM since EM can be trapped in local minima. Ortiz and Kaelbling (1999) also argued that EM suffer from local minima. However, they suggested methods for getting away from local minima. Furthermore, they also proposed that EM can be accelerated based on the type of the problem. In HTMM, the special form of the transition matrix reduce the time complexity of the algorithm to $O(|d|N^2)$, where $|d|$ is the length of the dialogue d, and N is the number of desired user intentions, given to the algorithm. The small time complexity of the algorithm enables the agent to apply it at any time to update the observation functions based on her recent observation.

3   EXPERIMENTS

We evaluated the performance of HTMM on SACTI data set (Weilhammer et al., 2004). There are about 180 dialogues between 25 users and a wizard on this corpus. The user's sentences are first confused using a speech recognition error simulator (Williams and Young, 2004; Williams et al., 2005), and then are sent to the wizard. However, the wizard's response to user is demonstrated on a screen in order to avoid speech confusion from wizard to the user. The dialogue is finished when the task is completed, or when the dialogue will last more than a limited time. This time is often more than 10 minutes. We assume that the intention transition is only possible from a sentence to the following one in a given utterance, which is more realistic than intention transition from a word to the following one within a sentence. We did our experiments on 95% dialogues with a vocabulary of 829 words, including some misspelled ones. On average, each dialogue contains of 13 sentences.

In our experiments, we removed the agent's response from the dialogues in order to test the algorithm only based on the noisy user utterances.

Table 2: Sample dialogue from SACTI.

[Is there a good restaurant week an hour tonight]
[No I think late like uh museum price restaurant]
[Can you tell me the name]
[Thank you can you show me i'm there now where it is]
[Thank u]
[I would like a hour there museum first]
…

Table 3: Sample results of experiments on SACTI.

U1	Is there a good restaurant we can go to tonight
	[Is there a good restaurant week an hour tonight]
	I4:0.9815 I2:0.0103 I1:0.0080
S1	Would you like an expensive restaurant
U2	No I think we'd like a medium priced restaurant
	[No I think late like uh museum price restaurant]
	I4:0.8930 I9:0.1005 I6:0.0041 I2:0.0015 I5:0.0005 I1:0.0001
S2	Cheapest restaurant is eight pounds per person
U3	Can you tell me the name
	[Can you tell me the name]
	I4:0.9956 I3:0.0034 I8:0.0008
S3	bochka
S4	b o c h k a
U4	Thank you can you show me on the map where it is
	[Thank you can you show me i'm there now where it is]
	I1:0.9970 I4:0.0029
S5	It's here
U5	Thank you
	[Thank u]
	I9:0.9854 I1:0.0114 I8:0.0013 I4:0.0007 I5:0.0006 I6:0.0003
U6	I would like to go to the museum first
	[I would like a hour there museum first]
	I9:0.9238 I6:0.0711 I5:0.0042 I2:0.0003 I7:0.0002
	…

Moreover, since HTTM is an unsupervised learning method, we did not have to annotate the dialogues, or any sort of preprocessing. Table 2 shows the sample dialogue in Section 1, after removing the agent's responds. As the table shows, this input data is quite corrupted. The results of our experiments show that the model is able to capture possible user intentions in the data set. Figure 3 shows 10 captured user intentions and their top 10 words. For each intention, we have highlighted the keywords which best distinguish the intention (the words which does not occur in many intentions). As Figure shows, intention 0, 1, 2, 5, and 6 represents the user asking information about tours, restaurants, hotels, museums, and bars, respectively. Intention 7, represents the user asking distance between two locations. Intentions 8 represents acknowledgement. Moreover, Intentions 3, 4, and 9 can represent hotels, restaurants, and acknowledgement, respectively. These three intentions have been previously recognized by the model; however, since the top words in each intention is slightly different, the agent assigns it in two different categories.

