Automated Speech Act Categorization of Chat Utterances in Virtual Internships

Dipesh Gautam
The University of Memphis
Memphis, TN 38152
dgautam@memphis.edu

Nabin Maharjan
The University of Memphis
Memphis, TN 38152
nmharjan@memphis.edu

Arthur C. Graesser
The University of Memphis
Memphis, TN 38152
graesser@memphis.edu

Vasile Rus
The University of Memphis
Memphis, TN 38152
vrus@memphis.edu

ABSTRACT

This work is a step towards full automation of auto-mentoring processes in multi-player online environments such as virtual internships. We focus on automatically identifying speaker's intentions, i.e. the speech acts of chat utterances, in such virtual internships. Particularly, we explore several machine learning methods to categorize speech acts, with promising results. A novel approach based on pre-training a neural network on a large set of (and noisy) labeled data and then on expert-labeled data led to best results. The proposed methods can help understand patterns of conversations among players in virtual internships which in turn could inform refinements of the design of such learning environments and ultimately the development of virtual mentors that would be able to monitor and scaffold students' learning, i.e., the acquisition of specific professional skills in this case.

Keywords

speech act, virtual internships, online tutoring, classification, neural networks, machine learning

1. INTRODUCTION

Virtual internships are simulations where interns gain professional experience while participating in an online fictional company. That is, they go through an internship experience without actually being present in a physical, actual company. In such virtual internships, the student interns participate in activities such as solving designated problems or tasks for which they actively interact with their mentor(s) as well as other interns through instant text messages, voice messages, chatrooms, and multimedia elements. The learning that occurs in engineering virtual internships, our focus, can be characterized by epistemic frame theory. This theory claims that professionals develop epistemic frames, or the network of skills, knowledge, identity, values, and epistemology that are unique to that profession [17]. For example, engineers share ways of understanding and doing (knowledge and skills); beliefs about which problems are worth investigating (values), characteristics that define them as members of the profession (identity), and a ways of justifying decisions (epistemology).

It is important to understand patterns of conversations between the various players in a virtual internship in order to refine the design of such virtual internships and to ultimately develop a virtual mentor that would be able to monitor and scaffold students' learning, i.e., the acquisition of specific professional skills in this case. Currently, virtual internship environments rely on human mentors. Our work here is a step towards a deeper understanding and full automation of the mentoring process. Indeed, understanding the mentoring process implies detecting patterns of actions by the mentor and by the students that are effective. Since conversations are the main type of interactions between the mentors and the student interns, understanding the actions or intents behind each utterance in the conversations is critical. We offer here such solutions to automatically detecting the intent, or speech act, behind chat utterances in virtual internships. Furthermore, such solution are critical to fully automate the mentoring process, i.e., to building automentors. Indeed, knowing students' speech acts can inform an automated mentoring agent to plan the best reply. For instance, if a student is greeting, the system should respond with a greeting or if a student is asking a question the system should plan to, for instance, answer the question.

Speech acts are a construct in linguistics and the philosophy of language that refers to the way natural language performs actions in human-to-human language interactions, such as dialogues. Speech act theory was developed based on the "language as action" assumption. The basic idea is that behind every utterance there is an underlying speaker intent, called the speech act. For instance, the utterance "Hello, John!" corresponds to a greeting, that is, the speaker's intention is to greet, whereas the utterance "Which web browser are you using?" is about asking a question. As already hinted earlier, discovering learners' patterns of actions in the form of patterns of (speech) acts in virtual internships could be revealing. For instance, we may find that interns that ask more

questions acquire better and faster target professional skills based on the theory that asking more relevant questions indicates a more active and engaged learner which typically leads to more effective and efficient learning processes.

