

WHAT NOT TO SAY

Jan Fornell
Department of Linguistics & Phonetics
Lund University
Helgonabacken 12, Lund, Sweden

ABSTRACT

A problem with most text production and language generation systems is that they tend to become rather verbose. This may be due to neglection of the pragmatic factors involved in communication. In this paper, a text production system, COMMENTATOR, is described and taken as a starting point for a more general discussion of some problems in Computational Pragmatics. A new line of research is suggested, based on the concept of unification.

I COMMENTATOR

A. The original model

1. General purpose

The original version of Commentator was written in BASIC on a small micro computer. It was intended as a generator of text (rather than just sentences), but has in fact proved quite useful, in a somewhat more general sense, as a generator of linguistic problems, and is often thought of as a "linguistic research tool".

The idea was to create a model that worked at all levels, from "raw data" like perceptions and knowledge, via syntactic, semantic and pragmatic components to coherent text or speech, in order to be able to study the various levels and the interaction between them at the same time. This means that the model is very narrow and "vertical", rather than like most other computational models, which are usually characterized by huge databases at a single level of representation.

2. The model

The system dynamically describes the movements and locations of a few objects on the computer screen. (In one version: two persons, called Adam and Eve, moving around in a yard with a gate and a tree. In another version, some ships outside a harbour). The comments are presented in Swedish or English in a written and a spoken version simultaneously (using a VOTRAX speech

synthesis device). No real perceptive mechanism (such as a video camera) is included in the system, (instead it is fed the successive coordinates of the moving objects) but otherwise all the other abovementioned components are present, to some extent.

For both practical and intuitive reasons the system is "pragmatically deterministic" in some sense. By this I mean that a certain state of affairs is investigated only if it might lead to an expressible comment. For every change of the scene, potentially relevant and commentable topics are selected from a question menu. If something actually has happened (i.e. a change of state [1] has occurred), a syntactic rule is selected and appropriate words and phrases are put in. A choice is made between pronouns and other nounphrases, depending on the previous sentences. If a change of focus has occurred, contrastive stress is added to the new focus. Some "discourse connectives" like *också* (also/too) and *heller* (neither) are also added. There are apparently some more or less obligatory contexts for this, namely when all parts (predicates and arguments) of two sentences are equal except for one. For example

"Adam is approaching the gate."
"Eve is also approaching it."

(predicates equal, but subjects different)

"John hit Mary."
"He kicked her too."

(subjects and objects equal, but different predicates), etc. Stating the respective second sentences of the examples above without the also/too sounds highly unnatural. This is however only part of the truth (see below).

Note that all selections of relevant topics and syntactic forms are made at an abstract level. Once words have begun being inserted, the sentence will be expressed, and it is never the case that a sentence is constructed, but not expressed. Neither are words first put in, and then deleted. This is in contrast with many other text production systems, where a range of sentences are constructed, and then compared to find the "best" way of expressing the proposition. That might be a possible approach when writing a (single) text, such as an instruction manual, or a paper like this, but it seems unsuitable for dynamic text production in a changing environment like Commentator's.

B. A new model

A new version is currently being implemented in Prolog on a VAX11/730, avoiding many of the drawbacks and limitations of the BASIC model. It is highly modular, and can easily be expanded in any given direction. It does not yet include any speech synthesis mechanism, but plans are being made to connect the system to the quite sophisticated ILS program package available at the department of linguistics. On the other hand, it does include some interactive components, and some facilities for (simple) machine translation within the specified domains, using Prolog as an intermediary level of representation.

The major aim, however, is not to re-implement a slightly more sophisticated version of the original Commentator, which is basically a monologue generator, but instead to develop a new, highly interactive model, nick-named CONVERSATOR, in order to study the properties of human discourse. What will be described in the following, is mostly the original Commentator, though.

II COMPUTATIONAL PRAGMATICS

A. Relevance Strategies in Commentator

The previous presentation of Commentator of course raises some questions, such as "What is a relevant topic?" It is a well known fact, that for most text production systems it is a major problem to restrict the computer output - to get the computer to shut up, as it were, and avoid stating the obvious. In many cases this problem is not solved at all, and the system goes on to become quite verbose. On the other hand, Commentator was developed with this in mind.

