Combining Task and Dialogue Streams in Unsupervised Dialogue Act Models

Aysu Ezen-Can and Kristy Elizabeth Boyer
Department of Computer Science
North Carolina State University
aezen,keboyer@ncsu.edu

Abstract

Unsupervised machine learning approaches hold great promise for recognizing dialogue acts, but the performance of these models tends to be much lower than the accuracies reached by supervised models. However, some dialogues, such as task-oriented dialogues with parallel task streams, hold rich information that has not yet been leveraged within unsupervised dialogue act models. This paper investigates incorporating task features into an unsupervised dialogue act model trained on a corpus of human tutoring in introductory computer science. Experimental results show that incorporating task features and dialogue history features significantly improve unsupervised dialogue act classification, particularly within a hierarchical framework that gives prominence to dialogue history. This work constitutes a step toward building high-performing unsupervised dialogue act models that will be used in the next generation of task-oriented dialogue systems.

1 Introduction

Dialogue acts represent the underlying intent of utterances (Austin, 1975; Searle, 1969), and constitute a crucial level of representation for dialogue systems (Sridhar et al., 2009). The task of automatic dialogue act classification has been extensively studied for decades within several domains including train fares and timetables (Allen et al., 1995; Core and Allen, 1997; Crook et al., 2009; Traum, 1999), virtual personal assistants (Chen and Di Eugenio, 2013), conversational telephone speech (Stolcke et al., 2000), Wikipedia talk pages (Ferschke et al., 2012) and as in the case of this paper, tutorial dialogue (Serafin and Di Eugenio, 2004; Forbes-Riley and Litman, 2005; Boyer et al., 2011; Dzikovska et al., 2013).

Most of the prior work on dialogue act classification has depended on manually applying dialogue act tags and then leveraging supervised machine learning (Di Eugenio et al., 2010; Keizer et al., 2002; Reithinger and Klesen, 1997; Serafin and Di Eugenio, 2004). This process involves engineering a dialogue act taxonomy (or using an existing one, though domain-specific phenomena can be difficult to capture within multi-purpose dialogue act taxonomies) and manually annotating each utterance in the corpus. Then, the tagged utterances are provided to a supervised machine learner. This supervised approach can achieve strong performance, in excess of 75% accuracy on manual tags, approaching the agreement level that is sometimes observed between human annotators (Sridhar et al., 2009; Serafin and Di Eugenio, 2004; Chen and Di Eugenio, 2013).

However, the supervised approach has several major drawbacks, including the fact that handcrafting dialogue act tagsets and applying them manually tend to be bottlenecks within the research and design process. To overcome these drawbacks, the field has recently seen growing momentum surrounding unsupervised approaches, which do not require any manual labels during model training (Crook et al., 2009; Joty et al., 2011; Lee et al., 2013). A variety of unsupervised machine learning techniques have been investigated for dialogue act classification, and each line of investigation has explored which features best support this goal. However, to date the best performing unsupervised models achieve in the range of 40% (Rus et al., 2012) to 60% (Joty et al., 2011) training set accuracy on manual tags, substantially lower than the mid-70% accuracy (Sridhar et al., 2009) often achieved on testing sets with supervised models.

Proceedings of the SIGDIAL 2014 Conference, pages 113–122,
Philadelphia, U.S.A., 18-20 June 2014.

In order to close this performance gap between unsupervised and supervised techniques, we suggest that it is crucial to enrich the features available to unsupervised models. In particular, when a dialogue is task-oriented and includes a rich source of information within a parallel task stream, these features may substantially boost the ability of an unsupervised model to distinguish dialogue acts. For example, in situated dialogue, features representing the state of the physical world may be highly influential for dialogue act modeling (Grosz and Sidner, 1986).

Human tutorial dialogue, which is the domain being considered in the current work, often exhibits this structure: the task artifact is external to the dialogue utterances themselves (in the case of our work, this artifact is a computer program that the student is constructing). Task features have already been shown beneficial for supervised dialogue act classification in our domain (Ha et al., 2012). We hypothesize that including these task features within an unsupervised model will significantly improve its performance. In addition, we hypothesize that including dialogue history as a prominent feature within an unsupervised model will provide significant improvement.