Labeling utterances with speech acts requires both an analysis of the utterance itself, e.g., "Hello" clearly indicates a greeting, but also accounting for the previous context, i.e., previous utterances in the conversation. For instance, after a question, a response most likely follows. This pattern holds in dialogues, i.e., interactions between two conversational partners where there is a clear pattern of turn-taking; that is, a speaker's turn is followed by a turn by the other speaker. However, in multi-player conversations such as the one that we deal with in this work, identifying the previous utterance that is most relevant to the current one is more difficult. For example, in the snapshot of conversation shown in Table 1 from one of our virtual internships, the question in chat utterance 3 from *player2* is addressed to the *mentor* whose reply is in utterance 6. The next *Player2's* reply is in utterance 9. Indeed, in such multi-party conversations, it becomes more challenging to link a target utterance to the previous one that triggered it. The complexity of untangling such multi-player conversations is further increased as the number of participants increases. Therefore, even though the speech act of an utterance is determined to some degree by the previous, related chat utterances, in this work we explore a method for speech act classification that relies only on the content of the target utterance itself, ignoring the previous context.

Table 1: A Snapshot of Conversation in Nephrotex

S.N.	Speaker	Utterance
1	mentor	I'm here to help you.
2	player1	hi!
3	player2	*Has anyone been able to get the tutorial notebook to open?*
4	player3	Hey
5	player4	Hello!
6	mentor	*Which web browser are you using?*
7	player3	are you guys real?
8	player1	yes we're real lol
9	player2	*I switched to Firefox, now everything is working. Thanks!*

To this end, we used various existing classifiers such as Naive Bayes and decision trees along with a Neural Network (NN) approach. Based on previous experience such as [15, 14], we selected leading words in each utterrance as the features of the underlying model. The feature-based representations of utterances were then fed into Naive Bayes and decision tree classifiers. For neural networks, we used the pre-trained sent2vec[11] model, trained on a large collection of Wikipedia articles, to map an entire utterance onto a vector representation or embedding. Nevertheless, our data is dialogue data which differs from Wikipedia texts to some degree. To compensate for this discrepancy, the basic model is used to further train a small neural network using a comparatively small domain specific dataset in order to improve the predictive power for the type of instances seen in our dataset. That is, this is a form of transfer learning where our model first uses generic knowledge from the pre-trained Wikipedia model which is then tranferred or adapted to a specific domain data by training with domain data. Furthermore, using pre-trained models can also lead to better parameter learning in NN [12].

We also investigated a novel approach to building a speech act classifier for multi-player conversational systems using a mix of noisy and golden data, as explained next. In this approach, we trained a decision tree model with a small set of human annotated data and then used that trained model to generate (noisy) labels for a much larger collection of utterances. The noisy labeled utterances were then used to pre-train the neural network and then further trained with the human annotated gold dataset. The advantage of pre-training here is to have a huge collection of training data to pre-train the network and then refine the training using the (smaller) human-annotated (noise-free or gold) dataset.

Next, we present a quick overview of related work in this area before presenting details of our methods and experiments and resuts.

2. BACKGROUND

As mentioned, our approach to label utterances with speech acts is based on the speech act theory according to which when we say something we do something [1, 16]. Austin theorized the acts performed by natural language utterances. Later on, Searle[16] refined Austin's idea of speech acts by emphasizing the psychological interpretation based on beliefs or intentions. According to Searle, there are three levels of actions carried by language in parallel. First, there is the locutionary act which consists of the actual utterance and its exterior meaning. Second, there is the illocutionary act, which is the real intended meaning of the utterance, its semantic force. Third, there is the perlocutionary act which is the practical effect of the utterance, such as persuading and encouraging. In a few words, the locutionary act is the act of saying something, the illocutionary act is an act performed in saying something, and the perlocutionary act is an actperformed by saying something. For example, the phrase "Don't go into the water" might be interpreted at the three act levels in the following way: the locutionary level is the utterance itself, the morphologically and syntactically correct usage of a sequence of words; the illocutionary level is the act of warning about the possible dangers of going into the water; finally, the perlocutionary level is the actual persuasion, if any, performed on the hearers of the message, to not go into the water.

