Overgeneration and ranking for spoken dialogue systems

Sebastian Varges
Center for the Study of Language and Information
Stanford University
Stanford, CA 94305, USA
varges@stanford.edu

Abstract

We describe an implemented generator for a spoken dialogue system that follows the 'overgeneration and ranking' approach. We find that overgeneration based on bottom-up chart generation is well-suited to a) model phenomena such as alignment and variation in dialogue, and b) address robustness issues in the face of imperfect generation input. We report evaluation results of a first user study involving 20 subjects.

1 Introduction

Overgeneration and ranking approaches have become increasingly popular in recent years (Langkilde, 2002; Varges, 2002). However, most work on generation for practical dialogue systems makes use of generation components that work toward a single output, often using simple templates. In the following, we first describe our dialogue system and then turn to the generator which is based on the overgeneration and ranking paradigm. We outline the results of a user study, followed by a discussion section.

The dialogue system: Dialogue processing starts with the output of a speech recognizer (Nuance) which is analyzed by both a statistical dependency parser and a topic classifier. Parse trees and topic labels are matched by the 'dialogue move scripts' of the dialogue manager (DM) (Mirkovic and Cavedon, 2005). The dialogue system is fully implemented and has been used in restaurant selection and MP3 player tasks (Weng et al., 2004). There are 41 task-independent, generic dialogue rules, 52 restaurant selection rules and 89 MP3 player rules. Query constraints are built by dialogue move scripts if the parse tree matches input patterns specified in the scripts. For example, a request "I want to find an inexpensive Japanese restaurant that takes reservations" results in constraints such as `restaurant:Cuisine = restaurant:japanese` and `restaurant:PriceLevel = 0-10`. If the database query constructed from these constraints returns no results, various constraint modification strategies such as constraint relaxation or removal can be employed. For example, 'Japanese food' can be relaxed to 'Asian food' since cuisine types are hierarchically organized.

2 Overgeneration for spoken dialogue

Table 1 shows some example outputs of the system. The wording of the realizations is informed by a wizard-of-oz data collection. The task of the generator is to produce these verbalizations given dialogue strategy, constraints and further discourse context, i.e. the input to the generator is non-linguistic. We perform mild overgeneration of candidate moves, followed by ranking. The highest-ranked candidate is selected for output.

2.1 Chart generation

We follow a bottom-up chart generation approach (Kay, 1996) for production systems similar to (Varges, 2005). The rule-based core of the generator is a set of productions written in a production system. Productions map individual database constraints to phrases such as "open for lunch", "within 3 miles", "a formal dress code", and recursively combine them into NPs. This includes the use of coordination to produce "restaurants with a 5-star rating and a formal dress code", for example. The NPs are integrated into sentence templates, several of which can be combined

Proceedings of the Fourth International Natural Language Generation Conference, pages 20–22,
Sydney, July 2006. ©2006 Association for Computational Linguistics

	$\|result\|$	mod	example realization	f_{exp}
s1	0	no	I'm sorry but I found no restaurants on Mayfield Road that serve Mediterranean food .	0
s2	small: $> 0, < t_1$	no	There are 2 cheap Thai restaurants in Lincoln in my database : Thai Mee Choke and Noodle House .	61
s3	medium: $>= t_1, < t_2$	no	I found 9 restaurants with a two star rating and a formal dress code that are open for dinner and serve French food . Here are the first ones :	212
s4	large: $>= t_2$	no	I found 258 restaurants on Page Mill Road, for example Maya Restaurant , Green Frog and Pho Hoa Restaurant . Would you like to try searching by cuisine ?	300
s5	large	yes	I found no restaurants that ... However, there are NUM restaurants that ... Would you like to ...?	16
s6	(any)	yes/no	I found 18 items .	2

Table 1: Some system responses ('$|result|$': size of database result set, 'mod': performed modifications). Last column: frequency in user study (180 tasks, 596 constraint inputs to generator)

to form an output candidate turn. For example, a constraint realizing template "I found no [NP-original] but there are [NUM] [NP-optimized] in my database" can be combined with a follow-up sentence template such as "You could try to look for [NP-constraint-suggestion]". 'NP-original' realizes constraints directly constructed from the user utterance; 'NP-optimized' realizes potentially modified constraints used to obtain the actual query result. To avoid generating separate sets of NPs independently for these two – often largely overlapping – constraint sets, we assign unique indices to the input constraints, overgenerate NPs and check their indices.

The generator maintains state across dialogue turns, allowing it to track its previous decisions (see 'variation' below). Both input constraints and chart edges are indexed by turn numbers to avoid confusing edges of different turns.

We currently use 102 productions overall in the restaurant and MP3 domains, 38 of them to generate NPs that realize 19 input constraints.

2.2 Ranking: alignment & variation

Alignment Alignment is a key to successful natural language dialogue (Brockmann et al., 2005). We perform alignment of system utterances with user utterances by computing an ngram-based overlap score. For example, a user utterance "I want to find a Chinese restaurant" is presented by the bag-of-words {'I', 'want', 'to', 'find', ...} and the bag-of-bigrams {'I want', 'want to', 'to find', ...}. We compute the overlap with candidate system utterances represented in the same way and combine the unigram and bigram match scores. Words are lemmatized and proper nouns of example items removed from the utterances.

Alignment allows us to prefer "restaurants that serve Chinese food" over "Chinese restaurants" if the user used a wording more similar to the first. The Gricean Maxim of Brevity, applied to NLG in (Dale and Reiter, 1995), suggests a preference for the second, shorter realization. However, if the user thought it necessary to use "serves", maybe to correct an earlier mislabeling by the classifier/parse-matching patterns, then the system should make it clear that it understood the user correctly by using those same words. On the other hand, a general preference for brevity is desirable in spoken dialogue systems: users are generally not willing to listen to lengthy synthesized speech.

