A Computational Lexicalization Approach

Feng-Jen Yang

Feng-Jen Yang, Computer Science Department, Prairie View A&M University
Prairie View, Texas 77446, USA

Abstract. Fine-Grained lexicalization has been treated as a post process to refine the machine planned discourse and make the machine generated language more coherent and more fluent. Without this process, a system can still generate comprehensible languages but may sound unnatural and sometimes frustrate its users. To this end, generating coherent and natural sounding language is a major concern in any natural language system. In this paper, a lexicalization approach is presented to refine the machine generated language.

1 Introduction

An obvious difference between a natural language system and an information management system is the user interface. If a user asks an information management system the same question twice, it is very likely that the system will respond with the same answer twice, but a natural language system hardly has this kind of dialogue. Instead of repeating the same answer, a natural language system tends to adapt answers according to the user's understanding and try to reach the dialogue goal [7]. The invention of natural language systems is somehow motivated by the desire to reform the interaction between human and computer.

As a tutoring system with a natural language interface, 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

Yang F. (2005).
A Computational Lexicalization Approach.
In *Proceedings of the 2nd International Workshop on Natural Language Understanding and Cognitive Science*, pages 128-146
DOI: 10.5220/0002558301280146

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.

2 Domain Knowledge

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 for 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 Why Lexicalization?

To benefit from a natural language interface, the tutoring system must be provided with the properties that make human natural language so effective [8]. With this concern in mind, CIRCSIM-Tutor tries to imitate the human tutor's language as much as possible.

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.

3.1 A Coarse-Grained Lexicalization Example

Figure 1 is an example dialogue before lexicalization, which reveals the lack of fluency and coherence in our machine dialogues. For example, in the second utterance of T3, the Inotropic State should be pronominalized in a sense of maintaining the same discourse focus. Also, the content based acknowledgements in T3 and T5 make the machine dialogue stilted.

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 unchanged in DR.

Fig. 1. An Example Dialogue before Lexicalization

3.2 A Fine-Grained Lexicalization Example

One of the important areas of research in computational discourse is finding out what information is contained in the sequence of utterances but goes beyond the meaning of individual utterances themselves [4]. To this end, having better lexical usages are absolutely essential and critical. The goal of this research is to make the machine dialogue fluent and coherent. I, therefore, have some range of options in deciding which lexical features to work on. The following features were chosen as the first attempt of lexicalization, because they seem relatively manageable and particularly important to make the dialogue in Figure 1 sound more natural and more fluent [15].

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 machine tutor always asks questions directly, but human tutors tend to ask questions politely instead of directly. They frequently use various politeness locutions, such as ***Can you tell me …*** or ***Do you know …***, before the question. So, turn T1 can be improved with a softener:
   T1: ***Can you tell me*** by what mechanism Inotropic State is controlled?

2. Abbreviated Variable Names
   The machine tutor 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, but the machine tutor does not use discourse markers, while human tutors tend to use 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. Whether an acknowledgment can be reduced and attached to the next sentence depends on the syntax of the next sentence, the relation of the next sentence to the answer being acknowledged, and whether the answer is correct or not. 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 the 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?

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.

Since the system is using schemata as planning operators, an efficient way of learning the rules for lexical selection is by searching for examples of lexical usage in transcripts marked up with tutoring schemata. I search for instances of the same schema expressed in different ways. After further in-depth analysis of these instances, I have established rules as a better guidance for lexical selection.

Addressing only the five lexical features discussed above, the dialogue in Figure 1 can be transformed into Figure 2.

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.

Fig. 2. An Example Dialogue after Lexicalization

The necessity of lexicalization can be justified by comparing the quality difference of machine generated dialogues with and without lexicalization.

4 Discourse Modeling

One of the major problems addressed in discourse research is:

How does an utterance’s context affect the meaning of the individual utterance or part of it [4]?

That is why a major result of most discourse analysis is dividing a discourse into discourse segments. The boundaries of segments have to be determined in a manner much like phrases group into sentences and sentences group into paragraphs, and so on. The meaning of a segment encompasses more than the meaning of individual parts [4]. While segmenting the discourse, the language behavior is also modeled.