Many researchers have explored the task of automatically classifying speech acts as well as the related task of discovering speech acts. For instance, Rus and colleagues [14] proposed a method to automatically discover speech act categories in dialogues by clustering utterances spoken by participants in educational games. In our case, we use a predefined taxonomy of speech acts which was inspired by Rus and colleagues' work and further refined by dialogue experts.

The same group of researchers explored the role of Hidden Markov Models (HMMs), a generative model, and Conditional Random Fields (CRFs), a discriminative model, in classifying speech acts in one to one human tutorial sessions [13]. They demonstrated that the CRF model with features constructed from the first three tokens and last token

of previous, next and current utterances, length of current utterance, and other surface features such as bigrams and the speech acts of context utterances performed better than HMM models. They have not worked with multi-party conversations as it is the case in our work.

In other work, Moldovan and colleagues [9] applied supervised machine learning methods to automatically classify chats in an online chat corpus. The corpus consisted of online chat sessions in English between speakers of different ages. Their supervised approach relied on an expert defined set of speech act categories. In their work, they hypothesized that the first few tokens were good predictors of chat's speech act. Samei et al. [15] adopted Moldovan's hypothesis about the predictive power of first few tokens and extended the supervised machine learning model with contextual information, i.e., previous and following utterances. From their experiments with data from an online collaborative learning game, they found that the role of context is minor and therefore context is not that important and can mostly be ignored in predicting speech acts. Similar to those works, we also explore the effectiveness of leading word tokens in utterances for Naive Bayes and decision tree based classifiers.

Ezen and Boyer [4] proposed an unsupervised method for dialogue act classification. They used a corpus from a collaborative learning programming tutor project which consisted of dialogues between pairs of tutors and students collaborating on the task of solving a programming problem. They applied an information retrieval approach in which the target utterance was considered as a query and the rest of the utterances were considered as documents. Based on the ranked list of relevant utterances to the query utterance, a vector representation is derived for each query utterance. The vector representation is then fed into a k-means clustering algorithm to identify clusters of utterances. For evaluation purposes, they used manually labeled data. Each cluster was assigned the majority human-generated label of all utterances in the cluster. An utterance that was placed in a particular cluster by the k-means clustering algorithm was assigned the label of that cluster as its speech act category for evaluation purposes. It should be noted that they varied the number of clusters to obtain a maximum overall accuracy of the discovered labels. Their algorithm outperformed a previous approach for dialogue act clustering, which Ezen and Boyer used for classification and which relied on a simple tf-idf representation and cosine similarity for clustering.

Kim and colleagues investigated the task of classifying dialogue acts in multi-party chats[8]. They analyzed two different types of live chats: (i) live forum chats with multiple participants from the US Library of Congress and (ii) Naval Postgraduate School (NPS) casual chats [5]. In order to classify the utterances in the chats in various speech act categories, Kim and colleagues [7] used speech act patterns which they defined manually using cue words derived from the utterances. They classified the discussion contributions into six speech act categories. They found that the previous chat utterances used as context did not contribute significantly to predicting speech acts in multi-party conversations until the entanglement amongst the utterances was resolved. Our work is similar to theirs in the sense that we analyze multi-party conversations. Nevertheless, our work is conducted in the context of the virtual internship Nephrotex, where learners focus on specific design problems as opposed to the types of conversations used by Kim and colleagues such as the casual NPS chats, which did not focus on a particular given task. We do not explore the accuracy of our methods in context. Furthermore, we do not resolve the entangled dialogues and then use contextual information for speech act classification. We do plan to address the role of context and entanglement in multi-party conversations in future work.