Intention 0		Intention 1		Intetion 2		Inteion 3		Intention 4	
is	0.0599	is	0.0703	a	0.0620	the	0.0531	the	0.0653
the	0.0523	you	0.0403	i	0.0528	you	0.0446	you	0.0488
are	0.0498	where	0.0318	i'm	0.0330	can	0.0344	me	0.0443
where	0.0361	a	0.0315	for	0.0213	me	0.0311	a	0.0441
on	0.0275	there	0.0289	uh	0.0213	please	0.0268	is	0.0389
what	0.0189	e	0.0282	looking	0.0197	of	0.0235	of	0.0267
ah	0.0177	**restaurant**	0.0270	**hotel**	0.0196	is	0.0214	**restaurant**	0.0241
at	0.0175	me	0.0267	the	0.0177	**hotel**	0.0202	can	0.0238
tours	0.0167	uh	0.0264	to	0.0162	a	0.0192	could	0.0211
i	0.0166	can	0.0262	want	0.0147	and	0.0183	where	0.0174

Intention 5		Intention 6		Intention 7		Intention 8		Intention 9	
the	0.0612	i	0.0752	how	0.0626	the	0.0351	i	0.0534
are	0.0373	a	0.0463	the	0.0495	a	0.0271	**thank**	0.0407
to	0.0259	the	0.0310	i	0.0379	**ok**	0.0265	u	0.0370
i'm	0.0235	to	0.0269	**to**	0.0360	you	0.0239	no	0.0319
in	0.0219	are	0.0261	it	0.0320	much	0.0217	you	0.0254
and	0.0210	no	0.0220	**from**	0.0304	i	0.0217	to	0.0221
um	0.0202	um	0.0202	a	0.0300	me	0.0195	think	0.0198
museum	0.0191	is	0.0170	**much**	0.0262	fine	0.0192	like	0.0181
at	0.0183	**bar**	0.0162	does	0.0240	to	0.0189	a	0.0170
a	0.0177	in	0.0161	**long**	0.0209	is	0.0172	er	0.0165

Figure 3: Captured Intentions by HTMM.

Table 3 shows highest obtained intentions for each sentence of the dialogue example in Section 1. As the table shows, the highest intention for U1 is ask information for restaurant, and with very small probability ask information for hotel. Interestingly, we can see that the obtained intention for U2 is I4, intention for restaurants, though the utterance consists of the word "museum" a strong observation for I4. This fact shows that the method is able to capture the Markovian property in U1 and U2. Another interesting observation is in U3, where the agent could estimate the user intention restaurants with 99% probability without receiving the word restaurant as observation. Yet another nice observation can be seen in the captured intentions for U4 and U5. The sentences in U4 and U5 contain "thank you" as observations. However, the captured intentions for U4 are I1 and I4, both of which represent restaurants. On the other hand, in utterance U5, the agent indeed is able to obtain intention I9, acknowledgement.

Moreover, we measured the performance of the model on the SACTI data set based on the definition of perplexity. For a learned language model on a train data set, perplexity can be considered as a measure of on average how many different equally most probable words can follow any given word, so the lower the perplexity the better the model. The perplexity of a test dialogue d after observing the first k words can be drawn using the following equation:

$$\text{Perplexity} = \exp\left(-\frac{\log Pr(w_{k+1}, \ldots, w_{|d|} | w_1, \ldots, w_k)}{|d| - k}\right)$$

To calculate the perplexity, we have:

$$Pr(w_{k+1}, \ldots, w_{|d|} | w_1, \ldots, w_k) =$$
$$\sum_i^N Pr(w_{k+1}, \ldots, w_{|d|} | z_i) Pr(z_i | w_1, \ldots, w_k)$$

where z_i is a user intention in the set of N captured user intentions from the train set. Given a user intention z_i, probability of observing $w_{k+1}, \ldots, w_{|d|}$ are independent of each other, so we have:

$$Pr(w_{k+1}, \ldots, w_{|d|} | w_1, \ldots, w_k) =$$
$$\sum_i^N \prod_{j=k+1}^{|d|} Pr(w_j | z_i) Pr(z_i | w_1, \ldots, w_k)$$

To find out the perplexity, we learned the intentions for each test dialogue d based on the first k observed words in d, i.e. $\theta_{new} = Pr(z_i | w_1, \ldots, w_k)$ is calculated for each test dialogue. However, $Pr(w_j | z_i)$ is drawn using β, learned from the train dialogues.