This paper represents the first investigation into combining task and dialogue features within an unsupervised dialogue act classification model. First, we discuss representation of these task features and dialogue structure features, and compare these representations within both flat and hierarchical clustering approaches. Second, we report on experiments that demonstrate that the inclusion of task features significantly improves dialogue act classification, and that a hierarchical cluster structure which explicitly captures dialogue history performs best. Finally, we break down the model's performance by dialogue act and investigate which features are most beneficial for distinguishing particular acts. These contributions constitute a step toward building high-performing unsupervised dialogue act models that can be used in the next generation of task-oriented dialogue systems.

2 Related Work

There is a rich body of work on dialogue act classification. Supervised approaches for dialogue act classification aimed at improving performance by using several features such as dialogue structure including position of the turn (Ferschke et al., 2012), speaker of an utterance (Tavafi et al., 2013), previous dialogue acts (Kim et al., 2010), lexical features such as words (Stolcke et al., 2000), syntactic features including part-of-speech tags (Bangalore et al., 2008; Marineau et al., 2000), task-subtask structure (Boyer et al., 2010) acoustic and prosodic cues (Sridhar et al., 2009; Jurafsky et al., 1998), and body posture (Ha et al., 2012).

For the growing body of work in unsupervised dialogue act classification a subset of these features have been utilized. The words (Crook et al., 2009), topic words (Ritter et al., 2010), function words (Ezen-Can and Boyer, 2013b), beginning portions of utterances (Rus et al., 2012), part-of-speech tags and dependency trees (Joty et al., 2011), and state transition probabilities in Markov models (Lee et al., 2013) are among the list of features investigated for unsupervised modeling of dialogue acts. However, the accuracies achieved by the best of these models are well below the accuracies achieved by supervised techniques. To improve performance of unsupervised models for task-oriented dialogue, utilizing a combination of task and dialogue features is a promising direction.

3 Corpus

The task-oriented dialogue corpus used in this work was collected in a computer-mediated human tutorial dialogue study. Students ($n = 42$) and tutors interacted through textual dialogue within an online learning environment for introductory Java programming (Ha et al., 2012). The students were novices, never having programmed in Java previously. The tutorial dialogue interface consisted of four windows, one describing the learning task, another where students wrote programming code, beneath that the output of either compiling or executing the program, and finally the textual dialogue window (Figure 1).

As students and tutors interacted through this interface, all dialogue messages and keystroke-level task events were logged to a database. Only students could compose, compile, and execute the code, so task actions represent student actions while dialogue messages were composed by both participants. The corpus contains six lessons for each student-tutor pair, of which only the first lesson was annotated with dialogue act tags (κ=0.80).

This annotated set contains 5,705 utterances (4,065 tutor and 1,640 student). The average num-

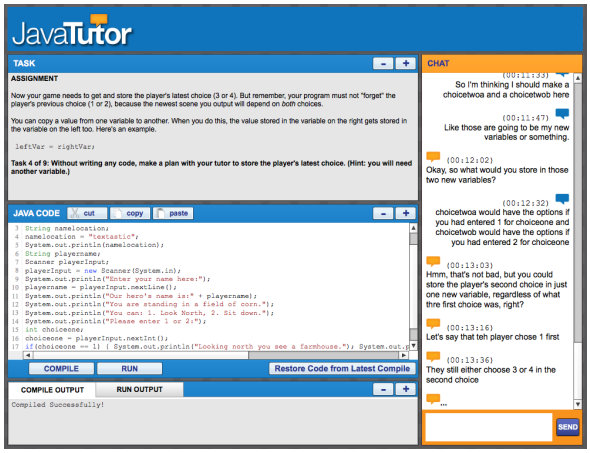

Figure 1: The tutorial dialogue interface with four windows.

ber of utterances (both tutor and student) per tutoring session was 116 (min = 70, max = 211). The average number of tutor utterances per session is 96 (min=44, max=156) whereas for students it is 39 (min=18, max=69) for the annotated set. The average number of words per utterance for students is 4.4 and for tutors it is 5.4. This annotated set is used in the current analysis for both training and testing where cross-validation is applied. As described later, a separate set containing 462 unannotated utterances is used as a development set for determining the number of clusters.

The dialogue stream of this corpus was manually annotated as part of previous work on supervised dialogue act modeling which achieved 69% accuracy with Conditional Random Fields (Ha et al., 2012). A brief description of the student dialogue act tags, which are the focus of the models reported in this paper, is shown in Table 1. The most frequent dialogue act (A) constitutes the baseline chance (39.85%). In the current work, the manually applied dialogue act labels are not utilized during model training, but are only used for evaluation purposes as our models' accuracies are reported for manual tags on a held-out test set.

