Dialogue Act Prediction Using Stochastic Context-Free Grammar Induction

Jeroen Geertzen
Research Centre for English & Applied Linguistics
University of Cambridge, UK
jg532@cam.ac.uk

Abstract

This paper presents a model-based approach to dialogue management that is guided by data-driven dialogue act prediction. The statistical prediction is based on stochastic context-free grammars that have been obtained by means of grammatical inference. The prediction performance of the method compares favourably to that of a heuristic baseline and to that of n-gram language models.

The act prediction is explored both for dialogue acts without realised semantic content (consisting only of communicative functions) and for dialogue acts with realised semantic content.

1 Introduction

Dialogue management is the activity of determining how to behave as an interlocutor at a specific moment of time in a conversation: which *action* can or should be taken at what *state* of the dialogue. The systematic way in which an interlocutor chooses among the options for continuing a dialogue is often called a *dialogue strategy*.

Coming up with suitable dialogue management strategies for dialogue systems is not an easy task. Traditional methods typically involve manually crafting and tuning frames or hand-crafted rules, requiring considerable implementation time and cost. More recently, statistical methods are being used to semi-automatically obtain models that can be trained and optimised using dialogue data.[1] These methods are usually based on two assumptions. First, the training data is assumed to be representative of the communication that may be encountered in interaction. Second, it is assumed that dialogue can be modelled as a Markov Decision Process (MDP) (Levin et al., 1998), which implies that dialogue is modelled as a sequential decision task in which each contribution (action) results in a transition from one state to another.

The latter assumption allows to assign a *reward* for action-state pairs, and to determine the dialogue management strategy that results in the maximum expected reward by finding for each state the optimal action by using *reinforcement learning* (cf. (Sutton and Barto, 1998)). Reinforcement learning approaches to dialogue management have proven to be successful in several task domains (see for example (Paek, 2006; Lemon et al., 2006)). In this process there is no supervision, but what is optimal depends usually on factors that require human action, such as task completion or user satisfaction.

The remainder of this paper describes and evaluates a model-based approach to dialogue management in which the decision process of taking a particular action given a dialogue state is guided by data-driven dialogue act prediction. The approach improves over n-gram language models and can be used in isolation or for user simulation, without yet providing a full alternative to reinforcement learning.

2 Using structural properties of task-oriented dialogue

One of the best known regularities that are observed in dialogue are the two-part structures, known as *adjacency pairs* (Schegloff, 1968), like QUESTION-ANSWER or GREETING-GREETING.

A simple model of predicting a plausible next dialogue act that deals with such regularities could be based on bigrams, and to include more context also higher-order n-grams could be used. For instance, Stolcke et al. (2000) explore n-gram models based on transcribed words and prosodic information for SWBD-DAMSL dialogue acts in the Switchboard corpus (Godfrey et al., 1992). After training back-off n-gram models (Katz, 1987) of

[1] See e.g. (Young, 2002) for an overview.

different order using frequency smoothing (Witten and Bell, 1991), it was concluded that trigrams and higher-order n-grams offer a small gain in predication performance with respect to bigrams.

Apart from adjacency pairs, there is a variety of more complex re-occurring interaction patterns. For instance, the following utterances with corresponding dialogue act types illustrate a clarification sub-dialogue within an information-request dialogue:

1	A: How do I do a fax?	QUESTION
2	B: Do you want to send or print one?	QUESTION
3	A: I want to print it	ANSWER
4	B: Just press the grey button	ANSWER

Such structures have received considerable attention and their models are often referred to as discourse/dialogue grammars (Polanyi and Scha, 1984) or conversational/dialogue games (Levin and Moore, 1988).

As also remarked by Levin (1999), predicting and recognising dialogue games using n-gram models is not really successful. There are various causes for this. The flat horizontal structure of n-grams does not allow (hierarchical) grouping of symbols. This may weaken the predictive power and reduces the power of the representation since nested structures such as exemplified above cannot be represented in a straightforward way.