1. Changes

A major strategy has been to only comment on changes [2]. Thus, for example, if Commentator notes that the object called Adam is approaching the object called the gate (where approach is defined as something like "moving in the direction of the goal, with diminishing distance" - this is not obvious, but perhaps a problem of pattern recognition rather than semantics), the system will say something like

(1) "Adam is approaching the gate".

Then, if in the next few scenes he's still approaching the gate, nothing more need to be said about it. Only when something new happens, a comment will be generated, such as if Adam reaches the gate, which is what one might expect him to do sooner or later, if (1) is to be at all appropriate. Or if Adam suddenly reverses his direction, a slightly more drastic comment might

be generated, such as

(2) "Now he's moving away from it".

Note however, that the Commentator can only observe Adam's behaviour and make guesses about his intentions. Since he is not Adam himself, he can never know what Adam's real intentions are. He can never say what Adam is in fact doing, only what he thinks Adam is doing, and any presuppositions or implicatures conveyed are only those of his beliefs. Thus, uttering (1) somehow implicates that the Commentator believes that Adam is approaching the gate in order to reach it, but not that Adam is in fact doing so. This might be quite important.

2. Nearness

Another criterion for relevance is nearness. It seems reasonable to talk about objects in relation to other objects close by [3], rather than to objects further away. For instance, if Adam is close to the gate, but the tree is on the other side of the yard, it would probably make more sense to say (3) than (4), even though they may be equally true.

(3) Adam is approaching the gate.

(4) Adam is moving away from the tree.

All of this, of course, presupposes that it is sensible to talk about these things at all, and this is not obvious. What is a text generation system supposed to do, really?

B. Why talk?

Expert systems require some kind of text generation module to be able to present output in a comprehensible way. This means that the input to the system (some set of data) is fairly well-known, as well as the desired format of the output. But this means that the quality of the output can only be measured against how well it meets the pre-determined standards. There is obviously much more to human communication than that. I believe that the serious limitations and unnaturalness of existing text generation systems (whether they are included in an expert system or not. There aren't really many of the latter type.) cannot be overcome, unless a certain important question is asked, namely "Why ever say anything at all?"

Two different dimensions can be recognized. One is prompted vs spontaneous speech, and the other is the informative content.

At one end of the information scale is talk that contains almost no information at all, such as most talk about the weather. This is usually a very ritualized behaviour [4], and is quite different from the exchange of data, which characterizes most interactions with computers and would be the other end of the scale.

Aside from the abovementioned kind of social interaction, it seems that one talks when one is in possession of some information, and believes that the listener-to-be is interested in this information. The most obvious case is when a question has been asked, or the speaker otherwise has been prompted. In fact, this is the only case that text generation systems ever seem to take care of. Expert systems speak only when spoken to. The Commentator is made to talk about what's happening, assuming that someone is listening, and interested in what it says. But for a conversating system this is not enough. The properties of spontaneous speech has to be investigated, in order to address questions like "When does one volunteer information?", "When does one initiate a conversation?" and "When does one change topic?" It will involve quite a lot of knowledge about the potential listener and the world in general, which might be extremely hard to implement, but which I believe is necessary anyway, for other reasons as well (see below).

C. Natural Language-Understanding

It has been pointed out (Green (1983), and references cited therein) that "communication is not usefully thought of as a matter of decoding someone's encryption of their thoughts, but is better considered as a matter of guessing at what someone has in mind, on the basis of clues afforded by the way that person says what s/he says". Still, much work in linguistics relies on the assumption that the meaning of a sentence can be identified with its truth-conditions, and that it can somehow be calculated from the meaning of its parts [5], where the meanings of the words themselves usually is left entirely untreated. But again, this is a far cry from what a speaker can be said to mean by uttering a sentence [6].

While some interesting work has been done trying to recognize Gricean conventional implicatures and presuppositions in a computational, model-theoretical framework (Gunji, 1981), the particularized conversational implicatures were left aside, and for a good reason too. With the kind of approaches used hitherto, they seem entirely untreatable.