An excerpt from the corpus is shown in Table 2. Note that the current work focuses on classifying student dialogue act tags, since in an automated dialogue system the tutor moves would be generated by the system and their dialogue acts tags would therefore be known.

4 Features

A key issue for dialogue act classification in task-oriented dialogue involves how to represent dialogue and task events. This section describes how features were extracted from the corpus of human tutorial dialogue.

Student Dialogue Act	Distribution
Answer (A)	39.85
Acknowledgement (ACK)	21.31
Statement (S)	21.20
Question (Q)	15.15
Request for Feedback (RF)	0.98
Clarification (C)	0.79
Other (O)	0.61

Table 1: Student dialogue act tags and their frequencies.

Tutor: ready? [*Q*]
Student: yep [*A*]
Tutor moves on to next task
Student: cool [*S*]
Student compiles and runs the code.
Program output: 'Hello World'
Tutor: excellent [*PF*]
Tutor: add a space to make the output look prettier [*DIR*]
Student: why doesnt it stop on the next line in this case? [*Q*]
Program halts
Tutor: it did [*A*]
Student runs the program successfully.
Tutor: good. [*PF*]

Table 2: Excerpt of dialogue from the corpus and the task action that follows utterances.

We use three sets of features: lexical features, dialogue context features, and task features. The lexical and dialogue context features are extracted from the textual dialogue utterances within the corpus. The task features are extracted from the interaction traces within the computer-mediated learning environment and represent a keystroke-level log of events as students worked toward solving the computer programming problems.

4.1 Lexical Features

Because one of the main goals of our work in the longer term is to perform automatic dialogue act classification in real time, we took as a primary consideration the ability to quickly extract lexical features. The features utilized in the current investigation consist only of word unigrams. In ad-

dition to their ease of extraction, our prior work has shown that addition of part-of-speech tags and and syntax features did not significantly improve the accuracy of supervised dialogue act classifiers in this domain (Boyer et al., 2010), and these features can be time-consuming to extract in real time (Ha et al., 2012).

The choice to use word unigrams rather than higher order n-grams is further facilitated by the fact that our clustering technique leverages the ***longest common sub-sequence (LCS)*** metric to measure distances between utterances. This metric counts shared sub-sequences of not-necessarily contiguous words (Hirschberg, 1975). In this way, the LCS metric provides a flexible way for n-grams and skip-n-grams to be treated as important units within the clustering, while the raw features themselves consist only of word unigrams. (We report on a comparison between LCS and bigrams later in the discussion section.) Utilizing LCS, there exists a distance (1-similarity) value from each utterance to every other utterance.

4.2 Dialogue Context Features

Based on previous work on a similar human tutorial dialogue corpus (Ha et al., 2012), we utilize four features that provide information about the dialogue structure. These features are depicted in Table 3. Note that our goal within this work is to classify *student* dialogue moves, not tutor moves, because in a dialogue system the tutor's moves are system-generated with associated known dialogue acts.

Feature	Description
Utterance position	The relative position of an utterance from the beginning of the dialogue.
Utterance length	The number of tokens in the utterance, including words and punctuation.
Previous author	Author of the previous dialogue message (tutor or student) at the time message sent.
Previous tutor dialogue act	Dialogue act of the previous tutor utterance.

Table 3: Dialogue context features and their descriptions.

4.3 Task Features

As described previously, the corpus contains two channels of information: the dialogue utterances, from which the lexical and dialogue context features were extracted, and in addition, the task stream consisting of student problem-solving activities such as authoring code, compiling, and executing the program. The programming activities of students were logged to a database along with all of the dialogue events during tutoring.

A set of task features was found to be important for dialogue act classification in this domain in prior work, including most recent programming action, status of the most recent task activity and task activity flag representing whether the utterance was preceded by a student's task activity (Ha et al., 2012). We expand this set of features as shown in Table 4.

5 Experiments

The goal of this work is to investigate the impact of including task and dialogue context features on unsupervised dialogue act models. We hypothesize that incorporating task features will significantly improve the performance of an unsupervised model, and we also hypothesize that properly incorporating dialogue context features, which are at a different granularity than the lexical features extracted from utterances, will substantially improve model accuracy.