A better solution would be to express the structure of dialogue games by a context-free grammar (CFG) representation in which the terminals are dialogue acts and the non-terminals denote conversational games. Construction of a CFG would require explicit specification of a discourse grammar, which could be done by hand, but it would be a great advantage if CFGs could automatically be induced from the data. An additional advantage of grammar induction is the possibility to assess the frequency of typical patterns and a stochastic context-free grammar (SCFG) may be produced which can be used for parsing the dialogue data.

3 Sequencing dialogue acts

Both n-gram language models and SCFG based models work on sequences of symbols. Using more complex symbols increases data sparsity: encoding more information increases the number of unique symbols in the dataset and decreases the number of reoccurring patterns which could be used in the prediction.

In compiling the symbols for the prediction experiments, three aspects are important: the identification of interlocutors, the definition of dialogue acts, and multifunctionality in dialogue.

The dialogue act taxonomy that is used in the prediction experiments is that of DIT (Bunt, 2000). A dialogue act is defined as a pair consisting of a communicative function (CF) and a semantic content (SC): $a =< CF, SC >$. The DIT taxonomy distinguishes 11 dimensions of communicative functions, addressing information about the task domain, feedback, turn management, and other generic aspects of dialogue (Bunt, 2006). There are also functions, called *the general-purpose functions*, that may occur in any dimension. In quite some cases, particularly when dialogue control is addressed and dimension-specific functions are realised, the SC is empty. General-purpose functions, by contrast, are always used in combination with a realised SC. For example:

	dialogue act	
utterance	*function*	*semantic content*
What to do next?	SET-QUESTION	`next-step(X)`
Press the button.	SET-ANSWER	`press(Y)` $\wedge$ `button(Y)`

The SC —if realised— describes objects, properties, and events in the domain of conversation.

In dialogue act prediction while taking multidimensionality into account, a dialogue D can be represented as a sequence of events in which an event is a set of one dialogue act or multiple dialogue acts occurring simultaneously. The information concerning interlocutor and multifunctionality is encoded in a single symbol and denoted by means of a n-tuple. Assuming that at most three functions can occur simultaneously, a 4-tuple is needed[2]: `(interlocutor,da1,da2,da3)`. An example of a bigram of 4-tuples would then look as follows:

(A,<SET-Q,"next-step(X)">,_,_),
(B,<SET-A,"press(Y) ∧ button(Y)">,_,_)

Two symbols are considered to be identical when the same speaker is involved and when the symbols both address the same functions. To make

[2] Ignoring the half percent of occurrences with four simultaneous functions.

it easy to determine if two symbols are identical, the order of elements in a tuple is fixed: functions that occur simultaneously are first ordered on importance of dimension, and subsequently on alphabet. The task-related functions are considered the most important, followed by feedback-related functions, followed by any other remaining functions. This raises the question how recognition performance using multifunctional symbols compares against recognition performance using symbols that only encode the primary function

4 N-gram language models

There exists a significant body of work on the use of language models in relation to dialogue management. Nagata and Morimoto (1994) describe a statistical model of discourse based on trigrams of utterances classified by custom speech act types. They report 39.7% prediction accuracy for the top candidate and 61.7% for the top three candidates.

In the context of the dialogue component of the speech-to-speech translation system VERBMOBIL, Reithinger and Maier (1995) use n-gram dialogue act probabilities to suggest the most likely dialogue act. In later work, Alexandersson and Reithinger (1997) describe an approach which comes close to the work reported in this paper: Using grammar induction, plan operators are semi-automatically derived and combined with a statistical disambiguation component. This system is claimed to have an accuracy score of around 70% on turn management classes.

Another study is that of Poesio and Mikheev (1998), in which prediction based on the previous dialogue act is compared with prediction based on the context of dialogue games. Using the Map Task corpus annotated with ‘moves’ (dialogue acts) and ‘transactions’ (games) they showed that by using higher dialogue structures it was possible to perform significantly better than a bigram model approach. Using bigrams, 38.6% accuracy was achieved. Additionally taking game structure into account resulted in 50.6%; adding information about speaker change resulted in an accuracy of 41.8% with bigrams, 54% with game structure.

All studies discussed so far are only concerned with sequences of communicative functions, and disregard the semantic content of dialogue acts.