Many methods have been proposed for analyzing the local discourse context. The most popular method is annotating a corpus of the type of discourse that you wish to generate. A set of general instructions for annotating discourse segments and identifying the purposes of discourse segments was proposed by [9]. By investigating the relationship between reference and segmentation, Passonneau [11] designed a protocol for coding discourse referential noun phrases and their antecedents. Other researchers such as Allen and Core [1], Nakatani and Traum [10] and Brennan and Clark [2] have also suggested methods for exploring lexical issues.

4.1 Discourse Coherence

A very important research resource in the CIRCSIM-Tutor project is a set of tutoring transcripts numbered from K1 to K76. These sessions were carried out in a keyboard-to-keyboard manner by our domain experts and their first-year students in physiology. This research, like most of our earlier work is based on the study and analysis of these transcripts.

Our discourse analysis is based on a fundamental discourse theory saying that a hierarchical organization of discourse around fixed schemata can guarantee good coherence and proper content selection [6]. 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 [5]. Based on these schemata, I started thinking about the approaches to refine our machine dialogue.

If we model the behavior of lexicalization in terms of discourse trees, it deals with integrating the leaf nodes into a coherent dialogue. This integration is related both to discourse planning and to surface sentence generation. So, a central problem with the lexicalization is how to make a smooth connection among the semantic representation, the pragmatic information, and the surface linguistic phenomena. In other words, the lexicalization has to consider the alternatives in terms of representing the content of the participants' utterances, performing the dialogue acts, and generating the surface language. These alternatives not only provide a certain level of implementation flexibility, but also introduce the possibility of optimization at some level.

Since the system is now using schemata to plan the discourse, having a coherent movement of discourse focus is no longer a problem. The remaining work is to produce a fine-grained lexicalization. This takes more in-depth of lexical analysis.

5 Lexical Analysis

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 [12]. In other words, by knowing the structure of the discourse in progress, we 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 [11]. The draft of DAMSL [1], 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.

5.1 Visualization of Lexical Usage

In order to predict the surface linguistic phenomena from the structural aspects of discourse, it is more useful to have a method that shows discourse structure and lexical usage at the same time. This will help the analysis to take both issues into consideration. I have developed a new 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 the session K12. 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.

<table>
<tr><th colspan="4">T-corrects-variable var=TPR</th></tr>
<tr><th>T-introduces-variable</th><th colspan="3">T-tutors-variable</th></tr>
<tr><th rowspan="3">T-informs</th><th colspan="3">T-does-neural-DLR</th></tr>
<tr><th>T-tutors-mechanism</th><th>T-tutors-DR-info</th><th>T-tutors-value</th></tr>
<tr><th>T-elicits</th><th>T-informs</th><th>T-elicits</th></tr>
<tr><td>T: Now how about TPR?
S: …</td><td>T: By what mechanism will it increase?
S: …</td><td>…</td><td>T: So what do you think about TPR now?
S: …</td></tr>
</table>

Fig. 3. Visualization of Variable Descriptions

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 lexicalization:

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.

Figure 4 is designed to help us visualize the usage of discourse markers while tutoring the variable TPR in the session K10.

<table>
<tr><th colspan="4">T-corrects-variable var=TPR</th></tr>
<tr><th>T-introduces-variable</th><th colspan="3">T-tutors-variable</th></tr>
<tr><th rowspan="3">T-informs</th><th colspan="3">T-does-neural-DLR</th></tr>
<tr><th>T-tutors-mechanism</th><th>T-tutors-DR-info</th><th>T-tutors-value</th></tr>
<tr><th>T-elicits</th><th>T-informs</th><th>T-elicits</th></tr>
<tr><td>T: Take the last one first.</td><td>T: Can you tell me how TPR is controlled?
S: …</td><td>T: And the predictions that you are making are for the period before any neural changes take place.</td><td>T: So what do you think about TPR now?
S: …</td></tr>
</table>

Fig. 4. Visualization of Discourse Marker Usage

The lexical phenomena in this example are:

The tutor uses the discourse marker ***And*** to move from one topic to a semantically continuous topic and uses the discourse marker ***So*** to mark the final topic as an appropriate conclusion.