We calculated the perplexity for 5% of the dialogues in data set, using the 95% rest for training. Figure 4 shows the average perplexity after observing the first k sentences of test dialogues (Remember

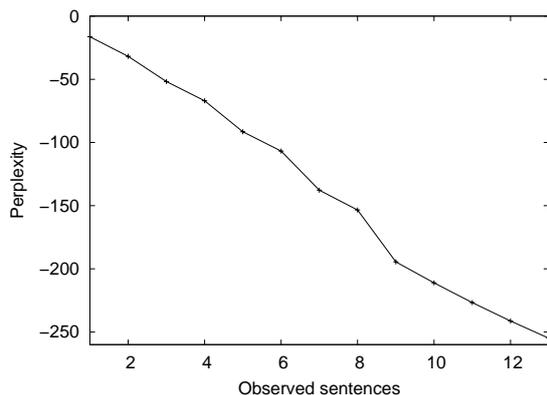Figure 4: Log likelihood for learning topics.

Figure 5: Log likelihood for learning topics.

that each sentence of the dialogue consists of only one user intention). As the figure shows, the perplexity reduce significantly by observing new sentences.

4 DISCUSSION

With the rise of spoken dialogue systems, the recent literature devoted on more robust methods of dialogue strategy design (Walker, 2000). Paek et al. (2006) evaluated the Markov assumption for spoken dialogue management. They argued that when there is not a proper estimate of reward in each state of dialogue, relaxing the Markovian assumption and estimating the total reward, using some features of the domain, could be more advantageous. Nevertheless, many researchers have found MDP and POMDP frameworks suitable for formulating a robust dialogue agent in spoken dialogue systems. In particular, Levin et al. (1997) learned dialogue strategies within the Markov Decision Process framework.

Pietquin and Dutoit (2006) used MDPs to model a dialogue agent. They interpreted the observation mostly in the speech level and based on the definition of perplexity. Williams and Young (2007) used POMDPs for modeling a dialogue agent and defined the observation function based on some features of the recognition system. However, these features are usually difficult to be determined and task dependent. We are particularly interested in the POMDP dialogue agent in (Doshi and Roy, 2007). The authors learned the observation function in a POMDP using Dirichlet distribution for the uncertainty in observation and transition parameters. However, for each state they consider only one keyword as observation.

In this work, we learned the observation model based on the received noisy data in the word level, and abstract away the speech recognition features. The used method consider all the words in a sentence as
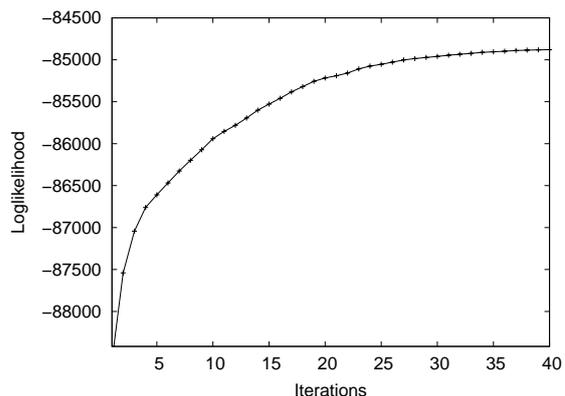
observations which represent one state. This is crucial for the frameworks such as POMDPs where the agent use an observation function to reason about the state of the system, and that the state of the system is the user intention. Based on our experiments on relatively small data set SACTI, we believe that this method can be used in the early design stage of practical dialogue agents, say in (Doshi and Roy, 2007), in order to define the possible states of the domain (possible user intentions), as well as observation function. Moreover, the result shows that HTMM is able to capture a robust observation model for practical applications with large number of observations such as (Atrash and Pineau, 2006). Moreover, since HTMM use EM algorithm, this method is quite fast, and can be used by the agent at any time to learn new observations and update the observation function. Figure 5 shows the log likelihood of data for 50 iterations of the algorithm. For the given observations, the likelihood is computed by averaging over possible states:

$$\text{loglikelihood} = \sum_{i=1}^{|D|} \sum_{j=1}^{|d_i|} \log \sum_{t=1}^{N} Pr(w_{i,j} = w | z_{i,j} = z_t)$$

As the figure shows the algorithm converges after about 30 iterations which is an evidence for small time expense of the algorithm. This fact suggests use of the algorithm after finishing some tasks by agent to learn new states, observations, and hopefully a better policy.