5 Dialogue grammars

To automatically induce patterns from dialogue data in an unsupervised way, grammatical inference (GI) techniques can be used. GI is a branch of unsupervised machine learning that aims to find structure in symbolic sequential data. In this case, the input of the GI algorithm will be sequences of dialogue acts.

5.1 Dialogue Grammars Inducer

For the induction of structure, a GI algorithm has been implemented that will be referred to as Dialogue Grammars Inducer (DGI). This algorithm is based on distributional clustering and alignment-based learning (van Zaanen and Adriaans, 2001; van Zaanen, 2002; Geertzen and van Zaanen, 2004). Alignment-based learning (ABL) is a symbolic grammar inference framework that has successfully been applied to several unsupervised machine learning tasks in natural language processing. The framework accepts sequences with symbols, aligns them with each other, and compares them to find interchangeable subsequences that mark structure. As a result, the input sequences are augmented with the induced structure.

The DGI algorithm takes as input time series of dialogue acts, and gives as output a set of SCFGs. The algorithm has five phases:

1. SEGMENTATION: In the first phase of DGI, the time series are —if necessary— segmented in smaller sequences based on a specific time interval in which no communication takes place. This is a necessary step in task-oriented conversation in which there is ample time to discuss (and carry out) several related tasks, and an interaction often consists of a series of short dialogues.

2. ALIGNMENT LEARNING: In the second phase a search space of possible structures, called hypotheses, is generated by comparing all input sequences with each other and by clustering sub-sequences that share similar context. To illustrate the alignment learning, consider the following input sequences:

A:SET-Q,	B:PRO-Q,	A:PRO-A,	B:SET-A.
A:SET-Q,	B:PAUSE,	B:RESUME,	B:SET-A.
A:SET-Q,	B:SET-A.		

The alignment learning compares all input sequences with each other, and produces the hypothesised structures depicted below. The induced structure is represented using bracketing.

$[_i$ A:SET-Q, $[_j$ B:PRO-Q, A:PRO-A, $]_j$ B:SET-A. $]_i$
$[_i$ A:SET-Q, $[_j$ B:PAUSE, A:RESUME, $]_j$ B:SET-A. $]_i$
$[_i$ A:SET-Q, $[_j$ $]_j$ B:SET-A. $]_i$

The hypothesis j is generated because of the similar context (which is underlined). The hypothesis i, the full span, is introduced by default, as it might be possible that the sequence is in itself a part of a longer sequence.

3. SELECTION LEARNING: The set of hypotheses that is generated during alignment learning contains hypotheses that are unlikely to be correct. These hypotheses are filtered out, overlapping hypotheses are eliminated to assure that it is possible to extract a context-free grammar, and the remaining hypotheses are selected and remain in the bracketed output. The decision of which hypotheses to select and which to discard is based on a Viterbi beam search (Viterbi, 1967).

4. EXTRACTION: In the fourth phase, SCFG grammars are extracted from the remaining hypotheses by means of recursive descent parsing. Ignoring the stochastic information, a CFG of the above-mentioned example looks in terms of grammar rules as depicted below:

S	⇒	A:SET-Q	**J**	B:SET-A
J	⇒	B:PRO-Q	A:PRO-A	
J	⇒	B:PAUSE	A:RESUME	
J	⇒	∅		

5. FILTERING: In the last phase, the SCFG grammars that have small coverage or involve many non-terminals are filtered out, and the remaining SCFG grammars are presented as the output of DGI.

Depending on the mode of working, the DGI algorithm can generate a SCFG covering the complete input or can generate a set of SCFGs. In the former mode, the grammar that is generated can be used for parsing sequences of dialogue acts and by doing so suggests ways to continue the dialogue. In the latter mode, by parsing each grammar in the set of grammars that are expected to represent dialogue games in parallel, specific dialogue games may be recognised, which can in turn be used beneficially in dialogue management.