5.1 Dialogue Act Modeling With k-medoids Clustering

The unsupervised models investigated here use k-medoids clustering, which is a well-known clustering technique that takes actual data points as the center of each cluster (Ng and Han, 1994), in contrast to k-means which may have synthetic points as centroids. In k-medoids, the centroids are initially selected and then the algorithm iterates, reassigning data points in each iteration, until the clusters converge. In standard k-medoids clustering the initial seeds are selected randomly and then a correct distribution of data points is identified through the iteration and convergence process. For dialogue act classification, the influence of the initial seeds is substantial because the frequencies across dialogue tags are typically unbalanced. To overcome this challenge, we use a greedy seed selection approach similar to the one used in k-means++ (Arthur and Vassilvitskii,

Feature	Description
prev_action	Most recent action of the student (composing a dialogue utterance, constructing code, compiling or executing code).
task_begin	Whether the student utterance is the first utterance since the beginning of the subtask.
task_stu	Whether the student utterance was preceded by a task event.
task_prev_tut	Task activity flag indicating whether the closest tutor utterance in this subtask was preceded by a task activity.
task_status	The status of the most recent coding action (*begin*, *stop*, *success*, *error* and *input_sent*).
time_elapsed	Time elapsed between the previous tutor message and the current student utterance.
errors	Number of errors in the student's latest code.
delta_errors	Difference in the number of errors in the task between two utterances in the same dialogue.
stu_#_task	Number of student dialogue messages sent within the current task.
stu_#_dial	Number of student dialogue messages sent within the current dialogue.
tut_#_task	Number of tutor dialogue messages sent within the current subtask.
tut_#_dial	Number of tutor dialogue messages sent within the current dialogue.

Table 4: Task features extracted from student computer programming activities.

2007) which selects the first seed randomly and then greedily chooses seeds that are farthest from the chosen seeds. The goal of using this approach in our application is to choose seeds from different dialogue acts so that the final model achieves good coverage. Our preliminary experiments demonstrated that this greedy seed selection combined with k-medoids outperforms other clustering approaches including those utilized in our prior work (Ezen-Can and Boyer, 2013a).

In order to select the number of clusters k, a subset of the corpus, constituting 25% of the full corpus (that were not tagged) composed of 462 utterances, was separated as a development set. First, we examined the coherence of clusters at different values of k using intra-cluster distances. This technique involves identifying an 'elbow' where the decrease in intra-cluster distance becomes less rapid (since adding more clusters can continue to decrease intra-cluster distance to the point of overfitting) (Figure 2). The graph suggests an elbow at k=5. Because there may be multiple elbows in the intra-cluster distance, a second method utilizing Bayesian Information Criterion (BIC) was used which penalizes models as the number of parameters increases. The lower the BIC value, the better the model is, achieved at k=5 as well.

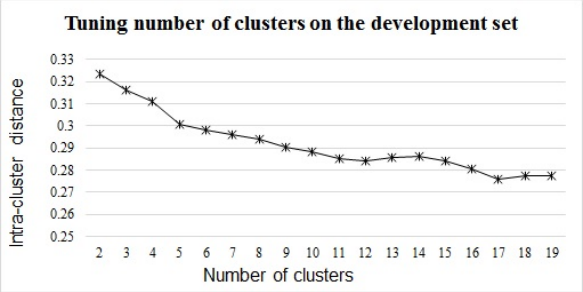

Figure 2: Intra-cluster distances with varying number of clusters.

Unlike many other investigations into unsupervised dialogue act classification, the current approach reports accuracy on held-out test data, not on the data on which the model was trained. Even though the model training process does not utilize available manual tags, requiring the learned unsupervised model to perform well on held-out test data more closely mimics the broader goal of our work which is to utilize these unsupervised models within deployed dialogue systems, where most utterances to be classified have never been encountered by the model before.

The procedure for model training and testing uses leave-one-student-out cross-validation. Rather than other forms of leave-one-out or stratified cross-validation, leave-one-student-out ensures that each student's set of dialogue utterances are treated as the testing set while the model is trained on all other students' utterances. This process is repeated until each student's utterances

have served as a held-out test set (in our case, this results in n=42 folds). Within each fold, the clusters are learned during training and then for each utterance in the test set, its closest cluster is computed by taking the average distance of the test utterance to the elements in the cluster. The majority label of the closest cluster is assigned as the dialogue act tag for the test utterance. If the assigned dialogue act tag matches the manual label of the test utterance, the utterance is counted as correct classification. The average accuracy is computed as the number of correct classifications divided by the total number of classifications.

