Leveraging Hidden Dialogue State to Select Tutorial Moves

Kristy Elizabeth Boyer[a] **Robert Phillips**[ab] **Eun Young Ha**[a] **Michael D. Wallis**[ab] **Mladen A. Vouk**[a] **James C. Lester**[a]

[a]Department of Computer Science, North Carolina State University
[b]Applied Research Associates
Raleigh, NC, USA

{keboyer, rphilli, eha, mdwallis, vouk, lester}@ncsu.edu

Abstract

A central challenge for tutorial dialogue systems is selecting an appropriate move given the dialogue context. Corpus-based approaches to creating tutorial dialogue management models may facilitate more flexible and rapid development of tutorial dialogue systems and may increase the effectiveness of these systems by allowing data-driven adaptation to learning contexts and to individual learners. This paper presents a family of models, including first-order Markov, hidden Markov, and hierarchical hidden Markov models, for predicting tutor dialogue acts within a corpus. This work takes a step toward fully data-driven tutorial dialogue management models, and the results highlight important directions for future work in unsupervised dialogue modeling.

1 Introduction

A central challenge for dialogue systems is selecting appropriate system dialogue moves (Bangalore, Di Fabbrizio, & Stent, 2008; Frampton & Lemon, 2009; Young et al., 2009). For tutorial dialogue systems, which aim to support learners during conceptual or applied learning tasks, selecting an appropriate dialogue move is particularly important because the tutorial approach could significantly influence cognitive and affective outcomes for the learner (Chi, Jordan, VanLehn, & Litman, 2009). The strategies implemented in tutorial dialogue systems have historically been based on handcrafted rules derived from observing human tutors (e.g., Aleven, McLaren, Roll, & Koedinger, 2004; Evens & Michael, 2006; Graesser, Chipman, Haynes, & Olney, 2005; Jordan, Makatchev, Pappuswamy, VanLehn, & Albacete, 2006). While these systems can achieve results on par with unskilled human tutors, tutorial dialogue systems have not yet matched the effectiveness of expert human tutors (VanLehn et al., 2007).

A more flexible model of strategy selection may enable tutorial dialogue systems to increase their effectiveness by responding adaptively to a broader range of contexts. A promising method for deriving such a model is to learn it directly from corpora of effective human tutoring. Data-driven approaches have shown promise in task-oriented domains outside of tutoring (Bangalore et al., 2008; Hardy et al., 2006; Young et al., 2009), and automatic dialogue policy creation for tutoring has been explored recently (Chi, Jordan, VanLehn, & Hall, 2008; Tetreault & Litman, 2008). Ultimately, devising data-driven approaches for developing tutorial dialogue systems may constitute a key step towards achieving the high learning gains that have been observed with expert human tutors.

The work presented in this paper focuses on learning a model of tutorial moves within a corpus of human-human dialogue in the task-oriented domain of introductory computer science. Unlike the majority of task-oriented domains that have been studied to date, our domain involves the separate creation of a persistent artifact by the user (the student). The modification of this artifact, in our case a computer program, is the focus of the dialogues. Our corpus consists of textual dialogue utterances and a separate synchronous stream of

task actions. Our goal is to extract a data-driven dialogue management model from the corpus, as evidenced by predicting system (tutor) dialogue acts.

In this paper, we present an annotation approach that addresses dialogue utterances and task actions, and we propose a unified sequential representation for these separate synchronous streams of events. We explore the predictive power of three stochastic models — first-order Markov models, hidden Markov models, and hierarchical hidden Markov models — for predicting tutor dialogue acts in the unified sequences. By leveraging these models to capture effective tutorial dialogue strategies, this work takes a step toward creating data-driven tutorial dialogue management models.

2 Related Work

Much of the research on selecting system dialogue acts relies on a Markov assumption (Levin, Pieraccini, & Eckert, 2000). This formulation is often used in conjunction with reinforcement learning (RL) to derive optimal dialogue policies (Frampton & Lemon, 2009). Sparse data and large state spaces can pose serious obstacles to RL, and recent work aims to address these issues (Ai, Tetreault, & Litman, 2007; Henderson, Lemon, & Georgila, 2008; Heeman, 2007; Young et al., 2009). For tutorial dialogue, RL has been applied to selecting a state space representation that best facilitates learning an optimal dialogue policy (Tetreault & Litman, 2008). RL has also been used to compare specific tutorial dialogue tactic choices (Chi et al., 2008).