A regular expression based speech act classifier was proposed by Olney et al[10]. Their classifier used regular expression which they called a finite state transducer to classify utterances of AutoTutor, an intelligent tutoring system. They showed that the classifier constructed by cascading parts of speech information, the finite state transducer, and word sense disambiguation rules yielded good performance in classifying utterances into 18 categories. We have not compared our work with a regular expression based classifier due to the labor intensive aspects of such an approach. Typically, such regular-expression approaches should lead to high-precision results and not generalize very well unless they target speech act categories which are more or less closed-class such as greeting expressions (there is a limited number of expressions in which someone can greet).

3. ENGINEERING VIRTUAL INTERNSHIPS

Our work presented here was conducted on conversations among students and mentors in *Nephrotex* (NTX), a virtual internship. Nephrotex was designed and created to improve engineering undergraduate students' professional skills. It was incorporated into first-year engineering undergraduate courses at the University of Wisconsin-Madison[3].

In NTX, groups of students work together on a design problem, e.g. designing filtration membranes for hemodialysis machines, with the help of a mentor. Working on a design problem involves choosing design specifications from a set of input categories. Each student is assigned to a team of five members. There were five such teams who were each expected to learn about one of five different materials.

After completing a set of preliminary tasks, students design five prototypes to submit for testing. Later, they receive performance results for these prototypes which they have to analyze and interpret. Overall, students in each internship complete two such cycles of designing, testing, and analysis before deciding on a final design to recommend. During these cycles, students hold team meetings via the virtual internship's chat interface in which they reflect on their design process and make decisions on how to move forward. Once teams recommend a final design, they present this design to their peers. The conversations among the participants take place virtually via an online chat interface in Nephrotex, or in person outside of the class.

As previously mentioned, in this work, we focus on analyzing chat utterances in Nephrotex in order to discover the underlying speech act. Automated speech acts classification could have significant impact on scaling virtual internships

to all students, anytime, anywhere via Internet-connected devices. This is not currently possible because the human mentors can only handle that much.

4. METHODS

Our approach to classifying learner utterances in virtual internships relies on machine learning algorithms that take as input utterances represented in a feature space. The features in our case are either surface features (such as leading words) or latent features (such as dimensions in neural sentence embeddings). We developed and compared the performance of two different categories of classifiers that rely on these two types of representations. We describe briefly those classifiers, the features we used, and the results obtained during experiments meant to validate the proposed classifiers.

4.1 Classifier Using Surface Features

The surface feature representation of a text uses a number of important lexical and syntactic elements such as leading words or the punctuation mark at the end of the utterance, e.g., the ending question mark at the end of a question. In conversation data such as chat utterances in virtual internship, lexical features such as leading words alone have competitive power in terms of speech act representation of the utterance. Therefore we adopted the model representation proposed previously [9, 14] due to its solid theoretical foundations and competitive results. The basis of this approach is that humans infer speakers' intention after hearing only few of the leading words of an utterance. One argument in favor of this assumption is the evidence that hearers start responding immediately (within milliseconds) or sometimes before speakers finish their utterances[6]. Accordingly, we selected few leading words (first few words) of the utterance as the features to represent the utterance. Although we have experimented with different number of leading words, we report here results with the six leading words (first six words) as this combination yielded best performance as explained later. Once each utterance was mapped onto such a feature-representation, we performed experiments with two different types of classifiers: naive Bayes and decision trees.

Before feature construction, we pre-processed the utterances by lemmatizing the words and removed the punctuations. Although some of the punctuations, such as "*question mark (?)*" or "*exclamation mark (!)*", are predictive on some of the speech acts, they seem to not always be present in or seem to appear at improper places in the utterance. Hence we ignored the punctuations for our analysis here.

4.2 Classifier Using Latent Features

The other category of classifiers we used relies on latent features that were automatically learned using neural networks. These features are the components of automatically generated vectors that represent sentences. Such neural network generated vectors are derived from textual units such as character, letter n-grams, words and words n-grams. In our model, we adopted sent2vec, a sentence representation model proposed by Pagliardini and colleagues [2, 11] and which was developed by training a neural network on a collection of Wikipedia articles.