5.2 Experimental Results

We conducted experiments with seven different feature combinations: L, lexical features only, T, task features only, D, dialogue context features only, and then the combinations of these features, $T + D$, $T + L$, $D + L$, and $T + D + L$. We hypothesized that the addition of task features would significantly improve the models' accuracy. As shown in Table 5, adding task features to dialogue context features significantly outperforms dialogue context features alone ($T + D > D$). Similarly, adding task features to lexical features provides significant improvement ($T + L > L$). However, adding task features to the dialogue context plus lexical features model does not provide benefit, and in fact slightly (not significantly) degrades performance ($T + D + L \not> D + L$). As reflected by the Kappa scores, the test set performance attained by these models is hardly better than would be expected by chance.

	Features	Accuracy (%)	Kappa
Flat Clustering	L	33	0.02
	T	37.7	0.07
	D	37.6	0.07
	T+D	39.1*	0.07
	T+L	38*	0.06
	D+L	38.3	0.07
	T+D+L	37.3	0.05

Table 5: Test set accuracies and Kappa for the flat clustering model (L: Lexical features, D: Dialogue context features, T: Task features) *indicates statistically significant compared to the similar model without task features ($p < 0.05$).

5.3 Utilizing Dialogue History

The importance of dialogue history, particularly the influence of the most recent turn on an upcoming turn, is widely recognized within dialogue research, notably by work on adjacency pairs (Schegloff and Sacks, 1973; Forbes-Riley et al., 2007; Midgley et al., 2009). Based on these findings, we hypothesized that dialogue history would be substantially beneficial for unsupervised dialogue act models as it has been observed to be in numerous studies on supervised classification. However, as seen in the previous section, adding these dialogue context features with equal weight to the model using Cosine distance only improved its performance slightly though statistically significantly (for example, $T+D > T$), while the overall performance is still barely above random chance.

In an attempt to substantially boost the performance of the unsupervised dialogue act classifier, we experimented with a hierarchical clustering structure in which the model first branches on the previous tutor move, and then the clustering models are learned as described previously at the leaves of the tree (Figure 3).

This branching approach results in some branches with too few utterances to train a multi-cluster model. To deal with this situation we set a threshold of n=10 utterances. For those subgroups with fewer than 10 utterances, we take a simple majority vote to classify test cases, and for those subgroups with 10 or larger utterances we train a cluster model and use it to classify test cases. For the entire corpus, the number of utterances in each branch is presented in Table 6.

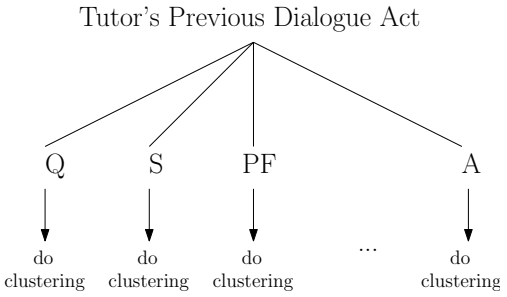

Figure 3: Branching student utterances according to previous tutor dialogue act.

As the results in Table 7 show, the performance of the model with hierarchical structure is significantly better than the flat clustering model. Note that each feature in this table leverages previous

Tutor Dialogue Act	# of student utterances
Q	818
S	464
H	125
PF	91
A	61
ACK	11
C	8
O	8
RACK	6

Table 6: The number of student utterances after branching on the previous tutor dialogue act.

tutor dialogue act while branching. Branching on previous tutor move boosted the model's accuracy for student move dialogue act classification by approximately 30% accuracy across all feature sets, a difference that is statistically significant in every case. With the hierarchical model structure, the best performance is achieved by including all three types of features: lexical, dialogue context and task. However, our hypothesis that task features would significantly improve the accuracy does not hold within the hierarchical clustering model ($T + D \not> D$ and $T + L \not> L$).