While RL learns a dialogue policy through exploration, our work assumes that a flexible, good (though possibly not *optimal*) dialogue policy is realized in successful human-human dialogues. We extract this dialogue policy by predicting tutor (system) actions within a corpus. Using human dialogues directly in this way has been the focus of work in other task-oriented domains such as finance (Hardy et al., 2006) and catalogue ordering (Bangalore et al., 2008). Like the parse-based models of Bangalore et al., our hierarchical hidden Markov models (HHMM) explicitly capture the hierarchical nesting of tasks and subtasks in our domain. In other work, this level of structure has been studied from a slightly different perspective as conversational game (Poesio & Mikheev, 1998).

For tutorial dialogue, there is compelling evidence that human tutoring is a valuable model for extracting dialogue system behaviors. The CIRCSIM-TUTOR (Evens & Michael, 2006), ITSPOKE (Forbes-Riley, Rotaru, Litman, & Tetreault, 2007; Forbes-Riley & Litman, 2009), and KSC-PAL (Kersey, Di Eugenio, Jordan, & Katz, 2009) projects have made extensive use of data-driven techniques based on human corpora. Perhaps most directly comparable to the current work are the bigram models of Forbes-Riley et al.; we explore first-order Markov models, which are equivalent to bigram models, for predicting tutor dialogue acts. In addition, we present HMMs and HHMMs trained on our corpus. We found that both of these models outperformed the bigram model for predicting tutor moves.

3 Corpus and Annotation

The corpus was collected during a human-human tutoring study in which tutors and students worked to solve an introductory computer programming problem (Boyer et al., in press). The dialogues were effective: on average, students exhibited a 7% absolute gain from pretest to posttest (N=48, paired t-test p<0.0001).

The corpus contains 48 textual dialogues with a separate, synchronous task event stream. Tutors and students collaborated to solve an introductory computer programming problem using an online tutorial environment with shared workspace viewing and textual dialogue. Each student participated in exactly one tutoring session. The corpus contains 1,468 student utterances, 3,338 tutor utterances, and 3,793 student task actions. In order to build the dialogue model, we annotated the corpus with dialogue act tags and task annotation labels.

3.1 Dialogue Act Annotation

We have developed a dialogue act tagset inspired by schemes for conversational speech (Stolcke et al., 2000), task-oriented dialogue (Core & Allen, 1997), and tutoring (Litman & Forbes-Riley, 2006). The dialogue act tags are displayed in Table 1. Overall reliability on 10% of the corpus for two annotators was κ=0.80.

Table 1. Dialogue act tags

DA	Description	Stu. Rel. Freq.	Tut. Rel. Freq.	κ
ASSESSING QUESTION (AQ)	Request for feedback on task or conceptual utterance.	.20	.11	.91
EXTRA-DOMAIN (EX)	Asides not relevant to the tutoring task.	.08	.04	.79
GROUNDING (G)	Acknowledgement/thanks	.26	.06	.92
LUKEWARM CONTENT FEEDBACK (LCF)	Negative assessment with explanation.	.01	.03	.53
LUKEWARM FEEDBACK (LF)	Lukewarm assessment of task action or conceptual utterance.	.02	.03	.49
NEGATIVE CONTENT FEEDBACK (NCF)	Negative assessment with explanation.	.01	.10	.61
NEGATIVE FEEDBACK (NF)	Negative assessment of task action or conceptual utterance.	.05	.02	.76
POSITIVE CONTENT FEEDBACK (PCF)	Positive assessment with explanation.	.02	.03	.43
POSITIVE FEEDBACK (PF)	Positive assessment of task action or conceptual utterance.	.09	.16	.81
QUESTION (Q)	Task or conceptual question.	.09	.03	.85
STATEMENT (S)	Task or conceptual assertion.	.16	.41	.82

3.2 Task Annotation

The dialogues focused on the task of solving an introductory computer programming problem. The task actions were recorded as a separate but synchronous event stream. This stream included 97,509 keystroke-level user task events. These events were manually aggregated and annotated for subtask structure and then for correctness. The task annotation scheme was hierarchical, reflecting the nested nature of the subtasks. An excerpt from the task annotation scheme is depicted in Figure 1; the full scheme contains 66 leaves. The task annotation scheme was designed to reflect the different depth of possible subtasks nested within the overall task. Each labeled task action was also judged for correctness according to the requirements of the task, with categories CORRECT, BUGGY, INCOMPLETE, and DISPREFERRED (technically correct but not accomplishing the pedagogical goals of the task).