Based on such latent representations of utterances, we designed a neural network model in two stages. First, the model obtained a latent representation for an utterance using the generic pre-trained sent2vec model. In a second stage, the embedded vector representation is used to further train our neural network to perform speech act classification.

Table 2: Speech Act Taxonomy with Examples

Speech Acts	Examples
expressive evaluation (xpe)	–It is excellent in all values except for cost –great –The lag is pretty bad
greeting (gre)	–Welcome back interns ! –Hello Team !
metastatements (mst)	–sorry littles confused here –Whoops , I was reading that wrong . –lol
other (oth)	–or addition –etc
question (que)	–Is biocompatability cummulative ? –who is going to write the email ?
reaction (rea)	–I 'm ok with this –alright , i think i agree with u guys
request (req)	–Please keep that in mind during your team selection of membrane prototypes . –K , I would like to start the team meeting now .
statement (stm)	–I read an article that said most dialyzers take 6 hours to run . –I can start the meeting with jamon ...

While training the neural network with domain specific data, we applied two methods of training. In the first method, we used a small set of human annotated gold data for training and validation. In the second method, we pre-trained the neural network with noisy labeled data generated from a domain corpus and then further trained and validated the model with gold data. We will discuss in detail the process of generating noisy labels in the next section.

5. EXPERIMENTS AND RESULTS

In this section, we present the experiments that were conducted and the results obtained, starting with a brief description of the data we used.

Table 3: Distribution of Speech Acts in Corpus

	Human Labeled		Noisy Labeled	
Speech act	#	%Dist	#	%Dist
expressive evaluation	24	2.4	256	1.26
greeting	14	1.4	285	1.40
metastatements	40	4.0	405	2.00
other	11	1.1	166	0.82
question	173	17.3	3098	15.25
reaction	202	20.2	3347	16.47
request	56	5.6	1041	5.12
statement	480	48.0	11719	57.68
Total	1000		20317	

5.1 The Virtual Internship Conversation Dataset

Our dataset consists of a collection of more than 22 thousands utterances from the Nephrotex virtual internship. The eight categories of speech acts we used are presented in Table 2 (acronyms are shown in parentheses) together with example utterances.

From the examples, it could be observed that the leading tokens in each utterance are indicative of the underlying speech act shown in the first column. For instance, *greetings* start with "Hello" and "Welcome back" whereas *questions* start with wh-words ("Who") or auxiliary verbs ("Is") while requests start with "Please", which is typically used to ask for something in a nice manner.

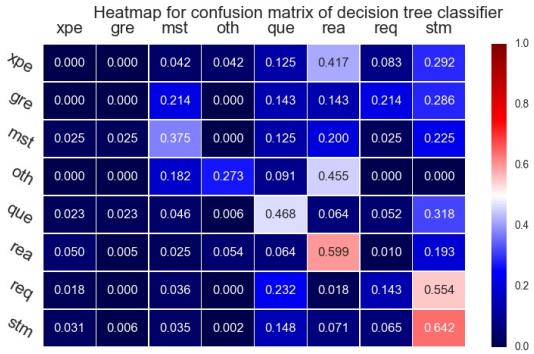

Figure 1: Confusion matrix for classification of decision tree (values refer to percentage expressed in decimal, acronyms refer to the speech acts defined in table 2)

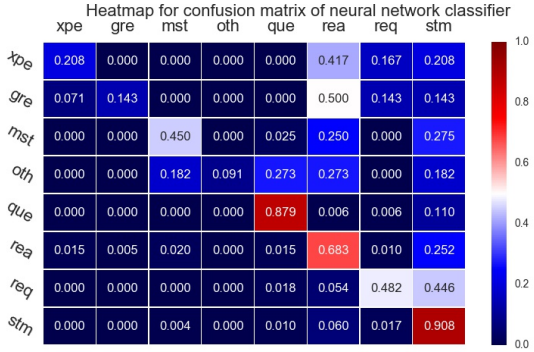

Figure 2: Confusion matrix for classification of neural network (values refer to percentage expressed in decimal, acronyms refer to the speech acts defined in table 2)

5.1.1 The Data Annotation Process

Of the 22,317 utterances, 2,000 utterances were manually annotated by three annotators. Out of these 2,000 utterances, 1,000 utterances were used for training the annotators. Agreement among annotators was computed as the average of Cohen's kappa between all possible pairs of annotators. The average agreement between any two annotators was 0.64.