The interesting property about HTMM includes in its combining LDA and HMM. On the one hand, LDA captures mixture of intentions for dialogues, and on the other hand, HMM adds the Markovian property. This makes the framework similar to POMDPs in terms of making a belief over possible states, besides the Markovian property. As Table 3 shows, the possible captured intentions for each sentence of the user can be seen as the agent's belief over possible

Figure 6: Smart Wheeler Platform.

Figure 7: The extended HTMM. The intention of user depends on both her words w_is and system actions a_{mi}s.

states. Moreover, using this method, we learned the value of $\varepsilon = 0.71$ on SACTI data set; which suggests that it is likely that the user changes his intention in a dialogue in SACTI data data set; whereas for instance Doshi and Roy (2007) assumes that the user may change his intention with a predefined low probability in wheelchair domain.

HTMM, however, assumes that the Dirichlet prior are known. During our experiments on SACTI, we observed that by feeding the Algorithm 1 with different α and η, the algorithm can derive slightly different intentions. However, some of these intentions makes sense, for instance intentions for cost, dialogue initiation, etc. Moreover, the number of intentions (N in Algorithm 1) need to be set. For instance, in our experiments we set $N = 10$ to be able to derive the maximum number of intentions, yet some intentions seem to be similar.

5 CONCLUSION AND FUTURE WORKS

Although we did not perform any quantitative comparison, our early observations on SACTI data set is promising. We observed that HTMM can be used for capturing possible user intentions in dialogues. The captured intentions together with the learned observation function could be used in design stage of a POMDP based dialogue agent. Moreover, the dialogue agent can use HTMM on the captured dialogues over time to update the observation function. Although, there is no notion of actions in HTMM, and it is a method which is used mostly on static data, the similarity of HTMM and POMDP in terms of Markovian property and generating a belief over possible states suggest considering both these two models in practical applications, where the time complexity of POMDPs burden the problem. Moreover, our observation on SACTI data set suggests future works for use of HTMM for automatically annotating the cor-
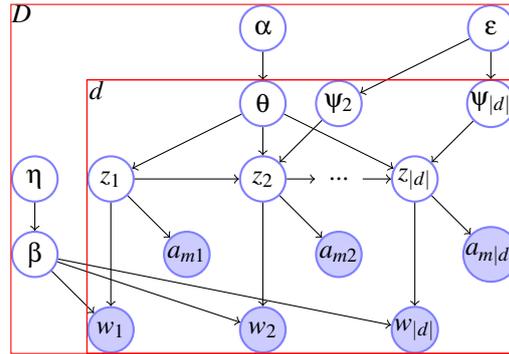
pus of dialogues, capturing the structure of dialogues, and evaluation of dialogue agents (Walker and Passonneau, 2001; Walker et al., 1997; Singh et al., 2000; Walker et al., 2001).

In the future work we are going to use HTMM to learn the model for a POMDP dialogue agent. For instance, we would like to use HTMM for a a wheelchair robot similar to Figure 6, taken from (Pineau and Atrash, 2007). This wheelchair is designed for patients with limited skills, say patients suffering from Multiple Sclerosis. The patients can direct the robot, with mentioning the goal, the path, and restrictions such as speed, instead of using a joystick.

We are going to augment HTMM by considering actions of the system in the model. Since the actions performed by the agent carries much less noise comparing to the user utterance (agent's observations), agent's action can have more effect on the Markovian property of the environment. That is, the intention of user depends on both her words and agent actions. Figure 7 shows HTMM augmented with system actions. We are going to apply augmented model on the captured dialogues for a dialogue POMDP agent and compare the agent's learned strategy with that of similar models.

ACKNOWLEDGMENT

REFERENCES

Atrash, A. and Pineau, J. (2006). Efficient planning and tracking in pomdps with large observation spaces. In *AAAI-06 Workshop on Empirical and*

Statistical Approaches for Spoken Dialogue Systems.