When these phenomena applied to lexicalization:

A discourse planned according to the schema ***T-does-neural-DLR*** will always have the first two topics semantically continuous. So, it will be always appropriate to use the discourse marker ***And*** to connect these two topics. Also in the last topic the tutor has to make a conclusion and the discourse marker ***So*** is a good way to make this conclusion.

Similarly, Figure 5 is a visualization of the way acknowledgments are used while tutoring TPR in the session K48. The lexical phenomena in this example are:

For the first two questions, the tutor gives a hint by asking some background knowledge and moving toward the final question. Fortunately, the student answers these two hints right. So the tutor uses the explicit word ***Right*** to accept these answers. Finally, the student figured out the correct answer and the tutor acknowledged it in a stronger manner to encourage the student and said ***Great***.

When these phenomena applied to lexicalization:

A discourse planned according to the schema ***T-does-neural-DLR*** will always have some digression before the student figures out the final correct answer. So, in the last topic, the machine tutor can acknowledge the student's answer more strongly than usual to encourage the student.

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: You predicted that TPR would increase.	T: What mechanism does this? S: Autonomic nervous system. T: ***Right***.	T: And during DR what changes in ANS activity occur? S: none. T: ***Right***.	T: So do you want to change your prediction: S: Yes, TPR has no change. T: ***Great!***

Fig. 5. Visualization of the Choice of Acknowledgments

5.2 Result of Lexical Analysis

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. Ultimately, I have found rules, as addressed in Appendix A, which can be used to as guidelines towards a finer-grained lexicalization in the CIRCSIM-Tutor domain.

6 Implementation

Lexicalization 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 [13], the interfaces have to be clearly defined.

6.1 The Interface between Discourse Planning and Lexicalization

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 lexicalization module can have a copy of the discourse in progress and apply lexical rules accordingly. Figure 6 is the lisp program template to get a copy of the current discourse. 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.
```

Fig. 6. Retrieve the Discourse in Progress

6.2 The Interface between Lexicalization and Sentence Generation

The sentence generator is using a template generation approach which 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 a feature set for 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.

7 Conclusion

The idea of lexicalization is not well-studied in natural language processing. Part of the reason is that a fine-grained lexicalization is related to something beyond sentence interpretation. The intentions of speakers and the understanding of listeners are the major factors dominate the evolving discourse and lexical usage.
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 text

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 dialogue specification for the surface sentence generator to generate more coherent and natural sounding sentences. This is a critical problem and I have taken the first step toward it.

Acknowledgement

This work was partially supported by the Cognitive Science Program, Office of Naval Research under Grant 00014-00-1-0660 to Stanford University as well as Grants N00014-94-1-0338 and N00014-02-1-0442 to Illinois Institute of Technology. The content does not reflect the position of policy of the government and no official endorsement should be inferred. Personal communications with Professor Martha Evens at Illinois Institute of Technology has been of great assistance to this research.

References

1. 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.
2. 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.
3. Grice, H.P. 1969, Utterer's Meaning and Intentions. *Journal of Philosophical Review*, Volume 68, Number 2, Pages 147 - 177.
4. Grosz, B. 1997. Discourse and Dialogue: 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 6, Cambridge, UK: Cambridge University Press.
5. Kim, J., 1998. *The SGML Markup Manual for CIRCSIM-Tutor*. Technical Report, Computer Science Department, Illinois Institute of Technology, Chicago, IL.
6. McKeown, K. 1985. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*, Cambridge, UK: Cambridge University Press.
7. McRoy, S, Ali, S., Restificar, A., and Channarukul, S. 1999. Building Intelligent Dialogue Systems, *Journal of Intelligence*, Volume 10, Number 1, Pages 14 - 23.
8. Moser, M., and Moore, J.D. 1995. Investigating Cue Selection and Placement in Tutorial Discourse, *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, Cambridge, MA, Pages 130 - 135.
9. 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.
10. Nakatani, C., and Traum, D. 1998. Draft: Discourse Structure Coding Manual, *The 3rd Discourse Resource Initiative (DRI) Meeting*, Chiba, Japan, Not Paged.
11. 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.
12. Reichman R. 1985. *Getting Computers to Talk like You and Me*, Chapter 5, Cambridge, MA: The MIT Press.

13. Reiter, E. and Dale, R. 2000. *Building Natural Language Generation System: Studies in Natural Language Processing*, Chapter 3, Cambridge, UK: Cambridge University Press.
14. Schiffrin, D. 1987. *Discourse Markers*, Cambridge, UK: Cambridge University Press.
15. Yang, F., Kim, J., Glass, M., & Evens, M. 2000. Lexical usage in the tutoring schema of CIRCSIM-Tutor: Analysis of variable references and discourse markers, *Proceedings of the 5th International Conference on Human Interaction with Complex System*, Urbana, IL, Pages 27 - 31.

Appendix A Lexical Rules

Based on the analysis of lexical phenomena in the tutoring schemata, I have developed lexical rules for ***polite locutions***, ***variable references***, ***discourse markers***, and ***acknowledgment choices***. These rules along with real life tutoring examples marked with SGML tags are listed and discussed in following sections.

A.1 Lexical Rules for Polite Locutions

Rule 1: Within the first topic of ***T-does-neural-DLR***, the tutor uses the locutions ***Can you tell me*** or ***Do you know*** to bring up a question politely.

Example:

```
<T-does-neural-DLR>
  <T-tutors-mechanism>
K10-tu-29-4: Can you tell me how TPR is controlled?
...
  </T-tutors-mechanism>
...
</T-does-neural-DLR>
```

A.2 Lexical Rules for Variable Descriptions

Rule 1: Use abbreviated variable names

Case 1: Within the topic ***T-introduces-variable***, the tutor uses the ***abbreviated name*** to introduce a new variable.

Example:

```
<T-introduces-variable>
K11-tu-41-1: You only have TPR left.
</T-introduces-variable>
```

Case 2: Within the topic immediately following ***T-introduces-variable***, the tutor keeps using the ***abbreviated name*** of the variable to maintain the same discourse focus.

Example:

```
<T-introduces-variable>
K11-tu-41-1: You only have TPR left.
</T-introduces-variable>
<T-tutors-variable>
```

```
    <T-does-neural-DLR>
      <T-tutors-mechanism>
  K11-tu-49-3: How is TPR controlled?
  ...
      </T-tutors-mechanism>
    </T-does-neural-DLR>
  </T-tutors-variable>
```

Case 3: Within the last topic of ***T-tutors-variable***, the tutor uses the ***abbreviated name*** of the variable to end digressions and bring back the discourse focus.

Example:

```
  <T-tutors-variable>
  ...
    <T-does-neural-DLR>
  K10-tu-29-4: Can you tell me how TPR is controlled?
  ...
  K10-tu-31-2: And the predictions that you are making are for
  the period before any neural changes take place.
      <T-tutors-value>
  K10-tu-31-3: So what about TPR?
  ...
      </T-tutors-value>
    </T-does-neural-DLR>
  </T-tutors-variable>
```

Rule 2: Use pronominal descriptions

Case 1: Within the topic immediately following ***T-introduces-variable***, the tutor uses ***it*** to refer to the variable and maintain the same discourse focus.

Example:

```
  <T-introduces-variable>
  K12-tu-31-1: Now how about TPR?
  </T-introduces-variable>
  ...
  <T-tutors-variable>
    <T-does-neural-DLR>
      <T-tutors-mechanism>
  K12-tu-33-1: By what mechanism will it increase?
  ...
      </T-tutors-mechanism>
    </T-does-neural-DLR>
  </T-tutors-variable>
```

Case 2: Within the topic immediately following ***T-introduces-variable***, the tutor uses ***this*** to refer to a proposition and maintain the same discourse focus.

Example:

```
  <T-tutors-variable>
  ...
    <T-explores-anomaly>
      <T-presents-anomaly>
  K26-tu-76-2: So, co decreases even though sv increases.
      </T-presents-anomaly>
      <T-tutors-anomaly>
  K26-tu-76-3: How can you explain this?
```

