A Computational Approach for Sentence Planning

Feng-Jen Yang
Computer Science Department, Prairie View A&M University, Texas, USA
fengjen_yang@pvamu.edu

ABSTRACT

Sentence planning is a critical processing closely related to the coherence and fluency of a natural language system. After a discourse is planned, there are still several open issues before surface sentence generation. By considering the great variety of human languages, a discourse should be realized by alternative sentential representations according to its context. This research was motivated by this concern and the result successfully justified the necessity of sentence planning. In this paper, a computational approach is presented to coherently realize a discourse topic by alternative sentence representations according to its discourse context. This is a critical quality issue from the design perspective of natural language systems.

1: INTRODUCTION

As modeled in most of the natural language systems, the processing of natural language generation is usually following the sequence of discourse planning, sentence planning and sentence generation. After the discourse planner selects a topic to be realized in natural language, the sentence planner narrows down the planning scope to an individual sentence and goes on arrange the prosodic sequence of each phrase in the sentence being planned. Finally, the sentence generator transforms each internal sentence representation into a grammatically correct and coherence sentence.

The simplest approach of sentence planning is using canned text by which generating a sentence is simply printing a predefined string without considering alternative representations and prosodic sequences. Planning sentences in this manner is of course simple and easy to implement in both single and multiple sentence generations, but tends to have repetitive sentences which sound unnatural and may frustrate its users. To mimic human language better, some more advanced approaches are usually considered, such as template-based planning, phrase-based planning and feature-based planning [1].

Template-based sentence planning is suitable for systems in which a sentence has to be generated several times with only a few parts alternated, such as generating weather forecast reports, medical reports and stock market reports. By leaving some parts of the sentence blank and to be filled according to discourse context, the sentence being generated can be less repetitive and more vivid than canned text.

Phrase-based sentence planning is a more advanced approach which follows the sequence of top-down grammatical analysis to replace each level of grammatical structure by specific phrasal patterns. A sample planning sequence is starting from replacing the entire sentence by a phrasal pattern consisting of a noun phrase structure and a verb phrase structure. Similarly, a noun phrase may be replaced by a specific article and a noun and a verb phrase may be replaced by a specific verb and an object, and so on. This process is continued until the entire sentence is realized.

Phrase-based planning can be very flexibly representing the variety of human languages, but, due to the difficulty of maintaining the inter-phrase relationships, it tends to have inappropriate phrase expansions unintentionally. This difficulty somehow limits its scope of planning.

The most advanced approach is feature-based sentence planning by which each sentence is planned by constructing a specific set of features where each feature is representing an alternative expression of a small part of the sentence. Internally, a sentence can be specified by features, such as asking something or informing something, using passive voice or active voice, and so on. The number of features specifying a sentence is up to the number of significant features which can adequately describe the sentence. A full sentence is incrementally planned by accumulating features until having enough features to realize a sentence.

The idea of feature-base planning is getting popular and attracting more and more researchers, but still facing the challenge of maintaining the complicated inter-feature relationships. For example, a small feature set representing a simple sentence may be a subset of a big feature set representing a compound sentence. A critical decision to be made here is whether to stop accumulating features and generate the simple sentence or keep accumulating features to generate the compound sentence. In the current natural language processing researches, this approach is only successfully applied in sing-sentence planning.

This paper describes a dialogue-based tutoring system which has been improved from purely canned sentences to feature-based sentence planning.

2: THE SYSTEM OVERVIEW

As a dialogue-based intelligent tutoring system, the CIRCSIM-Tutor tries to simulate human tutoring sessions in the domain of baroreceptor reflex. It has been tested to be effective and now being used as a class aid for first-year medical students at Rush Medical College in Chicago.

The baroreceptor reflex is the mechanism in charge of regulating blood pressure in the human body so that it will not go beyond the tolerable range. If something happens to change the blood pressure, such as a transfusion, hemorrhage or pacemaker malfunction, the baroreceptor reflex will attempt to regulate the blood pressure in a negative feedback manner so the blood pressure will go back to a stable state again.

While using this system the student is presented with a predefined perturbation and then is asked to predict the qualitative changes in seven physiological variables at three different chronological stages of the reflex cycle. These predictions are then used as the basis of a tutoring session to remediate any misconception that the student has revealed.