Each group of task keystrokes that occurred between dialogue utterances was tagged, possibly with many subtask labels, by a human judge. A second judge tagged 20% of the corpus in a reliability study for which one-to-one subtask identification was not enforced (giving judges maximum flexibility to apply the tags). To ensure a conservative reliability statistic, all unmatched subtask tags were treated as disagreements. The resulting unweighted kappa statistic was κ_{simple}= 0.58, but the weighted Kappa $\kappa_{weighted}$=0.86 is more meaningful because it takes into account the ordinal nature of the labels that result from sequential subtasks. On task actions for which the two judges agreed on subtask tag, the agreement statistic for correctness was κ_{simple}=0.80.

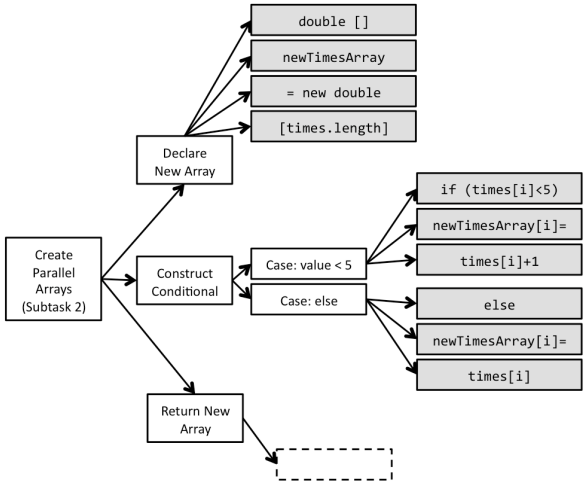

Figure 1. Portion of task annotation scheme

3.3 Adjacency Pair Joining

Some dialogue acts establish an expectation for another dialogue act to occur next (Schegloff & Sacks, 1973). Our previous work has found that identifying the statistically significant *adjacency pairs* in a corpus and joining them as atomic observations prior to model building produces more interpretable descriptive models. The models reported here were trained on hybrid sequences of dialogue acts and adjacency pairs. A full description of the adjacency pair identification methodology and joining algorithm is reported in (Boyer et al., 2009). A partial list of the most highly statistically significant adjacency pairs,

which for this work include task actions, is displayed in Table 2.

Table 2. Subset of significant adjacency pairs

CorrectTaskAction-CorrectTaskAction;
ExtraDomain$_S$-ExtraDomain$_T$; Grounding$_S$-Grounding$_T$;
AssessingQuestion$_T$-PositiveFeedback$_S$;
AssessingQuestion$_S$-PositiveFeedback$_T$; Question$_T$-Statements$_S$;
AssessingQuestion$_T$-Statement$_S$; ExtraDomain$_T$-ExtraDomain$_S$;
Question$_S$-Statement$_T$; NegativeFeedback$_S$-Grounding$_T$;
IncompleteTaskAction-IncompleteTaskAction;
PositiveFeedback$_S$-Grounding$_T$;
BuggytaskAction-BuggyTaskAction

4 Models

We learned three types of models using cross-validation with systematic sampling of training and testing sets.

4.1 First-Order Markov Model

The simplest model we discuss is the first-order Markov model (MM), or bigram model (Figure 2). A MM that generates observation (state) sequence $o_1o_2...o_t$ is defined in the following way. The observation symbols are drawn from the alphabet $\sum=\{\sigma_1, \sigma_2, ..., \sigma_M\}$, and the initial probability distribution is $\Pi=[\pi_i]$ where π_i is the probability of a sequence beginning with observation symbol σ_i. The transition probability distribution is A=$[a_{ij}]$, where a_{ij} is the probability of observation j occurring immediately after observation i.