	Features	Accuracy (%)	Kappa
Hierarchical	T	64.2†	0.45
	D	63.2†	0.46
	L	60.7†	0.41
	T+D	62.1†	0.44
	T+L	63.3*†	0.45
	D+L	63.6†	0.46
	T+D+L	**65***†	0.48

Table 7: Test set accuracies and Kappa for branching on previous tutor dialogue act (L: Lexical features, D: Dialogue context features, T: Task features) *indicates statistically significant compared to the similar model without task features and † indicates hierarchical clustering performing significantly better than flat with same features. ($p < 0.05$).

6 Discussion

The experimental results provide compelling evidence that an inclusive approach to features for unsupervised dialogue act modeling holds great promise. However, we observed a stark difference in model performance when the tutor's previous move was simply included as one of many features within a flat clustering model compared to when the previous tutor move was treated as a branching feature. In this section we take a closer look and discuss the features that help distinguish particular dialogue acts from each other.

Using the hierarchical $T + D + L$ model which performed best within the experiments, we examine the confusion matrix (Figure 4). Statements and acknowledgments prove challenging for the model, 51.3% and 61.5% accuracy overall. Moreover, these two tags are easily confused with each other: 29.7% of statements were misclassified as acknowledgments, while 21.2% of acknowledgments were misclassified as statements. The worst overall classification accuracy was for questions (6%) and the best was achieved for answers (95.3%).

True \ Predicted	S	A	Q	ACK	C	RF	O
S	181	48	17	105	2	0	0
A	15	645	6	11	0	0	0
Q	83	78	14	58	0	0	0
ACK	74	44	14	206	0	0	0
C	8	4	1	1	0	0	0
RF	6	5	2	2	0	0	0
O	5	5	0	0	0	0	0

Figure 4: Confusion matrix for hierarchical model utilizing all features: T+D+L.

When we analyze the performance of different sets of features with respect to individual dialogue acts, some interesting results emerge. The analysis shows that task features are especially good for classifying statements. Using only task features, the model correctly classified 61.8% statements, compared to the lower 51.3% accuracy that the overall best model ($T + D + L$) achieved on statements. When we consider the nature of the statement dialogue act within this corpus, we note that it is a large category that encompasses a variety of utterances, some of which have lexical features in common with acknowledgments. In this case, task features are particularly helpful.

For acknowledgments, a combination of task and lexical features performed best (63.6% ac-

curacy) compared to the overall best performing model which achieved a slightly lower 61.5% accuracy on acknowledgments. Acknowledgments are another example of an act that may take ambiguous surface form; for example, in our corpus an utterance 'yes' appears as both an answer and an acknowledgment depending on its context. Therefore, higher level features such as the ones provided by task may be more helpful.

For questions, the highest performing feature set is L. However, as shown in Table 8, the model performed poorly on questions. Inspection of the models reveals that questions are varied in terms of structure throughout the corpus and it is hard to distinguish them from other dialogue acts. For instance there are two consequent utterances "i need a write statement" and "don't i", both of which are manually labeled as questions. However, in terms of the structure, the first utterance looks very similar to a statement and therefore the model has difficulty grouping it with questions. Due to the large variety of question forms in the corpus, it is possible that the clustering performed poorly on this dialogue act. In future work it will be promising to investigate the dialogue structures which produce questions and to weight them more in the feature set in order to increase performance of clustering for questions.

We performed one additional experiment to compare the performance of the LCS metric with bigrams. For bigrams, the average leave-one-student-out test accuracy was 25% with flat clustering compared to the lexical-only case using LCS (L) which reached 33%.

Features	S	A	Q	ACK
L	21.5	41.3	**14.2**	20.4
T	**61.76**	95.27	7.30	40.90
D	48.16	95.27	3.00	60.30
T+D	52.69	94.68	3.43	51.64
T+L	42.78	95.13	6.01	**63.58**
D+L	43.63	94.98	8.58	62.09
T+D+L	51.27	95.27	6.01	61.49

Table 8: Accuracies for individual dialogue acts. Acts with fewer than 10 utterances after branching are omitted from the table.

7 Conclusion and Future Work

Dialogue act classification is crucial for dialogue management, and unsupervised modeling approaches hold great promise for automatically extracting classification models from corpora. This paper has focused on unsupervised dialogue act classification for task-oriented dialogue, investigating the impact of task features and dialogue context features on model accuracy within both flat and hierarchical clusterings. Experimental results confirm that utilizing a combination of task and dialogue features improves accuracy and that incorporating one previous tutor move as a high-level branching feature a provides particularly marked benefit. Moreover, it was found that task features are particularly important for identifying particular dialogue moves such as statements, for which the model with task features only outperformed the model with all features.