In order to simulate the dialogue of human tutors as much as possible and provide learners with a coherent and fluent natural language interface, this paper presents a lexicalization approach as a post process to refine our machine planned discourse. The discourse planner leaves a certain number of decisions open before surface sentence generation and I choose five lexical features as the first attempt to improve the quality of our machine dialogue. These features are chosen because they seem relatively manageable but particularly important in our domain.

The behavior of the baroreceptor reflex can be described by the qualitative influences among seven physiological variables over three stages. The seven core variables as they appear in the prediction table are Central Venous Pressure (CVP), Inotropic State (IS), Stroke Volume (SV), Heart Rate (HR), Cardiac Output (CO), Total Peripheral Resistance (TPR) and Mean Arterial Pressure (MAP). The three stages in the order of occurrence are the Direct Response (DR) Stage, which is the time immediately after the perturbation and before the reflex is activated, the Reflex Response (RR) Stage, when the changes caused by the baroreceptor reflex begin to take effect, and the Steady State (SS) Stage, the time after restabilization.

The causal relationships between these variables can be modeled by either direct or inverse qualitative influence among variables. With a direct influence, increasing the parameter on the cause side results in increasing the parameter on the effect side or decreasing parameter in the cause side results in decreasing the parameter on the effect side. For example, increasing the CO results in decreasing the CVP, but increasing the CO results in increasing the MAP.

It is possible that a parameter to have two determinants. In such cases, learners have to think about which determinant is stronger, since the result is based on qualitative changes. The change in the stronger determinant will dominate the total qualitative change, even if the other determinant has the opposite qualitative influences. For example, the SV has two determinants, the CVP and the IS, but the IS is stronger than the CVP. So if the CVP decreased but the IS increased, the increase in the IS is stronger than the decrease in the CVP and the SV will still increase.

3: THE OUTCOME

Like most natural language systems, the CIRCSIM-Tutor has a discourse planner to produce a discourse plan that specifies both the content and overall structure of a tutoring session. In terms of determining the deep structure, knowing the content and structure of a dialogue is enough and the discourse planner has been doing a good job. Nevertheless, to make a dialogue fluent and coherent, knowing only the deep structure is far from enough. There is still a considerable range of details to form the shallow structure and feed it to the surface sentence generator. The discourse planner leaves open a certain number of decisions about the surface form of the dialogue to be generated. Figure 1 is a sample dialogue while the system is using purely canned sentences. This dialogue reveals the lack of fluency and coherence in the system.

T1: By what mechanism is Inotropic State controlled?
S2: nervous system
T3: Correct, Inotropic State is controlled by the nervous system.
What is the value of Inotropic State in DR?
S4: decreased
T5: Nope, the value of Inotropic State is not decreased in DR.
Remember. Inotropic State is neurally controlled.
What is the value of Inotropic State in DR?
S6: not changed
T7: Correct, the value of Inotropic State is

Figure 1. A Canned Dialogue

The goal of this research is to make the machine dialogue fluent and coherent and I have decided to take an approach by refining its lexical usages. I, therefore, have some range of options in deciding which lexical features to work on. The following lexical features were chosen as the first attempt to make the dialogue in Figure 1 sound more natural and more fluent. To illustrate each lexical feature and its corresponding refinement, the related dialogue turns are chosen and improved step by step while a feature is discussed.

1. Polite Locutions

   The canned dialogue always asks questions directly, but human tutors tend to ask questions politely by using various polite locutions, such as ***Can you tell me …*** or ***Do you know …***, to make the asking sound polite. So, turn T1 can be improved to read:

   T1: ***Can you tell me*** by what mechanism Inotropic State is controlled?

2. Abbreviated Variable Names

   The canned dialogue always spells out variable names while asking questions and giving acknowledgments, but human tutors are much more likely to use the full name when the variable is first introduced and then abbreviate or pronominalize it later. So, turns T3 and T5 can be improved to read:

   T3: Correct, ***IS*** is controlled by the nervous system.
   What is the value of ***IS*** in DR?
   S4: decreased
   T5: Nope, the value of ***IS*** is not decreased in DR.
   Remember, ***IS*** is neurally controlled.
   What is the value of ***IS*** in DR?