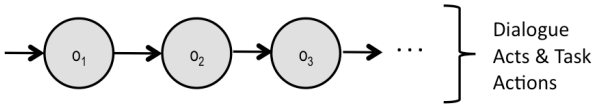

Figure 2. Time-slice topology of MM

We trained MMs on our corpus of dialogue acts and task events using ten-fold cross-validation to produce a model that could be queried for the next predicted tutorial dialogue act given the history.

4.2 Hidden Markov Model

A hidden Markov model (HMM) augments the MM framework, resulting in a doubly stochastic structure (Rabiner, 1989). For a first-order HMM, the observation symbol alphabet is defined as above, along with a set of hidden states $S=\{s_1,s_2,...,s_N\}$. The transition and initial probability distributions are defined analogously to MMs, except that they operate on hidden states rather than on observation symbols (Figure 3). That is, $\Pi=[\pi_i]$ where π_i is the probability of a sequence beginning in hidden state s_i. The transition matrix is $A=[a_{ij}]$, where a_{ij} is the probability of the model transitioning from hidden state i to hidden state j. This framework constitutes the first stochastic layer of the model, which can be thought of as modeling hidden, or unobservable, structure. The second stochastic layer of the model governs the production of observation symbols: the emission probability distribution is $B=[b_{ik}]$ where b_{ik} is the probability of state i emitting observation symbol k.

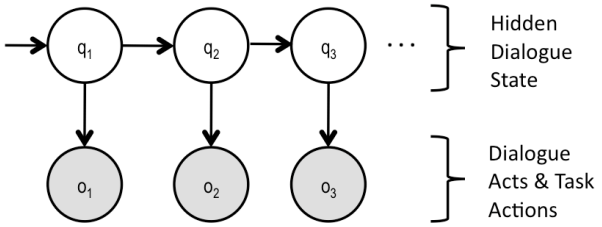

Figure 3. Time-slice topology of HMM

The notion that dialogue has an overarching unobservable structure that influences the observations is widely accepted. In tutoring, this overarching structure may correspond to tutorial strategies. We have explored HMMs' descriptive power for extracting these strategies (Boyer et al., 2009), and this paper explores the hypothesis that HMMs provide better predictive power than MMs on our dialogue sequences. We trained HMMs on the corpus using the standard Baum-Welch expectation maximization algorithm and applied state labels that reflect post-hoc interpretation (Figure 4).

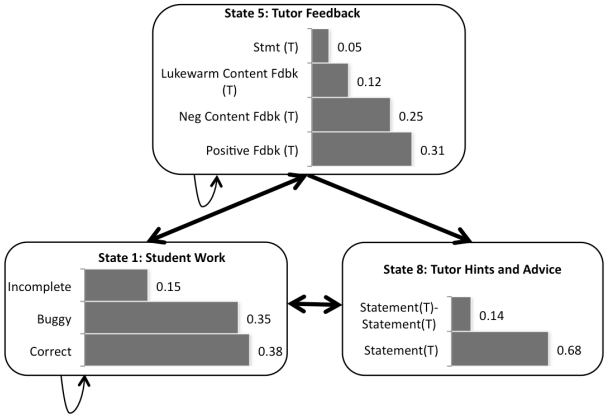

Figure 4. Portion of learned HMM

4.3 Hierarchical Hidden Markov Model

Hierarchical hidden Markov models (HHMMs) allow for explicit representation of multilevel stochastic structure. A complete formal definition of HHMMs can be found in (Fine, Singer, & Tishby, 1998), but here we present an informal description. HHMMs include two types of hidden states: internal nodes, which do not produce observation symbols, and production nodes, which do produce observations. An internal node includes a set of substates that correspond to its potential children, $S=\{s_1, s_2, \ldots, s_N\}$, each of which is itself the root of an HHMM. The initial probability distribution $\Pi=[\pi_i]$ for each internal node governs the probability that the model will make a vertical transition to substate s_i from this internal node; that is, that this internal node will produce substate s_i as its leftmost child. Horizontal transitions are governed by a transition probability distribution similar to that described above for flat HMMs. Production nodes are defined by their observation symbol alphabet and an emission probability distribution over the symbols; HHMMs do not require a global observation symbol alphabet. The generative topology of our HHMMs is illustrated in Figure 5.