5.2 A worked example

In testing the algorithm, DGI has been used to infer a set of SCFGs from a development set of 250 utterances of the DIAMOND corpus (see also Section 6.1). Already for this small dataset, DGI produced, using default parameters, 45 'dialogue games'. One of the largest produced structures was the following:

4	**S**	⇒	A:SET-Q , **NTAX** , **NTBT** , B:SET-A
4	**NTAX**	⇒	B:PRO-Q , NTFJ
3	**NTFJ**	⇒	A:PRO-A
1	**NTFJ**	⇒	A:PRO-A , A:CLARIFY
2	**NTBT**	⇒	B:PRO-Q , A:PRO-A
2	**NTBT**	⇒	∅

In this figure, each CFG rule has a number indicating how many times the rules has been used. One of the dialogue fragments that was used to induce this structure is the following excerpt:

	utterance	*dialogue act*
A_1	how do I do a short code?	SET-Q
B_1	do you want to program one?	PRO-Q
A_2	no	SET-A
A_3	I want to enter a kie* a short code	CLARIFY
B_2	you want to use a short code?	PRO-Q
A_4	yes	PRO-A
B_3	press the VK button	SET-A

Unfortunately, many of the 45 induced structures were very small or involved generalisations already based on only two input samples. To ensure that the grammars produced by DGI generalise better and are less fragmented, a post-processing step has been added which traverses the grammars and eliminates generalisations based on a low number of samples. In practice, this means that the post-processing requires the remaining grammatical structure to be presented N times or more in the data.[3]. The algorithm without post-processing will be referred to as DGI1; the algorithm with post-processing as DGI2.

6 Act prediction experiments

To determine how to behave as an interlocutor at a specific moment of time in a conversation, the DGI algorithm can be used to infer a SCFG that models the structure of the interaction. The SCFG

[3] $N = 2$ by default, but may increase with the size of the training data.

can then be used to suggest a next dialogue act to continue the dialogue. In this section, the performance of the proposed SCFG based dialogue model is compared with the performance of the well-known n-gram language models, both trained on intentional level, i.e. on sequences of sets of dialogue acts.

6.1 Data

The task-oriented dialogues used in the dialogue act prediction tasks were drawn from the DIAMOND corpus (Geertzen et al., 2004), which contains human-machine and human-human Dutch dialogues that have an assistance seeking nature. The dataset used in the experiments contains $1,214$ utterances representing $1,592$ functional segments from the human-human part of corpus. In the domain of the DIAMOND data, i.e. operating a fax device, the predicates and arguments in the logical expressions of the SC of the dialogue acts refer to entities, properties, events, and tasks in the application domain. The application domain of the fax device is complex but small: the domain model consists of 70 entities with at most 10 properties, 72 higher-level actions or tasks, and 45 different settings.

Representations of semantic content are often expressed in some form of predicate logic type formula. Examples are Quasi Logical Forms (Alshawi, 1990), Dynamic Predicate Logic (Groenendijk and Stokhof, 1991), and Underspecified Discourse Representation Theory (Reyle, 1993). The SC in the dataset is in a simplified first order logic similar to quasi logical forms, and is suitable to support feasible reasoning, for which also theorem provers, model builders, and model checkers can be used. The following utterances and their corresponding SC characterise the dataset:

1 wat moet ik nu doen?
(what do I have to do now?)
λx . next-step(x)

2 druk op een toets
(press a button)
λx . press(x) $\wedge$ button(x)

3 druk op de groene toets
(press the green button)
λx . press(x) $\wedge$ button(x) $\wedge$ color(x,'green')

4 wat zit er boven de starttoets?
(what is located above the starttoets?)
λx . loc-above(x,'button041')

Three types of predicate groups are distinguished: action predicates, element predicates, and property predicates. These types have a fixed order. The action predicates appear before element predicates, which appear in turn before property predicates. This allows to simplify the semantic content for the purpose of reducing data sparsity in act prediction experiments, by stripping away e.g. property predicates. For instance, if desired the SC of utterance 3 in the example could be simplified to that of utterance 2, making the semantics less detailed but still meaningful.

6.2 Methodology and metrics

Evaluation of overall performance in communication is problematic; there are no generally accepted criteria as to what constitutes an objective and sound way of comparative evaluation. An often-used paradigm for dialogue system evaluation is PARADISE (Walker et al., 2000), in which the performance metric is derived as a weighted combination of subjectively rated user satisfaction, task-success measures and dialogue cost. Evaluating if the predicted dialogue acts are considered as positive contributions in such a way would require the model to be embedded in a fully working dialogue system.