3. Discourse Markers

   Using discourse markers can make clear the logical relationship between two sentences or the relationship of a sentence to the tutorial goal structure. In turns T3 and T5, the two consecutive utterances belong to different topics which can be refined by using discourse markers like ***so***, ***now***, ***but***, ***therefore,*** etc., to move from one discourse focus to another. So, turns T3 and T5 can be further improved to read:

   T3: Correct, IS is controlled by the nervous system.
   So, what is the value of IS in DR?
   T5: Nope, the value of IS is not decreased in DR.
   Remember, IS is neurally controlled.
   Now, what is the value of IS in DR?

4. Acknowledgments

   In turns T3 and T5, the acknowledgments are both explicit and content-based which make the sentences sound redundant and stilted. In human dialogues, acknowledgments following student answers are often ***reduced to a single word***, ***appended to the next sentence***, or ***even omitted entirely***. Our human tutoring transcripts show that usually correct answers are not repeated, but wrong answers are repeated in a sense of pointing out the student's misconception. So, turns T3 and T5 can be even further improved to read:

   T3: ***Right***.
   So, what is the value of IS in DR?
   S4: decreased
   T5: ***No, the value of IS is not decreased in DR***.
   Remember, IS is neurally controlled.
   Now, what is the value of IS in DR?

   A special phenomenon of acknowledging a student's answer is that human tutors tend to acknowledge the student's finally correct answer more strongly than usual, especially when the student has made some mistakes and finally got the correct answer. So, turn T7 can improved to read:

   T7: ***Very good***.

5. Pronouns

   In turn T5, the intended variable name has been mentioned in the previous turn. In this case, human tutors tend to use the pronoun it to refer to the variable previously mentioned and stay in the same discourse focus. So, the turn T5 can be improved to read:

   T5: No, IS is not decreased in DR.
   Remember, ***it*** is neurally controlled.
   Now, what is the value of IS in DR?

Addressing only these five lexical features, the dialogue in Figure 1 can be transformed into Figure 2. The necessity of sentence planning is self justified by comparing the quality difference these two dialogues.

T1: Can you tell me by what mechanism Inotropic State is controlled?
S2: nervous system
T3: Right.
So, what is the value of IS in DR?
S4: decreased
T5: No, IS is not decreased in DR.
Remember, it is neurally controlled.
Now, what is the value of IS in DR?
S6: not changed
T7: Very good.

Figure 2. A Sentence Planned Dialogue

Generally speaking, these refinements are instances of lexical selection. This is also an illustration of the fact that lexical variation is not random but planned and purposeful.

4: MINING LEXICAL RULES

Many methods have been proposed for analyzing the local discourse context and lexical usages. The most popular method is annotating a corpus of the type of discourse that a system wishes to generate. A set of general instructions for annotating discourse segments and identifying the purposes of discourse segments was proposed by Nakatani and Grosz [2]. By investigating the relationship between reference and segmentation, Passonneau designed a protocol for coding discourse referential noun phrases and their antecedents [3]. Other researchers such as Allen and Core [4], Nakatani and Traum [5] and Brennan and Clark [6] have also suggested methods for exploring lexical issues.

Our discourse modeling is based on a fundamental discourse theory stating that a hierarchical organization of discourse around fixed schemata can guarantee good coherence and proper content selection [7]. When the same idea is applied to the CIRCSIM-Tutor domain, a set of hierarchical tutoring schemata has been discovered to model the discourse of tutoring sessions performed by our domain experts and their students [8]. Based on these schemata, I started thinking about the approaches of mining lexical features.

My lexical analysis is based on the concept that a good discourse theory must be able to account for the ordering of major discourse constituents and predict the surface linguistic phenomena that depend on structural aspects of discourse [9]. In other words, by knowing the structure of the discourse in progress, the system should be able to predict their corresponding surface linguistic usages. I, thus, focused my analysis on discovering the relationship between a discourse structure and its corresponding surface language usage. Another useful idea comes from Passonneau's protocol, especially for the problem of finding the inference relationships between different discourse segments [3]. The draft of DAMSL [4], which uses a backward looking function to capture how the current utterance relates to its antecedent, is also a helpful reference.