```
      </T-tutors-anomaly>
    </T-explores-anomaly>
  </T-tutors-variable>
```

Rule 3: Use definite descriptions

Case 1: Within the topic of ***T-introduces-variable***, the tutor uses ***the last one*** or ***this issue*** to introduce the variable.

Example:

```
  K10-tu-29-2: Let's take a look at some of your predictions.
  <T-introduces-variable>
      K10-tu-29-3: Take the last one first.
  </T-introduces-variable>
```

Case 2: Within the topic immediately following ***T-introduces-variable***, the tutor uses ***that prediction*** to refer to both the variable and its change and maintain the same discourse focus.

Example:

```
  <T-introduces-variable>
  K48-tu-44-3: you predicted that TPR would increase.
  </T-introduces-variable>
  ...
  <T-tutors-variable>
    <T-does-neural-DLR>
      <T-tutors-mechanism>
  K48-tu-44-4: Can you explain how you arrived at that
  prediction?
  ...
      </T-tutors-mechanism>
    </T-does-neural-DLR>
  </T-tutors-variable>
```

Case 3: Within the last topic of ***T-tutors-variable***, the tutor uses ***your prediction*** to end digressions and bring back the discourse focus.

Example:

```
  <T-tutors-variable>
    <T-does-neural-DLR>
  K48-tu-44-4: Can you explain how you arrived at that
  prediction?
  ...
  K48-tu-48-2: and during DR what changes in ANS activity occur?
  ...
      <T-tutors-value>
  K48-tu-50-1: So do you want to change your prediction?
      </T-tutors-value>
    </T-does-neural-DLR>
  </T-tutors-variable>
```

A.3 Lexical Rules for Discourse Markers

Rule 1: Use so and now

Case 1: ***so*** and ***now*** are used in ***T-introduces-variable*** to initiate a discourse focus. This is similar to behavior observed by Schiffrin [14].

Example:

<T-introduces-variable>
K11-tu-53-2: ***So*** let me ask you, are there any other of these variables that are primarily under neural control?
</T-introduces-variable>

Case 2: ***so*** and ***now*** are used to conclude ***T-tutors-variable***. This is similar to the idea of marking results discussed by Schiffrin [14].

Example:

<T-tutors-variable>
<T-does-neural-DLR>
...
<T-tutors-value>
K10-tu-31-3: ***So*** what about TPR?
...
</T-tutors-value>
</T-does-neural-DLR>
</T-tutors-variable>

Rule 2: Use ***first*** in ***T-introduces-variable*** to introduce the first topic of the first variable being tutored.

Example:

<T-introduces-variable>
K13-tu-37-3: ***First***, what parameter determines the value of rap?
</T-introduces-variable>

Rule 3: Use ***but*** in ***T-presents-contradiction*** to contrast two ideas.

Example:

<T-shows-contradiction>
<T-presents-contradiction>
K10-tu-41-2: You predicted that it would go up.
...
K10-tu-43-1: ***But*** remember that we're dealing with the period before there can be any neural changes.
</T-presents-contradiction>
</T-shows-contradiction>

Rule 4: Use ***and*** to initiate a semantically continuous topic.

Example:

<T-does-neural-DLR>
<T-tutors-mechanism>
K10-tu-29-4: Can you tell me how TPR is controlled?
...
</T-tutors-mechanism>
<T-tutors-DR-info>
K10-tu-31-2: ***And*** the predictions that you are making are for the period before any neural changes take place.
</T-tutors-DR-info>
...
<T-does-neural-DLR>

Rule 5: Use ***therefore*** to summarize ***T-tutors-via-deeper-concepts***.

Example:

```
<T-tutors-via-deeper-concepts>
  <T-tutors-determinant>
K27-tu-52-1: If I have a single blood vessel, what parameter
most strongly determines its resistance to flow?
...
    <T-moves-to-previous-concepts>
      <T-tutors-determinant>
K27-tu-54-1: And physiologically, what determines the
diameter of the blood vessels?
      </T-tutors-determinant>
    </T-moves-to-previous-concepts>
  </T-tutors-determinant>
  <T-tutors-determinant>
K27-tu-56-2: Therefore, what determines TPR?
  </T-tutors-determinant>
</T-tutors-via-deeper-concepts>
```

A.4 Lexica Rules for Acknowledgments

Rule 1: Use a negative acknowledgment such as ***no*** or ***not quite*** to reject the student's first wrong answer.

Example:

```
K12-tu-31-1: Now how about TPR?
<T-elicits>
K12-tu-33-1: By what mechanism will it increase?
  <S-ans catg=incorrect>
K12-st-34-1: If you increase pressure will you momentarily
increase resistance
  </S-ans>
  <T-ack type=negative>
K12-tu-35-1: No.
  </T-ack>
</T-elicits>
```

Rule 2: Use a partial acknowledgment, such as ***partly correct***, to partially accept the student's answer.

Example:

```
<T-elicits>
K47-tu-56-5: Can you tell me what you think that IS means?
<S-ans catg=near-miss>
K47-st-57-1: the contractility of the heart caused by preload
and sympathetic stimulation
</S-ans >
<T-ack type= partially-correct >
K47-tu-58-1: Partly correct.
</T-ack >
</T-elicits>
```

Rule 3: Use of positive acknowledgments

Case 1: Use ***yes*** or ***right*** to accept the student's first correct answer.

Example:

```
K10-tu-29-2: Let's take a look at some of your predictions.
K10-tu-29-3: Take the last one first.
<T-elicits>
K10-tu-29-4: Can you tell me how TPR is controlled?
  <S-ans catg=correct>
K10-st-30-1: Autonomic nervous system
  </S-ans>
  <T-ack type=positive>
K10-tu-31-1: Yes.
  </T-ack>
</T-elicits>
```

Case 2: Use a strong positive acknowledgment, such as ***good***, ***very good***, ***absolutely***, ***exactly***, or ***great*** to accept the student's final correct answer, especially when the student had some difficulty in reaching this goal.

Example:

```
<T-elicits>
K27-tu-72-2: How is this possible?
  <S-ans catg=correct>
K27-st-73-1: Hr is down more than sv is up
  </S-ans>
  <T-ack type=positive>
K27-tu-74-1: Very good.
  </T-ack>
</T-elicits>
```

Rule 4: Acknowledgment is ***omitted*** in some special situations, such as when the tutor is identifying the student's problem, or the student has a near miss answer.

Case 1: the tutor tries to identify the student's problem without giving any acknowledgment.

Example:

```
<T-diagnoses-errors>
  <T-identifies-problem>
    <T-elicits>
K27-tu-50-2: Why do you think that TPR will decrease?
      <S-ans catg=incorrect>
K27-st-51-1: Since HR decreased, CO will decrease and the
direct response would be decreased TPR.
      </S-ans>
    </T-elicits>
  </T-identifies-problem>
</T-diagnoses-errors>
K27-tu-52-1: If I have a single blood vessel, what parameter
most strongly determines its resistance to flow?
(Acknowledgment omitted)
```

Case 2: The tutor does not give any acknowledgment when the student gives a ***near-miss*** answer, but tries other methods to guide the student toward the correct answer.

Example:

```
<T-tutors-via-determinants>
  <T-tutors-determinant>
    <T-elicits>
K25-tu-48-3: What parameter determines rap?
      <S-ans catg=near-miss>
K25-st-49-1: Central venous pressure.
      </S-ans>
    </T-elicits>
    <T-moves-toward-PT method-type=inner>
      <T-tutors-determinant>
        <T-elicits>
K25-tu-50-1: And what determines cvp?
(Acknowledgment omitted)
          <S-ans catg=correct>
K25-st-51-1: Blood volume and "compliance" of the Venous side
of the circ.
          </S-ans>
          <T-ack type=positive>
K25-tu-52-1: Right.
          </T-ack>
        </T-elicits>
    <T-moves-toward-PT>
  </T-tutors-determinant>
</T- tutors-via-determinants>
```