To assess whether the models that are learned produce human-like behaviour without resorting to costly user interaction experiments, it is needed to compare their output with real human responses given in the same contexts. This will be done by deriving a model from one part of a dialogue corpus and applying the model on an 'unseen' part of the corpus, comparing the suggested next dialogue act with the observed next dialogue act. To measure the performance, *accuracy* is used, which is defined as the proportion of suggested dialogue acts that match the observed dialogue acts.

In addition to the accuracy, also *perplexity* is used as metric. Perplexity is widely used in relation to speech recognition and language models, and can in this context be understood as a metric that measures the number of equiprobable possible choices that a model faces at a given moment. Perplexity, being related to entropy is defined as follows:

$$\begin{aligned} Entropy &= -\sum_i p(w_i|h) \cdot log_2\, p(w_i|h) \\ Perplexity &= 2^{Entropy} \end{aligned}$$

where h denotes the conditioned part, i.e. w_{i-1} in the case of bigrams and w_{i-2}, w_{i-1} in the case of trigrams, et cetera. In sum, accuracy could be described as a measure of correctness of the hypothesis and perplexity could be described as how probable the correct hypothesis is.

For all n-gram language modelling tasks reported, good-turing smoothing was used (Katz, 1987). To reduce the effect of imbalances in the dialogue data, the results were obtained using 5-fold cross-validation.

To have an idea how the performance of both the n-gram language models and the SCFG models relate to the performance of a simple heuristic, a baseline has been computed which suggests a majority class label according to the interlocutor role in the dialogue. The information seeker has SET-Q and the information provider has SET-A as majority class label.

6.3 Results for communicative functions

The scores for communicative function prediction are presented in Table 1. For each of the three kinds of symbols, accuracy and perplexity are calculated: the first two columns are for the main CF, the second two columns are for the combination of speaker identity *and* main CF, and the third two columns are for the combination of speaker identity and all CFs. The scores for the latter two codings could not be calculated for the 5-gram model, as the data were too sparse.

As was expected, there is an improvement in both accuracy (increasing) and perplexity (decreasing) for increasing n-gram order. After the 4-gram language model, the scores drop again. This could well be the result of insufficient training data, as the more complex symbols could not be predicted well.

Both language models and SCFG models perform better than the baseline, for all three groups. The two SCFG models, DGI1 and DGI2, clearly outperform the n-gram language models with a substantial difference in accuracy. Also the perplexity tends to be lower. Furthermore, model DGI2 performs clearly better than model DGI1, which indicates that the 'flattening' of non-terminals which is described in Section 5 results in better inductions.

When comparing the three groups of sequences, it can be concluded that providing the speaker identity combined with the main communicative function results in better accuracy scores of 5.9% on average, despite the increase in data sparsity. A similar effect has also been reported by Stolcke et al. (2000).

Only for the 5-gram language model, the data become too sparse to learn reliably a language model from. There is again an increase in performance when also the last two positions in the 4-tuple are used and all available dialogue act assignments are available. It should be noted, however, that this increase has less impact than adding the speaker identity. The best performing n-gram language model achieved 66.4% accuracy; the best SCFG model achieved 78.9% accuracy.

6.4 Results for dialogue acts

The scores for prediction of dialogue acts, including SC, are presented in the left part of Table 2. The presentation is similar to Table 1: for each of the three kinds of symbols, accuracy and perplexity were calculated. For dialogue acts that may include semantic content, computing a useful baseline is not obvious. The same baseline as for communicative functions was used, which results in lower scores.

The table shows that the attempts to learn to predict additionally the semantic content of utterances quickly run into data sparsity problems. It turned out to be impossible to make predictions from 4-grams and 5-grams, and for 3-grams the combination of speaker and all dialogue acts could not be computed. Training the SCFGs, by contrast, resulted in fewer problems with data sparsity, as the models abstract quickly.

As with predicting communicative functions, the SCFG models show better performance than the n-gram language models, for which the 2-grams show slightly better results than the 3-grams. Where there was a notable performance difference between DGI1 and DGI2 for CF prediction, for dialogue act prediction there is only a very little difference, which is insignificant considering the relatively high standard deviation. This small difference is explained by the fact that DGI2 becomes less effective as the size of the training data decreases.