The lexical analysis described here is focused on the semantic and pragmatic relationships among the tutoring schemata as well as looking for special phenomena of lexical usage in the dialogue context. To this end, it is more useful to have a method that shows discourse structure and lexical usage at the same time. I have developed a representation for lexical usage that allows the researcher to visualize lexical research. This method begins by representing the hierarchical tutoring schemata as tables and then maps the lexical items of interest onto those table entries according to their original positions in the schemata. In this manner, we can visualize both the discourse structure and lexical usage simultaneously.

Figure 3 illustrates the visualization of the variable descriptions used by our domain experts while tutoring the variable TPR in a transcript of a life tutoring session performed by our domain experts and their students. The discourse structure of this dialogue is modeled by a schema called ***T-corrects-variable*** which is realized by two subschemata, ***T-introduces-variable*** and ***T-tutors-variables***, and then the ***T-tutors-variable*** is realized by ***T-does-neural-DLR***. The ***T-does-neural-DLR*** is further realized by ***T-tutors-mechanism***, ***T-tutors-DR-info***, and ***T-tutors-value***, and so on. This process keeps going until each of them is finally realized by a surface utterance.

In this example, I used typography to indicate the lexical features that interest me. The variable ***TPR*** is marked, along with the anaphoric references to it. The lexical phenomena here are: "the tutor first uses the abbreviated variable name ***TPR*** to bring up this variable to teach. In the immediately following topic, the tutor uses the pronoun ***it*** to refer to the previous mentioned TPR. After that the tutor goes on to convey some other related explanations and in the final topic the tutor uses the abbreviated variable name ***TPR*** again to bring back the discourse focus."

When these phenomena applied to sentence planning, a lexical rule is derived as: "a discourse planned using the schema ***T-corrects-variable*** will always have the variable introduced in the first topic. So, in the second topic the machine tutor can always use a pronoun to refer to the same variable and maintain the same discourse focus. Also, in the sense of making a conclusion, it is appropriate to use abbreviated variable name to bring back focus in the last topic."

The purpose of visualization is to gather together all the instances of lexical phenomena and the contexts in which they occur. I look at two types of context, the surrounding text and the position within the tutorial dialogue schema. Similar analyses are performed on the usages of discourse markers and acknowledgements. As a result, a set of lexical rules has been formulated, which can be used to as guidelines to prepar future sets for sentence generations [10].

T-corrects-variable var=TPR			
T-introduces-variable	**T-tutors-variable**		
T-informs	**T-does-neural-DLR**		
	T-tutors-mechanism	**T-tutors-DR-info**	**T-tutors-value**
	T-elicits	**T-informs**	**T-elicits**
T: Now how about ***TPR***? S: …	T: By what mechanism will ***it*** increase? S: …	…	T: So what do you think about ***TPR*** now? S: …

Figure 3. Visualization of Variable Descriptions

5: THE SENTENCE PLANNING

Sentence planning is a processing after discourse planning and before surface sentence generation. To form a pipeline from discourse planning to sentence generation as suggested by Reiter and Dale [11], the interfaces have to be clearly defined.

5.1: THE DISCOURSE IN PROGRESS

The discourse planner is using a set of hierarchical schemata as plan operators and the operators currently in use are stored in a working storage. By consulting the working storage the sentence planning module can have a copy of the discourse in progress and apply lexical rules to construct feature sets for sentence generation.

The following lisp program is a template to get a copy of the current discourse in progress. After executing these codes the variables ***w-stage***, ***w-topic***, ***w-primitive*** will be holding the current tutoring ***stage***, ***topic*** and ***primitive***, respectively.

```
(setq w-stage
  (get-value-from-KB
    '(w-stage-is ?x)
  )
)
(setq w-topic
  (get-value-from-KB
    '(w-topic-is ?x)
   )
)
(setq w-primitive
  (get-value-from-KB
    '(w-primitive-is ?x)
  )
)
...
and so on.
```

5.2: PREPARING FEATURE SETS

The sentence generator takes a feature set and generating a sentence accordingly. For example, feeding the feature set ***"((primitive informs) (topic mechanism) (stage dr) (var ((var-name CC))"*** to the sentence generator will have the sentence ***"CC is under neural control."*** generated.