Instead, I would say that understanding language is very much a creative ability. To understand what someone means by uttering some sentence, is to construct a context where the utterance fits in. This involves not only the linguistic context (what has been said before) and the extra-linguistic context (the speech situation), but also the listener's knowledge about the speaker and the world in general. It also involves recognizing that every utterance is made for a purpose. The speaker says what s/he does rather than something else. The used mode of expression (e.g. syntactic construction) was selected, rather than some other. In this sense, what is not said is as important as what is actually said. Note that I said "a context" rather

than "the context": one can do no more than guess what the speaker had in mind, since it strictly is impossible to know.

D. Text Generation Revisited

A text generation system would also need the same kind of creative ability, in order to have some conception of how the listener will interpret the message. This will of course affect how the message is put forward. One does not say what one believes the listener already knows, or is uninterested in, and on the other hand, one does not use words or syntactic constructions that one believes the listener is unfamiliar with. Since speakers generally will tend to avoid stating the obvious, and at the same time say as much as possible with as few words as possible, conversational implicatures will be the rule, rather than the exception.

For example, using words like "too" and "also" means that the current sentence is to be connected to something previous. Only in a few, very obvious cases (such as the Commentator examples above) will the "previous" sentence actually have been stated. In most cases, the speaker will rely on the listener's ability to construct that sentence (or rather context) for himself.

III CONCLUSIONS

Does this paint too grim a picture of the future for text generation and natural language understanding systems? I don't think so. I have just wanted to point out that unless quite a lot of information about the world is included, and a suitable Context Creating Mechanism is constructed, these systems will never rise above the phrase-book level, and any questions of "naturalness" will be more or less irrelevant, since what is discussed is something highly artificial, namely a "speaker" with the grammar and dictionary of an adult, but no knowledge of the world whatsoever.

How is this Creative Mechanism supposed to work? Well, that is the question that I intend to explore. The concept of unification seems very promising [7]. Unification is currently used in several syntactic theories for the handling of features, but I can see no reason why it shouldn't be useful in handling semantics, discourse structure and the connections with world-knowledge as well. Any suggestions would be greatly appreciated.

NOTES

[1] In this sense, something like "X is approaching Y" is as much a state as "X is in front of Y".

[2] This is apart from an initial description of the scene for a listener who can't see it for himself, or is otherwise unfamiliar with it. Cf a radio sports commentator, who would hardly describe what a tennis court looks like, or the general rules of the game, but will probably say something about who is playing, the weather and other conditions, etc.

[3] Though closeness is of course not just a physical property. Two people in love might be said to be very close, even though they are physically far apart. This is something, however, that the Commentator would have to know, since it's usually not immediately observable.

[4] For instance, if someone says "Nice weather today, isn't it?", you're supposed to answer "Yes" no matter what you really think about the weather. Not much information can be said to be exchanged.

[5] This is of course valuable in the sense that it says that "John hit Bill" means that somebody called John did something called hitting to somebody called Bill, rather than vice versa.

[6] And, importantly, it is the speaker who means something, and not the words used.

[7] Unification is an operation a bit like putting together two pieces of a jigsaw puzzle. They can be fitted together (unified) if they have something in common (some edge), and are then, for all practical purposes, moved around as a single, slightly larger piece. For an excellent introduction to unification and its linguistic applications see Karttunen (1984). Unification is also very much at the heart of Prolog.

REFERENCES

Fornell, Jan (1983): "Commentator - ett mikrodatorbaserat forskningsredskap för lingvister", Praktisk lingvistik 8, Dept of Linguistics, Lund University.

Green, Georgia M. (1983): Some Remarks on How Words Mean, Indiana University Linguistics Club, Bloomington, Indiana.

Gunji, Takao (1981): Toward a Computational Theory of Pragmatics, Indiana University Linguistics Club, Bloomington, Indiana.

Karttunen, Lauri (1984): "Features and Values", in this volume?

Sigurd, Bengt (1983): "Commentator: A Computer Model of Verbal Production", Linguistics 20-9/10.