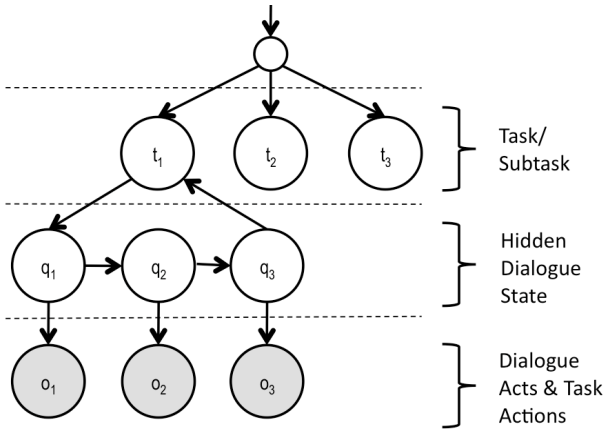

Figure 5. Generative topology of HHMM

HHMMs of arbitrary topology can be trained using a generalized version of the Baum-Welch algorithm (Fine et al., 1998). Our HHMMs featured a pre-specified model topology based on known task/subtask structure. A Bayesian view of a portion of the best-fit HHMM is depicted in Figure 6. This model was trained using five-fold cross-validation to address the absence of symbols from the training set that were present in the testing set, a sparsity problem that arose from splitting the data hierarchically.