The major steps and their corresponding lisp codes to prepare the feature set for a sentence generation are summarized as follows:

1. Initially, the feature set is empty.
```
(let
  ((features ())
)
```
2. The feature set could be multi-level. So the program goes on to call ***subfeature constructors*** to construct subfeatures for all discourse operators currently in use, such as ***(primitive-feature w-primitive)***, ***(topic-feature w-topic)***, ***(stage-feature w-stage)***, ... etc., and append them to the overall feature set.
```
(setq features
  (append features
    (primitive-feature w-primitive)
   )
)
(setq features
  (append features
    (topic-feature w-topic)
  )
)
(setq features
  (append features
    (stage-feature w-stage)
  )
)
...
and so on.
```
3. Each subfeature is then constructed according to each discourse plan operator currently in use. For example, since there are only two possible values for the primitive operator, the primitive subfeature can only be either ***(primitive elicits)*** or ***(primitive informs)***.
```
(defun primitive-feature (value)
  (cond
    ((equal value elicits)
      '((primitive elicits)))
    ((equal value informs)
      '((primitive informs)))
  )
)
```
Other subfeature constructors are implemented in the same manner.
4. After all subfeatures are constructed and appended to the overall feature set, the entire feature set is ready for a sentence generation.

6: CONCLUSION

Sentence planning is discourse-based, grammatical-based, pragmatics-based, and semantics-based. The detail of this multidimensional processing is still evolving. Many natural language research groups have found that a certain number of natural language generation issues are beyond the consideration of discourse planning and surface generation, but they are nonetheless important in building high-quality language generation systems. A certain level of cognitive related issues has to be taken into consideration. In this research, I focus on the task of lexical refinement to produce a more detailed sentence specification for the surface generator to generate more coherent and natural sounding sentences. This is a critical problem in the system and I have taken the first step toward it.

ACKNOWLEDGE

This work was supported by the Cognitive Science Program, Office of Naval Research under Grant No. N00014-94-1-0338 to Illinois Institute of Technology. The content does not reflect the position or policy of the government and no official endorsement should be inferred.

REFERENCES

[1] Hovy, E. 1997. Language Generation: Overview, In Varile, G., Zampolli, A., Cole, R., Mariani, J., Uszkoreit, H., Zaenen, A., and Zue, V. (editors), *Survey of the State of the Art in Human Language Technology*, Chapter 4.1, Cambridge, England: Cambridge University Press.

[2] Nakatani, C., Grosz, B., Ahn, D., and Hirschberg, J. 1995. *Instructions for Annotating Discourses*, Technical Report Number TR-21-95, Center for Research in Computing Technology, Harvard University, Cambridge, MA.

[3] Passonneau, R. 1994. *Protocol for Coding Discourse Referential Noun Phrases and Their Antecedents*, Revised November 1994, Technical Report, CARD Project, Department of Computer Science, Columbia University, New York City, NY.

[4] Allen, J., and Core, M. 1997. Draft of DAMSL: Dialog Act Markup in Several Layers, *The Multiparty Discourse Group at the Discourse Research Initiative (DRI) Meeting*, Schloss Dagstuhl, Not Paged.

[5] Nakatani, C., and Traum, D. 1998. Draft: Discourse Structure Coding Manual, *The 3rd Discourse Resource Initiative (DRI) Meeting*, Chiba, Japan, Not Paged.

[6] Brennan, S.E., and Clark, H.H. 1996. Conceptual Pacts and Lexical Choice in Conversation, *Journal of Experimental Psychology: Learning, Memory, and Cognition*, Volume 22, Pages 1482 - 1493.

[7] McKeown, K. 1985. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*, Cambridge, UK: Cambridge University Press.

[8] Kim, J., 1998. *The SGML Markup Manual for Circsim-Tutor*. Technical Report, Computer Science Department, Illinois Institute of Technology, Chicago, IL.

[9] Reichman, R. 1985. *Getting Computers to Talk like You and Me*, Chapter 5, Cambridge, MA: The MIT Press.

[10] Yang, F. 2005. A Computational Lexicalization Approach. *Proceedings of the 2nd International Workshop in Natural Language Understanding and Cognitive Science (NLUCS 2005)*. Miami, FL, Pages 128-145.

[11] Reiter, E. and Dale, R. 2000. *Building Natural Language Generation System: Studies in Natural Language Processing*, Chapter 3, Cambridge, UK: Cambridge University Press.