Variation We use a variation score to 'cycle' over sentence-level paraphrases. Alternative candidates for realizing a certain input move are given a unique alternation ('alt') number in increasing order. For example, for the simple move `continuation_query` we may assign the following alt values: "Do you want more?" (alt=1) and "Do you want me to continue?" (alt=2). The system cycles over these alternatives in turn. Once we reach alt=2, it starts over from alt=1. The actual alt 'score' is inversely related to recency and normalized to [0...1].

Score combination The final candidate score is a linear combination of alignment and variation scores:

$$score_{final} = \lambda_1 \cdot align_{uni,bi} + (1 - \lambda_1) \cdot variation \quad (1)$$

$$align_{uni,bi} = \lambda_2 \cdot align_{uni} + (1 - \lambda_2) \cdot align_{bi} \quad (2)$$

where $\lambda_1, \lambda_2 \in \{0...1\}$. A high value of λ_1 places more emphasis on alignment, a low value yields candidates that are more different from previously chosen ones. In our experience, alignment should be given a higher weight than variation, and, within alignment, bigrams should be weighted higher than unigrams, i.e. $\lambda_1 > 0.5$ and $\lambda_2 < 0.5$. Deriving weights empirically from corpus data is an avenue for future research.

3 User study

Each of 20 subjects in a restaurant selection task was given 9 scenario descriptions involving 3 constraints. We use a back-end database of 2500 restaurants containing the 13 attributes/constraints for each restaurant.

On average, the generator produced 16 output candidates for inputs of two constraints, 160 candidates for typical inputs of 3 constraints and 320 candidates for 4 constraints. For larger constraint sets, we currently reduce the level of overgeneration but in the future intend to interleave overgeneration with ranking similar to (Varges, 2002).

Task completion in the experiments was high: the subjects met all target constraints in 170 out of 180 tasks, i.e. completion rate was 94.44%. To the question "The responses of the system were appropriate, helpful, and clear." (on a scale where 1 = 'strongly agree', 5 = 'strongly disagree'), the subjects gave the following ratings: 1: 7, 2: 9, 3: 2, 4: 2 and 5: 0, i.e. the mean user rating is 1.95.

4 Discussion & Conclusions

Where NLG affects the dialogue system: Discourse entities introduced by NLG add items to the system's salience list as an equal partner to NLU.

Robustness: due to imperfect ASR and NLU, we relax completeness requirements when doing overgeneration, and reason about the generation input by adding defaults for missing constraints, checking ranges of attribute values etc. Moreover, we use a template generator as a fall-back if NLG fails to at least give some feedback to the user (s6 in table 1).

What-to-say vs how-to-say-it: the classic separation of NLG into separate modules also holds in our dialogue system, albeit with some modifications: 'content determination' is ultimately performed by the user and the constraint optimizer. The presentation dialogue moves do microplanning, for example by deciding to present retrieved database items either as examples (s4 in table 1) or as part of a larger answer list of items. The chart generator performs realization.

In sum, flexible and expressive NLG is crucial for the robustness of the entire speech-based dialogue system by verbalizing what the system understood and what actions it performed as a consequence of this understanding. We find that overgeneration and ranking techniques allow us to model alignment and variation even in situations where no corpus data is available by using the discourse history as a 'corpus'.

Acknowledgments This work is supported by the US government's NIST Advanced Technology Program. Collaborating partners are CSLI, Robert Bosch Corporation, VW America, and SRI International. We thank the many people involved in this project, in particular Fuliang Weng and Heather Pon-Barry for developing the content optimization module; Annie Lien, Badri Raghunathan, Brian Lathrop, Fuliang Weng, Heather Pon-Barry, Jeff Russell, and Tobias Scheideck for performing the evaluations and compiling the results; Matthew Purver and Florin Ratiu for work on the CSLI dialogue manager. The content optimizer, knowledge manager, and the NLU module have been developed by the Bosch Research and Technology Center.

References

Carsten Brockmann, Amy Isard, Jon Oberlander, and Michael White. 2005. Modelling alignment for affective dialogue. In *Proc. of the UM'05 Workshop on Adapting the Interaction Style to Affective Factors*.

Robert Dale and Ehud Reiter. 1995. Computational Interpretations of the Gricean Maxims in the Generation of Referring Expressions. *Cognitive Science*, 19:233–263.

Martin Kay. 1996. Chart Generation. In *Proceedings of ACL-96*, pages 200–204.

Irene Langkilde. 2002. An Empirical Verification of Coverage and Correctness for a General-Purpose Sentence Generator. In *Proc. of INLG-02*.

Danilo Mirkovic and Lawrence Cavedon. 2005. Practical Plug-and-Play Dialogue Management. In *Proceedings of the 6th Meeting of the Pacific Association for Computational Linguistics (PACLING)*.

Sebastian Varges. 2002. Fluency and Completeness in Instance-based Natural Language Generation. In *Proc. of COLING-02*.

Sebastian Varges. 2005. Chart generation using production systems (short paper). In *Proc. of 10th European Workshop On Natural Language Generation*.

Fuliang Weng, L. Cavedon, B. Raghunathan, D. Mirkovic, H. Cheng, H. Schmidt, H. Bratt, R. Mishra, S. Peters, L. Zhao, S. Upson, E. Shriberg, and C. Bergmann. 2004. Developing a conversational dialogue system for cognitively overloaded users. In *Proceedings of the International Congress on Intelligent Transportation Systems (ICSLP)*.