As with CF prediction, it can be concluded that providing the speaker identity with the main dialogue act results in better scores, but the difference is less big than observed with CF prediction due to the increased data sparsity.

Table 1: Communicative function prediction scores for n-gram language models and SCFGs in accuracy (*acc*, in percent) and perplexity (*pp*). CF$_{main}$ denotes the main communicative function, SPK speaker identity, and CF$_{all}$ all occurring communicative functions.

	CF$_{main}$		SPK + CF$_{main}$		SPK + CF$_{all}$	
	acc	*pp*	*acc*	*pp*	*acc*	*pp*
baseline	39.1±0.23	24.2±0.19	44.6±0.92	22.0±0.25	42.9±1.33	23.7±0.41
2-gram	53.1±0.88	17.9±0.35	58.3±1.84	16.8±0.31	61.1±1.65	16.3±0.59
3-gram	58.6±0.85	17.1±0.47	63.0±1.98	14.5±0.26	65.9±1.92	14.0±0.23
4-gram	60.9±1.12	16.7±0.15	65.4±1.62	15.2±1.07	66.4±2.03	14.2±0.44
5-gram	60.3±0.43	18.6±0.21	-	-	-	-
DGI1	67.4±3.05	18.3±1.28	74.6±1.94	14.8±1.47	76.5±2.13	13.9±0.35
DGI2	71.8±2.67	16.1±1.25	78.3±2.50	14.0±2.39	78.9±1.98	13.6±0.35

Table 2: Dialogue act prediction scores for n-gram language models and SCFGs. DA$_{main}$ denotes the dialogue act with the main communicative function, and DA$_{all}$ all occurring dialogue acts.

	DA$_{main}$		SPK + DA$_{main}$		SPK + DA$_{all}$			
					full SC		simplified SC	
	acc	*pp*	*acc*	*pp*	*acc*	*pp*	*acc*	*pp*
baseline	18.5±2.01	31.0±1.64	19.3±1.79	27.6±0.93	18.2±1.93	31.6±1.38	18.2±1.93	31.6±1.38
2-gram	31.2±1.42	28.5±1.03	34.6±1.51	24.7±0.62	35.1±1.30	26.9±0.47	37.5±1.34	26.2±2.37
3-gram	29.0±1.14	34.7±2.82	31.9±1.21	30.5±2.06	-	-	29.1±1.28	28.0±2.59
4-gram	-	-	-	-	-	-	-	-
5-gram	-	-	-	-	-	-	-	-
DGI1	38.8±3.27	25.1±0.94	42.5±0.96	25.0±1.14	42.9±2.44	27.3±1.98	46.6±2.01	24.6±2.24
DGI2	39.2±2.45	25.0±1.28	42.7±1.03	25.3±0.99	42.4±2.19	28.0±1.57	46.4±1.94	24.7±2.55

The prediction scores of dialogue acts with full semantic content and simplified semantic content are presented in the right part of Table 2. For both cases multifunctionality is taken into account by including all occurring communicative functions in each symbol. As can be seen from the table, the simplification of the semantic content leads to improvements in the prediction performance for both types of model. The best n-gram language model improved with 2.4% accuracy from 35.1% to 37.5%; the best SCFG-based model improved with 3.7% from 42.9% to 46.6%.

Moreover, the simplification of the semantic content reduced the problem of data-sparsity, making it also possible to predict based on 3-grams although the accuracy is considerably lower than that of the 2-gram model.

7 Discussion

Both n-gram language models and SCFG based models have their strengths and weaknesses. n-gram models have the advantage of being very robust and they can be easily trained. The SCFG based model can capture regularities that have gaps, and allow to model long(er) distance relations. Both algorithms work on sequences and hence are easily susceptible to data-sparsity when the symbols in the sequences get more complex. The SCFG approach, though, has the advantage that symbols can be clustered in the non-terminals of the grammar, which allows more flexibility.

The multidimensional nature of the DIT^{++} functions can be adequately encoded in the symbols of the sequences. Keeping track of the interlocutor and including not only the main communicative function but also other functions that occur simultaneously results in better performance even though it decreases the amount of data to learn from.