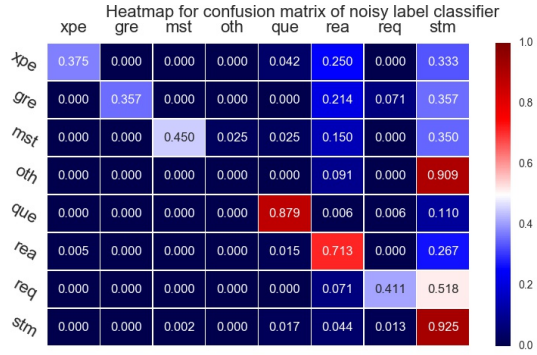

Figure 3: Confusion matrix for classification of noise label trained neural network (values refer to percentage expressed in decimal, acronyms refer to the speech acts defined in table 2)

The remaining 1,000 utterances were labeled by the annotators after finishing their training. The average agreement, measured as Cohen's kappa, among the coders was 0.69. To generate a final, unique label for each annotated utterance in cases in which there were any disagreements, a discussion among the annotators took place as well as the group of co-workers in the project team. We used the 1,000 human-labeled utterances as a gold dataset on which a 10-fold cross validation evaluation methodology was applied to evaluate the proposed speech act classification methods.

5.1.2 The Noisy Label Generation

The rest of the utterances in the whole dataset of 22,317 utterances was automatically labeled using the decision tree model trained on the first 1,000 instances labeled by trainee annotators. We chose decision trees to generate noisy labels because decision trees performed better than the Naive Bayes classifier. It should be noted that we used the other 1000 human-labeled gold data for 10 folds cross validations of our classifier models. Table 3 shows the distribution of speech acts in the gold and noisy labeled datasets. From the table, we observe that the noisy labels generated follow roughly comparable pattern of distribution for the speech acts that are more frequent in corpus. Therefore it makes sense to some extent to use those noisy labels to pre-train the neural network model.

5.2 Results

The results of the 10-fold cross-validation evaluation are summarized in Table 4 and Table 5. We report performance in terms of precision, recall, F-1 score, accuracy, and kappa. The data in Table 4 suggests that the performance of the neural network classifier is highest of all with an average F-1 score and accuracy of 0.764 and 0.779, respectively, and kappa of 0.666, which are the highest among all three types of classifiers including Naive Bayes and decision trees. Moreover, the two sample t-test on 10-fold cross validation accuracies revealed that, neural network performed significantly better than Naive Bayes ($p-value \approx 0.00$) and decision tree with ($p-value \approx 0.00$).

The results shown in Table 5 shows that the neural network model pre-trained with noisy labels improved the per-

Table 4: Performance of Naive Bayes, Decision Tree and Neural Network Classifiers