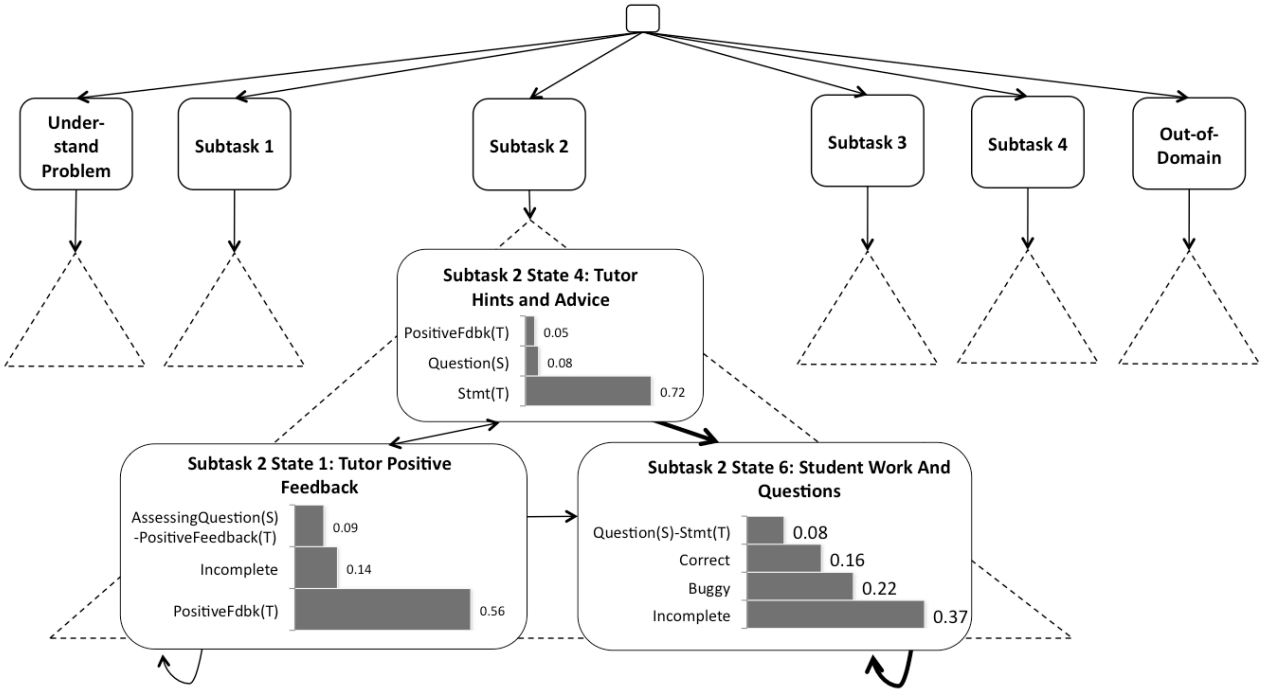

Figure 6. Portion of learned HHMM

5 Results

We trained and tested MMs, HMMs, and HHMMs on the corpus and compared prediction accuracy for tutorial dialogue acts by providing the model with partial sequences from the test set and querying for the next tutorial move. The baseline prediction accuracy for this task is 41.1%, corresponding to the most frequent tutorial dialogue act (STATEMENT). As depicted in Figure 7, a first-order MM performed worse than baseline (p<0.001)[1] at 27% average prediction accuracy ($\hat{\sigma}_{MM}$=6%). HMMs performed better than baseline (p<0.0001), with an average accuracy of 48% ($\hat{\sigma}_{HMM}$=3%). HHMMs averaged 57% accuracy, significantly higher than baseline (p=0.002) but weakly significantly higher than HMMs (p=0.04), and with high variation ($\hat{\sigma}_{HHMM}$=23%).

Figure 7. Average prediction accuracies of three model types on tutor dialogue acts

To further explore the performance of the HHMMs, Figure 8 displays their prediction accuracy on each of six labeled subtasks. These subtasks correspond to the top level of the hierarchical task/subtask annotation scheme. The UNDERSTAND THE PROBLEM subtask corresponds to the initial phase of most tutoring sessions, in which the student and tutor agree to some extent on a problem-solving plan. Subtasks 1, 2, and 3 account for the implementation and debugging of three distinct modules within the learning task, and Subtask 4 involves testing and assessing the student's finalized program. The EXTRA-DOMAIN subtask involves side conversations whose topics are outside of the domain.

The HHMM performed as well as or better (p<0.01) than baseline on the first three in-domain subtasks. The performance on SUBTASK 4 was not distinguishable from baseline (p=0.06); relatively few students reached this subtask. The model did not outperform baseline (p=0.40) for the UNDERSTAND THE PROBLEM subtask, and qualitative inspection of the corpus reveals that the dialogue during this phase of tutoring exhibits limited regularities between students.

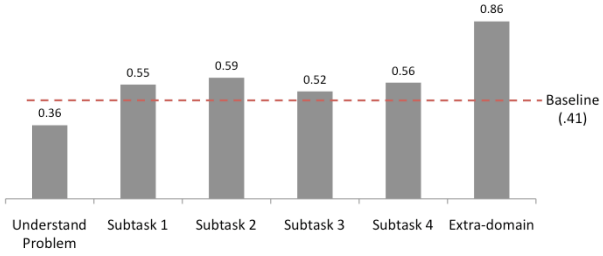

Figure 8. Average prediction accuracies of HHMMs by subtask

6 Discussion

The results support our hypothesis that HMMs, because of their capacity for explicitly representing dialogue structure at an abstract level, perform better than MMs for predicting tutor moves. The results also suggest that explicitly modeling hierarchical task structure can further improve prediction accuracy of the model. The below-baseline performance of the bigram model illustrates that in our complex task-oriented domain, an immediately preceding event is not highly predictive of the next move. While this finding may not hold for conversational dialogue or some task-oriented dialogue with a more balanced distribution of utterances between speakers, the unbalanced nature of our tutoring sessions may not be as easily captured.

In our corpus, tutor utterances outnumber student utterances by more than two to one. This large difference is due to the fact that tutors frequently guided students and provided multi-turn explanations, the impetus for which are not captured in the corpus, but rather, involve external pedagogical goals. The MM, or bigram model, has no mechanism for capturing this layer of stochastic behavior. On the other hand, the HMM can account for unobserved influential variables, and the HHMM can do so to an even greater extent by explicitly modeling task/subtask structure.

Considering the performance of the HHMM on individual subtasks reveals interesting properties of our dialogues. First, the HHMM is unable to outperform baseline on the UNDERSTAND THE PROBLEM subtask. To address this issue, our ongoing work investigates taking into account

[1] All p-values in this section were produced by two-sample one-tailed t-tests with unequal sample variances.

student characteristics such as incoming knowledge level and self-confidence. On all four in-domain subtasks, the HHMM achieved a 30% to 50% increase over baseline. For extra-domain dialogues, which involve side conversations that are not task-related, the HHMM achieved 86% prediction accuracy on tutor moves, which constitutes a 115% improvement over baseline. This high accuracy may be due in part to the fact that out-of-domain asides were almost exclusively initiated by the student, and tutors rarely engaged in such exchanges beyond providing a single response. This regularity likely facilitated prediction of the tutor's dialogue moves during out-of-domain talk.