The prediction experiments based on main communicative functions assume that in case of multifunctionality, a main function can clearly be identified. Moreover, it is assumed that task-related functions are more important than feedback functions or other functions. For most cases, these assumptions are justified, but in some cases they are problematic. For instance, in a heated discussion, the turn management function could be considered more important for the dialogue than a simultaneously occurring domain specific function. In other cases, it is impossible to clearly identify a main function as all functions occurring simultaneously are equally important to the dialogue.

In general, n-grams of a higher order have a higher predictability and therefore a lower perplexity. However, using high order n-grams is problematic due to sparsity of training data, which clearly is the case with 4-grams, and particularly with 5-grams in combination with complex symbols as for CF prediction.

Considerably more difficult is the prediction of dialogue acts with realised semantic content, as is evidenced in the differences in accuracy and perplexity for all models. Considering that the data set, with about $1,600$ functional segments, is rather small, the statistical prediction of logical expressions increases data sparsity to such a degree that from the n-gram language models, only 2-gram (and 3-grams to some extent) could be trained. The SCFG models can be trained for both CF prediction and dialogue act prediction.

As noted in Section 6.2, objective evaluation of dialogue strategies and behaviour is difficult. The evaluation approach used here compares the suggested next dialogue act with the next dialogue act as observed. This is done for each dialogue act in the test set. This evaluation approach has the advantage that the evaluation metric can easily be understood and computed. The approach, however, is also very strict: in a given dialogue context, continuations with various types of dialogue acts may all be equally appropriate. To also take other possible contributions into account, a rich dataset is required in which interlocutors act differently in similar dialogue context with a similar established common ground. Moreover, such a dataset should contain for each of these cases with similar dialogue context a representative set of samples.

8 Conclusions and future work

An approach to the prediction of communicative functions and dialogue acts has been presented that makes use of grammatical inference to automatically extract structure from corpus data. The algorithm, based on alignment-based learning, has been tested against a baseline and several n-gram language models. From the results it can be concluded that the algorithm outperforms the n-gram models: on the task of predicting the communicative functions, the best performing n-gram model achieved 66.4% accuracy; the best SCFG model achieved 78.9% accuracy. Predicting the semantic content in combination with the communicative functions is difficult, as evidenced by moderate scores. Obtaining lower degree n-gram language models is feasible, whereas higher degree models are not trainable. Prediction works better with the SCFG models, but does not result in convincing scores. As the corpus is small, it is expected that with scaling up the available training data, scores will improve for both tasks.

Future work in this direction can go in several directions. First, the grammar induction approach shows potential of learning dialogue game-like structures unsupervised. The performance on this task could be tested and measured by applying the algorithm on corpus data that have been annotated with dialogue games. Second, the models could also be extended to incorporate more information than dialogue acts alone. This could make comparisons with the performance obtained with reinforcement learning or with Bayesian networks interesting. Third, it would be interesting to learn and apply the same models on other kinds of conversation, such as dialogue with more than two interlocutors. Fourth, datasets could be drawn from a large corpus that covers dialogues on a small but complex domain. This makes it possible to evaluate according to the possible continuations as found in the data for situations with similar dialogue context, rather than to evaluate according to a single possible continuation. Last, the rather unexplored parameter space of the DGI algorithm might be worth exploring in optimising the system's performance.

References

Jan Alexandersson and Norbert Reithinger. 1997. Learning dialogue structures from a corpus. In *Proceedings of Eurospeech 1997*, pages 2231–2234, Rhodes, Greece, September.

Hiyan Alshawi. 1990. Resolving quasi logical forms. *Computational Linguistics*, 16(3):133–144.

Harry Bunt. 2000. Dialogue pragmatics and context specification. In Harry Bunt and William Black, editors, *Abduction, Belief and Context in Dialogue; Studies in Computational Pragmatics*, pages 81–150. John Benjamins, Amsterdam, The Netherlands.

Harry Bunt. 2006. Dimensions in dialogue annotation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 1444–1449, Genova, Italy, May.