In addition to the task stream, future work should consider other sources of nonverbal cues such as posture, gesture and facial expressions to investigate the extent to which these can be successfully incorporated in unsupervised dialogue act models. Second, models that are built in specialized ways to different user groups (e.g., by gender or by incoming skill level) should be investigated. Finally, the performance of unsupervised dialogue act classification models must ultimately move toward evaluation within implemented dialogue systems (Ezen-Can and Boyer, 2013a). The overarching goal of these investigations is to create unsupervised dialogue act models that perform well enough to be used within deployed dialogue systems and enable the system to respond successfully. It is hoped that in the future, dialogue act classification models for many domains can be extracted automatically from corpora of human dialogue in those domains without the need for any manual annotation.

Acknowledgments

Thanks to the members of the LearnDialogue group at North Carolina State University for their helpful input. This work is supported in part by the National Science Foundation through Grant DRL-1007962 and the STARS Alliance, CNS-1042468. 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

James F. Allen, Lenhart K. Schubert, George Ferguson, Peter Heeman, Chung Hee Hwang, Tsuneaki Kato, Marc Light, Nathaniel Martin, Bradford Miller, Massimo Poesio, et al. 1995. The TRAINS project: A case study in building a conversational planning agent. *Journal of Experimental & Theoretical Artificial Intelligence*, 7(1):7–48.

David Arthur and Sergei Vassilvitskii. 2007. k-means++: The advantages of careful seeding. In *Proceedings of the 18th ACM-SIAM Symposium on Discrete Algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics.

John Langshaw Austin. 1975. *How To Do Things with Words*, volume 1955. Oxford University Press.

Srinivas Bangalore, Giuseppe Di Fabbrizio, and Amanda Stent. 2008. Learning the structure of task-driven human–human dialogs. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(7):1249–1259.

Kristy Elizabeth Boyer, Eun Young Ha, Robert Phillips, Michael D. Wallis, Mladen A. Vouk, and James C. Lester. 2010. Dialogue act modeling in a complex task-oriented domain. In *Proceedings of SIGDIAL*, pages 297–305. Association for Computational Linguistics.

Kristy Elizabeth Boyer, Eun Young Ha, Robert Phillips, and James Lester. 2011. The impact of task-oriented feature sets on HMMs for dialogue modeling. In *Proceedings of SIGDIAL*, pages 49–58. Association for Computational Linguistics.

Lin Chen and Barbara Di Eugenio. 2013. Multimodality and dialogue act classification in the RoboHelper project. In *Proceedings of SIGDIAL*, pages 183–192.

Mark G. Core and James Allen. 1997. Coding dialogs with the DAMSL annotation scheme. In *Proceedings of the AAAI Fall Symposium on Communicative Action in Humans and Machines*, pages 28–35.

Nigel Crook, Ramon Granell, and Stephen Pulman. 2009. Unsupervised classification of dialogue acts using a Dirichlet process mixture model. In *Proceedings of SIGDIAL*, pages 341–348. Association for Computational Linguistics.

Barbara Di Eugenio, Zhuli Xie, and Riccardo Serafin. 2010. Dialogue act classification, higher order dialogue structure, and instance-based learning. *Dialogue & Discourse*, 1(2):1–24.

Myroslava O. Dzikovska, Elaine Farrow, and Johanna D. Moore. 2013. Combining semantic interpretation and statistical classification for improved explanation processing in a tutorial dialogue system. In *Artificial Intelligence in Education*, pages 279–288.

Aysu Ezen-Can and Kristy Elizabeth Boyer. 2013a. In-context evaluation of unsupervised dialogue act models for tutorial dialogue. In *Proceedings of SIGDIAL*, pages 324–328.

Aysu Ezen-Can and Kristy Elizabeth Boyer. 2013b. Unsupervised classification of student dialogue acts with query-likelihood clustering. In *International Conference on Educational Data Mining*, pages 20–27.

Oliver Ferschke, Iryna Gurevych, and Yevgen Chebotar. 2012. Behind the article: Recognizing dialog acts in Wikipedia talk pages. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 777–786.