We are aware of only one recent project that reports extensively on predicting system actions from a corpus of human-human dialogue. Bangalore et al.'s (2008) flat task/dialogue model in a catalogue-ordering domain achieved a prediction accuracy of 55% for system dialogue acts, a 175% improvement over baseline. When explicitly modeling the hierarchical task/subtask dialogue structure, they report a prediction accuracy of 35.6% for system moves, approximately 75% above baseline (Bangalore & Stent, 2009). These findings were obtained by utilizing a variety of lexical and syntactic features along with manually annotated dialogue acts and task/subtask labels. In comparison, our HHMM achieved an average 42% improvement over baseline using only annotated dialogue acts and task/subtask labels. In ongoing work we are exploring the utility of additional features for this prediction task.

Our best model performed better than baseline by a significant margin. The absolute prediction accuracy achieved by the HHMM was 57% across the corpus, which at first blush may appear too low to be of practical use. However, the choice of tutorial move involves some measure of subjectivity, and in many contexts there may be no uniquely appropriate dialogue act. Work in other domains has dealt with this uncertainty by maintaining multiple hypotheses (Wright Hastie, Poesio, & Isard, 2002) and by mapping to clustered sets of moves rather than maintaining policies for each possible system selection (Young et al., 2009). Such approaches may prove useful in our domain as well, and may help to more fully realize the potential of a learned dialogue management model.

7 Conclusion and Future Work

Learning models that predict system moves within a corpus is a first step toward building fully data-driven dialogue management models. We have presented Markov models, hidden Markov models, and hierarchical hidden Markov models trained on sequences of manually annotated dialogue acts and task events. Of the three models, the hierarchical models appear to perform best in our domain, which involves an intrinsically hierarchical task/subtask structure.

The models' performance points to promising future work that includes utilizing additional lexical and syntactic features along with fixed user (student) characteristics within a hierarchical hidden Markov modeling framework. More broadly, the results point to the importance of considering task structure when modeling a complex domain such as those that often accompany task-oriented tutoring. Finally, a key direction for data-driven dialogue management models involves learning unsupervised dialogue act and task classification models.

Acknowledgements. This work is supported in part by the North Carolina State University Department of Computer Science and the National Science Foundation through a Graduate Research Fellowship and Grants CNS-0540523, REC-0632450 and IIS-0812291. Any opinions, findings, conclusions, or recommendations expressed in this report are those of the participants, and do not necessarily represent the official views, opinions, or policy of the National Science Foundation.

References

Ai, H., Tetreault, J. R., & Litman, D. J. (2007). Comparing user simulation models for dialog strategy learning. *Proceedings of NAACL HLT, Companion Volume,* Rochester, New York. 1-4.

Aleven, V., McLaren, B., Roll, I., & Koedinger, K. (2004). Toward tutoring help seeking: Applying cognitive modeling to meta-cognitive skills. *Proceedings of ITS,* 227-239.

Bangalore, S., Di Fabbrizio, G., & Stent, A. (2008). Learning the structure of task-driven human-human dialogs. *IEEE Transactions on Audio, Speech, and Language Processing, 16*(7), 1249-1259.

Bangalore, S., & Stent, A. J. (2009). Incremental parsing models for dialog task structure. *Proceedings of the EACL,* 94-102.

Boyer, K. E., Phillips, R., Ha, E. Y., Wallis, M. D., Vouk, M. A., & Lester, J. C. (2009). Modeling dialogue structure with adjacency pair analysis and hidden Markov models. *Proceedings of NAACL HLT (Short Papers),* 19-26.

Boyer, K. E., Phillips, R., Ingram, A., Ha, E. Y., Wallis, M. D., Vouk, M. A., & Lester, J. C. (In press). Characterizing the effectiveness of tutorial dialogue with hidden Markov models. *Proceedings of ITS,* Pittsburgh, Pennsylvania.

Chi, M., Jordan, P., VanLehn, K., & Hall, M. (2008). Reinforcement learning-based feature selection for developing pedagogically effective tutorial dialogue tactics. *Proceedings of EDM,* Montreal, Canada. 258-265.