Blei, D. M. and Moreno, P. J. (2001). Topic segmentation with an aspect hidden Markov model. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '01)*, pages 343–348.

Church, K. W. (1988). A stochastic parts program and noun phrase parser for unrestricted text. In *Proceedings of the second conference on Applied Natural Language Processing (ANLP '88)*, pages 136–143, Morristown, NJ, USA.

Doshi, F. and Roy, N. (2007). Efficient model learning for dialog management. In *Proceedings of the ACM/IEEE international conference on Human-Robot Interaction (HRI '07)*, pages 65–72.

Griffiths, T. and Steyvers, J. (2004). Finding scientific topics. *Proceedings of the National Academy of Science*, 101:5228–5235.

Gruber, A., Rosen-Zvi, M., and Weiss, Y. (2007). Hidden Topic Markov Models. In *Artificial Intelligence and Statistics (AISTATS '07)*, San Juan, Puerto Rico.

Hofmann, T. (1999). Probabilistic latent semantic analysis. In *Proceedings of the fifteenth conference on Uncertainty in Artificial Intelligence (UAI '99)*, pages 289–296.

Levin, E., Pieraccini, R., and Eckert, W. (1997). Learning dialogue strategies within the Markov decision process framework. *1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 72–79.

Ortiz, L. E. and Kaelbling, L. P. (1999). Accelerating EM: An empirical study. In *Proceedings of the fifteenth conference on Uncertainty in Artificial Intelligence (UAI '99)*, pages 512–521, Stockholm, Sweden.

Paek, Tim, Chickering, and David (2006). Evaluating the Markov assumption in Markov Decision Processes for spoken dialogue management. *Language Resources and Evaluation*, 40(1):47–66.

Pietquin, O. and Dutoit, T. (2006). A probabilistic framework for dialog simulation and optimal strategy learning. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(2):589–599.

Pineau, J. and Atrash, A. (2007). Smartwheeler: A robotic wheelchair test-bed for investigating new models of human-robot interaction. In *AAAI Spring Symposium on Multidisciplinary Collaboration for Socially Assistive Robotics*.

Rabiner, L. R. (1990). A tutorial on hidden Markov models and selected applications in speech recognition. pages 267–296.

Singh, S. P., Kearns, M. J., Litman, D. J., and Walker, M. A. (2000). Empirical evaluation of a reinforcement learning spoken dialogue system. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 645–651. AAAI Press / The MIT Press.

Walker, M. and Passonneau, R. (2001). DATE: a dialogue act tagging scheme for evaluation of spoken dialogue systems. In *Proceedings of the first international conference on Human Language Rechnology research (HLT '01)*, pages 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

Walker, M. A. (2000). An application of reinforcement learning to dialogue strategy selection in a spoken dialogue system for email. *Journal of Artificial Intelligence Research (JAIR)*, 12:387–416.

Walker, M. A., Litman, D. J., Kamm, A. A., and Abella, A. (1997). PARADISE: A Framework for Evaluating Spoken Dialogue Agents. In *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 271–280, Somerset, New Jersey. Association for Computational Linguistics.

Walker, M. A., Passonneau, R. J., and Boland, J. E. (2001). Quantitative and qualitative evaluation of darpa communicator spoken dialogue systems. In *Meeting of the Association for Computational Linguistics*, pages 515–522.

Weilhammer, K., Williams, J. D., and Young, S. (2004). The SACTI-2 Corpus: Guide for Research Users, Cambridge University. Technical report.

Williams, J. D., Poupart, P., and Young, S. (2005). Factored partially observable markov decision processes for dialogue management. In *The 4th IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, Edinburgh, Scotland.

Williams, J. D. and Young, S. (2004). Characterizing task-oriented dialog using a simulated asr channel. In *Proceedings of International Conference on Spoken Language Processing (ICSLP '04), Jeju, South Korea*.

Williams, J. D. and Young, S. (2007). Partially observable markov decision processes for spoken dialog systems. *Computer Speech and Language*, 21:393–422.