Kate Forbes-Riley and Diane J. Litman. 2005. Using bigrams to identify relationships between student certainness states and tutor responses in a spoken dialogue corpus. In *Proceedings of the SIGDIAL Workshop*, pages 87–96.

Kate Forbes-Riley, Mihai Rotaru, Diane J. Litman, and Joel Tetreault. 2007. Exploring affect-context dependencies for adaptive system development. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, pages 41–44.

Barbara J. Grosz and Candace L. Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Eun Young Ha, Joseph F. Grafsgaard, Christopher M. Mitchell, Kristy Elizabeth Boyer, and James C. Lester. 2012. Combining verbal and nonverbal features to overcome the 'information gap' in task-oriented dialogue. In *Proceedings of SIGDIAL*, pages 247–256.

Daniel S. Hirschberg. 1975. A linear space algorithm for computing maximal common subsequences. *Communications of the ACM*, 18(6):341–343.

Shafiq Joty, Giuseppe Carenini, and Chin-Yew Lin. 2011. Unsupervised modeling of dialog acts in asynchronous conversations. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 1807–1813.

Daniel Jurafsky, Elizabeth Shriberg, Barbara Fox, and Traci Curl. 1998. Lexical, prosodic, and syntactic cues for dialog acts. In *Proceedings of the ACL/COLING-98 Workshop on Discourse Relations and Discourse Markers*, pages 114–120.

Simon Keizer, Rieks op den Akker, and Anton Nijholt. 2002. Dialogue act recognition with Bayesian networks for Dutch dialogues. In *Proceedings of the SIGDIAL Workshop*, pages 88–94.

Su Nam Kim, Lawrence Cavedon, and Timothy Baldwin. 2010. Classifying dialogue acts in one-on-one live chats. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 862–871.

Donghyeon Lee, Minwoo Jeong, Kyungduk Kim, and Seonghan Ryu. 2013. Unsupervised spoken language understanding for a multi-domain dialog system. *IEEE Transactions On Audio, Speech, and Language Processing*, 21(11):2451–2464.

Johanna Marineau, Peter Wiemer-Hastings, Derek Harter, Brent Olde, Patrick Chipman, Ashish Karnavat, Victoria Pomeroy, Sonya Rajan, Art Graesser, Tutoring Research Group, et al. 2000. Classification of speech acts in tutorial dialog. In *Proceedings of the Workshop on Modeling Human Teaching Tactics and Strategies at the Intelligent Tutoring Systems Conference*, pages 65–71.

T. Daniel Midgley, Shelly Harrison, and Cara MacNish. 2009. Empirical verification of adjacency pairs using dialogue segmentation. In *Proceedings of SIGDIAL*, pages 104–108.

Raymond Ng and Jiawei Han. 1994. Efficient and effective clustering methods for spatial data mining. In *Proceedings of the 20th International Conference on Very Large Data Bases*, pages 144–155.

Norbert Reithinger and Martin Klesen. 1997. Dialogue act classification using language models. In *Proceedings of EuroSpeech*, pages 2235–2238.

Alan Ritter, Colin Cherry, and Bill Dolan. 2010. Unsupervised modeling of twitter conversations. In *Proceedings of the Association for Computational Linguistics*, pages 172–180.

Vasile Rus, Cristian Moldovan, Nobal Niraula, and Arthur C. Graesser. 2012. Automated discovery of speech act categories in educational games. In *International Conference on Educational Data Mining*, pages 25–32.

Emanuel A. Schegloff and Harvey Sacks. 1973. Opening up closings. *Semiotica*, 8(4):289–327.

John R. Searle. 1969. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press.

Riccardo Serafin and Barbara Di Eugenio. 2004. FLSA: Extending latent semantic analysis with features for dialogue act classification. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 692–699. Association for Computational Linguistics.

Rangarajan Sridhar, Vivek Kumar, Srinivas Bangalore, and Shrikanth Narayanan. 2009. Combining lexical, syntactic and prosodic cues for improved online dialog act tagging. *Computer Speech & Language*, 23(4):407–422.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–373.

Maryam Tavafi, Yashar Mehdad, Shafiq Joty, Giuseppe Carenini, and Raymond Ng. 2013. Dialogue act recognition in synchronous and asynchronous conversations. In *Proceedings of SIGDIAL*, pages 117–121.

David R. Traum. 1999. Speech acts for dialogue agents. In *Foundations of Rational Agency*, pages 169–201. Springer.