	NB			DT			NN		
Speech Act	P	R	F1	P	R	F1	P	R	F1
expressive evaluation	0.200	0.042	0.069	0.000	0.000	0.000	0.556	0.208	0.303
greeting	1.000	0.143	0.250	0.000	0.000	0.000	0.667	0.143	0.235
metastatements	0.000	0.000	0.000	0.283	0.375	0.323	0.692	0.450	0.545
other	0.099	1.000	0.180	0.176	0.273	0.214	1.000	0.091	0.167
question	0.000	0.000	0.000	0.429	0.468	0.448	0.921	0.879	0.899
reaction	0.354	0.342	0.348	0.630	0.599	0.614	0.687	0.683	0.685
request	0.000	0.000	0.000	0.143	0.143	0.143	0.614	0.482	0.540
statement	0.581	0.831	0.684	0.680	0.642	0.660	0.791	0.908	0.846
Weighted Average	0.370	0.482	0.406	0.549	0.536	0.542	**0.774**	**0.779**	**0.764**
	Accuracy = 0.482			Accuracy = 0.536			Accuracy = **0.779**		
	Kappa = 0.177			Kappa = 0.341			Kappa = 0.666		

Table 5: Performance of Noise Label Trained Neural Network Classifier

Speech Act	P	R	F1
expressive evaluation	0.900	0.375	0.529
greeting	1.000	0.357	0.526
metastatements	0.947	0.450	0.610
other	0.000	0.000	0.000
question	0.921	0.879	0.899
reaction	0.774	0.713	0.742
request	0.742	0.411	0.529
statement	0.762	0.925	0.835
Weighted Average	**0.796**	**0.795**	**0.781**
Accuracy	**0.795**		
Kappa	0.685		

formance. The overall improvement in precision, recall, F-1 score, and accuracy is about 2% with about 2% better kappa when compared to the neural network classifier (Table 4) without using the much larger, noisy label dataset. However, a t-test showed that the accuracy of the noisy label trained neural network is not significantly better than neural network trained without noisy label data ($p-value \approx 0.53$). This could have happened because of the small samples used for the t-test: 10 from 10-folds cross validations. Using a larger number of folds, say, 50, could help us getting a large sample of accuracy values. It can be observed from the table that the performance for the "*other*" category is the weakest among all four classifiers. The reason is because of the nature of those utterances which contain only a few tokens, i.e., one or two words (see Table 2), with a lot of variation in terms of lexical content. In addition, the human labeled dataset contained few instances for this category which resulted in poor performance when the neural network model was trained using the human labeled data. Similarly, in the noisy, automatically-labeled dataset there are many misclassified "*other*" instances which led to poor training of the neural network model. Furthermore, the next phase of training the pre-trained neural network model with the human labeled data did not compensate enough because there were not sufficient "other" instances in the human labeled data to correct the pre-trained model. This is further supported by analyzing the confusion matrix where the number of true positives for the "*other*" category is 0%; the "*other*" category is labeled as "*statement*" 90% of the time in the case of the neural network model pre-trained with noisy labels (see Figure 3). Further evidence for this is provided by analyzing the confusion matrix for neural network trained only with gold labels where true positives for "*other*" utterances was 9% (see Figure 2). In this case, "other" utterances were labeled as "*question*" and "*reaction*". Other challenging speech acts are "*request*", which is most often confused with "*statement*". This is not surprising as the lexical composition of requests and statements is similar to some degree.

For decision trees, a quick analysis of the confusion matrix (see Figure 1) revealed that the true positives for "*expressive evaluation*" was 0%, being confused mostly with "*reaction*" or "*statement*" (41% and 29% of the time, respectively). Also, "*greeting*" is confused with "*metastatement*" by 21%, "*request*" by 21%, and "*statement* by 28%".

6. CONCLUSIONS

In this work, we explored several methods for speech act classification. We explored various classifier models with different categories of features as well as training strategies. We found that the latent features generated by a pre-trained sentence embeddings model (derived from a large Wikipedia corpus) yielded better performance compared to the other models. Besides that, the predictive power of the neural network model was further boosted when pre-trained with noisy label before training with expert-annotated data.

In future work, we plan to expand the current models by using more contextual information. Given the multi-party nature of our conversation data, before we can use contextual information, it is necessary to disentangle the conversations into sets of related utterances. Our future models will disentagle the multi-party conversations before attempting to use contextual information for speech act classification.