Jeroen Geertzen and Menno M. van Zaanen. 2004. Grammatical inference using suffix trees. In *Proceedings of the 7th International Colloquium on Grammatical Inference (ICGI)*, pages 163–174, Athens, Greece, October.

Jeroen Geertzen, Yann Girard, and Roser Morante. 2004. The DIAMOND project. Poster at the 8th Workshop on the Semantics and Pragmatics of Dialogue (CATALOG 2004), Barcelona, Spain, July.

John Godfrey, Edward Holliman, and Jane McDaniel. 1992. SWITCHBOARD: Telephone speech corpus for research and development. In *Proceedings of the ICASSP-92*, pages 517–520, San Francisco, USA.

Jeroen Groenendijk and Martin Stokhof. 1991. Dynamic predicate logic. *Linguistics and Philosophy*, 14(1):39–100.

Slava M. Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(3):400–401.

Oliver Lemon, Kallirroi Georgila, and James Henderson. 2006. Evaluating effectiveness and portability of reinforcement learned dialogue strategies with real users: The talk towninfo evaluation. In *Spoken Language Technology Workshop*, pages 178–181.

Joan A. Levin and Johanna A. Moore. 1988. Dialogue-games: metacommunication structures for natural language interaction. *Distributed Artificial Intelligence*, pages 385–397.

Esther Levin, Roberto Pieraccini, and Wieland Eckert. 1998. Using markov decision process for learning dialogue strategies. In *Proceedings of the ICASSP'98*, pages 201–204, Seattle, WA, USA.

Lori Levin, Klaus Ries, Ann Thymé-Gobbel, and Alon Lavie. 1999. Tagging of speech acts and dialogue games in spanish call home. In *Proceedings of ACL-99 Workshop on Discourse Tagging*, College Park, MD, USA.

Masaaki Nagata and Tsuyoshi Morimoto. 1994. First steps towards statistical modeling of dialogue to predict the speech act type of the next utterance. *Speech Communication*, 15(3-4):193–203.

Tim Paek. 2006. Reinforcement learning for spoken dialogue systems: Comparing strenghts and weaknesses for practical deployment. In *Interspeech Workshop on "Dialogue on Dialogues"*.

Massimo Poesio and Andrei Mikheev. 1998. The predictive power of game structure in dialogue act recognition: Experimental results using maximum entropy estimation. In *Proceedings International Conference on Spoken Language Processing (ICSLP-98)*, Sydney, Australia, December.

Livia Polanyi and Remko Scha. 1984. A syntactic approach to discourse semantics. In *Proceedings of the 10th international conference on Computational linguistics*, pages 413–419, Stanford, CA, USA.

Norbert Reithinger and Elisabeth Maier. 1995. Utilizing statistical dialogue act processing in VERBMOBIL. In *Proceedings of the 33rd annual meeting on the Association for Computational Linguistics (ACL)*, pages 116–121, Cambridge, Massachusetts. Association for Computational Linguistics (ACL).

Uwe Reyle. 1993. Dealing with ambiguities by underspecification: Construction, representation and deduction. *Journal of Semantics*, 10(2):123–179.

Emanuel A. Schegloff. 1968. Sequencing in conversational openings. *American Anthropologist*, 70:1075–1095.

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.

Richard S. Sutton and Andrew G. Barto. 1998. *Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning)*. MIT Press, March.

Menno van Zaanen and Pieter W. Adriaans. 2001. Comparing two unsupervised grammar induction systems: Alignment-Based Learning vs. EMILE. Technical Report TR2001.05, University of Leeds, Leeds, UK, March.

Menno M. van Zaanen. 2002. *Bootstrapping Structure into Language: Alignment-Based Learning*. Ph.D. thesis, University of Leeds, Leeds, UK, January.

Andrew J. Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, April.

Marilyn Walker, Candace Kamm, and Diane Litman. 2000. Towards developing general models of usability with paradise. *Natural Language Engineering*, 6(3-4):363–377.

Ian H. Witten and Timothy C. Bell. 1991. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory*, 37(4):1085–1094.

Steve Young. 2002. The statistical approach to the design of spoken dialogue systems. Technical Report CUED/F-INFENG/TR.433, Engineering Department, Cambridge University, UK, September.