Chi, M., Jordan, P., VanLehn, K., & Litman, D. (2009). To elicit or to tell: Does it matter? *Proceedings of AIED,* 197-204.

Core, M., & Allen, J. (1997). Coding dialogs with the DAMSL annotation scheme. *AAAI Fall Symposium on Communicative Action in Humans and Machines,* 28–35.

Evens, M., & Michael, J. (2006). *One-on-one tutoring by humans and computers*. Mahwah, New Jersey: Lawrence Erlbaum Associates.

Fine, S., Singer, Y., & Tishby, N. (1998). The hierarchical hidden Markov model: Analysis and applications. *Machine Learning, 32*(1), 41-62.

Forbes-Riley, K., Rotaru, M., Litman, D. J., & Tetreault, J. (2007). Exploring affect-context dependencies for adaptive system development. *Proceedings of NAACL HLT (Short Papers),* 41-44.

Forbes-Riley, K., & Litman, D. (2009). Adapting to student uncertainty improves tutoring dialogues. *Proceedings of AIED,* 33-40.

Frampton, M., & Lemon, O. (2009). Recent research advances in reinforcement learning in spoken dialogue systems. *The Knowledge Engineering Review, 24*(4), 375-408.

Graesser, A. C., Chipman, P., Haynes, B. C., & Olney, A. (2005). AutoTutor: An intelligent tutoring system with mixed-initiative dialogue. *IEEE Transactions on Education, 48*(4), 612-618.

Hardy, H., Biermann, A., Inouye, R. B., McKenzie, A., Strzalkowski, T., Ursu, C., Webb, N., & Wu, M. (2006). The Amitiés system: Data-driven techniques for automated dialogue. *Speech Communication, 48*(3-4), 354-373.

Heeman, P. A. (2007). Combining reinforcement learning with information-state update rules. *Proceedings of NAACL HLT,* 268-275.

Henderson, J., Lemon, O., & Georgila, K. (2008). Hybrid reinforcement/supervised learning of dialogue policies from fixed data sets. *Computational Linguistics, 34*(4), 487-511.

Jordan, P., Makatchev, M., Pappuswamy, U., VanLehn, K., & Albacete, P. (2006). A natural language tutorial dialogue system for physics. *Proceedings of FLAIRS,* 521-526.

Kersey, C., Di Eugenio, B., Jordan, P., & Katz, S. (2009). KSC-PaL: A peer learning agent that encourages students to take the initiative. *Proceedings of the NAACL HLT Workshop on Innovative use of NLP for Building Educational Applications,* Boulder, Colorado. 55-63.

Levin, E., Pieraccini, R., & Eckert, W. (2000). A stochastic model of human-machine interaction for learning dialog strategies. *IEEE Transactions on Speech and Audio Processing, 8*(1), 11-23.

Litman, D., & Forbes-Riley, K. (2006). Correlations between dialogue acts and learning in spoken tutoring dialogues. *Natural Language Engineering, 12*(2), 161-176.

Poesio, M., & Mikheev, A. (1998). The predictive power of game structure in dialogue act recognition: Experimental results using maximum entropy estimation. *Proceedings of ICSLP*, 90-97.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE, 77*(2), 257-286.

Schegloff, E., & Sacks, H. (1973). Opening up closings. *Semiotica, 7*(4), 289-327.

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Van Ess-Dykema, C., & Meteer, M. (2000). Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics, 26*(3), 339-373.

Tetreault, J. R., & Litman, D. J. (2008). A reinforcement learning approach to evaluating state representations in spoken dialogue systems. *Speech Communication, 50*(8-9), 683-696.

VanLehn, K., Graesser, A. C., Jackson, G. T., Jordan, P., Olney, A., & Rose, C. P. (2007). When are tutorial dialogues more effective than reading? *Cognitive Science, 31*(1), 3-62.

Wright Hastie, H., Poesio, M., & Isard, S. (2002). Automatically predicting dialogue structure using prosodic features. *Speech Communication, 36*(1-2), 63-79.

Young, S., Gasic, M., Keizer, S., Mairesse, F., Schatzmann, J., Thomson, B., & Yu, K. (2009). The hidden information state model: A practical framework for POMDP-based spoken dialogue management. *Computer Speech and Language, 24*(2), 150-174.