7. ACKNOWLEDGMENTS

This work was funded in part by the National Science Foundation (DRL-0918409, DRL-0946372, DRL-1247262, DRL-1418288, DRL-1661036, DRL-1713110, DUE-0919347, DUE-1225885, EEC-1232656, EEC-1340402, REC-0347000), the MacArthur Foundation, the Spencer Foundation, the Wisconsin Alumni Research Foundation, and the Office of the Vice Chancellor for Research and Graduate Education at the University of Wisconsin-Madison. The opinions, findings, and conclusions do not reflect the views of the funding agencies, cooperating institutions, or other individuals.

8. REFERENCES

[1] Austin, J.L. 1962. *How to do Things with Words*. Oxford University Press.

[2] Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. 2016. Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606. Retrived from https://arxiv.org/abs/1607.04606

[3] D'Angelo, C. M., Arastoopour, G., Chesler, N. C., & Shaffer, D. W. 2011. Collaborating in a virtual engineering internship. In *Computer Supported Collaborative Learning Conference*. Hong Kong SAR, Hong Kong, China, 4-8.

[4] Ezen-Can, A., & Boyer, K. E. 2013. Unsupervised classification of student dialogue acts with query-likelihood clustering. In *Educational Data Mining*.

[5] Forsyth, E. N. 2007. *Improving automated lexical and discourse analysis of online chat dialog*. Doctoral dissertation. Naval Postgraduate School, Monterey, California.

[6] Jurafsky, D., & Martin, J. H. 2009. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Prentice Hall series in artificial intelligence, p.814.

[7] Kim, J., Chern, G., Feng, D., Shaw, E., & Hovy, E. 2006. Mining and assessing discussions on the web through speech act analysis. In *Proceedings of the Workshop on Web Content Mining with Human Language Technologies at the 5th International Semantic Web Conference*.

[8] Kim, S. N., Cavedon, L., & Baldwin, T. 2012. Classifying dialogue acts in multi-party live chats. In *Proceedings of the 26th Pacific Asia Conference on Language, Information, and Computation*. 463-472.

[9] Moldovan, C., Rus, V., & Graesser, A. C. 2011. Automated Speech Act Classification For Online Chat. *MAICS*. 710, 23-29.

[10] Olney, A., Louwerse, M., Matthews, E., Marineau, J., Hite-Mitchell, H., & Graesser, A. 2003. Utterance classification in AutoTutor. In *Proceedings of the HLT-NAACL 03 workshop on Building educational applications using natural language processing*. 2. Association for Computational Linguistics, 1-8.

[11] Pagliardini, M., Gupta, P., & Jaggi, M. 2017. Unsupervised learning of sentence embeddings using compositional n-gram features. arXiv preprint arXiv:1703.02507. Retrived from https://arxiv.org/abs/1703.02507

[12] Pan, S. J., & Yang, Q. A survey on transfer learning. 2010. *IEEE Transactions on knowledge and data engineering*. 22(10), 1345-1359.

[13] Rus, V., Maharjan, N., Tamang, L. J., Yudelson, M., Berman, S., Fancsali, S. E. & Ritter, S. 2017. An Analysis of Human Tutors' Actions in Tutorial Dialogues. In *Proceedings of the Thirtieth International Florida Artificial Intelligence Research Society Conference*. AAAI, 122-127.

[14] Rus, V., Moldovan, C., Niraula, N., & Graesser, A. C. 2012. Automated Discovery of Speech Act Categories in Educational Games. *International Educational Data Mining Society*.

[15] Samei, B., Li, H., Keshtkar, F., Rus, V., & Graesser, A. C. 2012. Context-based speech act classification in intelligent tutoring systems. In *International Conference on Intelligent Tutoring Systems*. Springer, Cham, 236-241.

[16] Searle, J.R. 1969. *Speech Acts*. Cambridge University Press, GB.

[17] Shaffer, D. W. 2006. *How Computer Games Help Children Learn*. Macmillan.