A CHART PARSER TO ANALYZE LARGE MEDICAL CORPORA

P.O. El Guedj[1] and P. Nugues[2]

(1) Université de Caen, Computer Science Laboratory,
6, boulevard du Maréchal Juin, F–14050 Caen, France

(2) Institut des Sciences de la Matière et du Rayonnement, Computer Science Laboratory,
6, boulevard du Maréchal Juin, F–14050 Caen, France

Email: {elguedj, pnugues}@L2I.ISMRA.FR

***Abstract*—In this paper we describe a natural language parser for large medical corpora. Sentence parsing is a necessary step to build a sentence representation and to support a wide-coverage semantic interpretation. When applied to limited domains, a good syntax coverage can be obtained from Phrase-Structure rules. Large medical corpora show a strong variability in word and phrase order that requires more specific parsing strategies. We describe a parser based on Chart techniques. It parses constituents from left to right as they appear in a sentence. It enables the incremental partial parsing of words and phrases coming from a speech recognition input. We report here first results obtained from a large corpus of cancer treatment reports.**

I. INTRODUCTION

Text dictation systems enter a commercial age in the medical area. These systems [1, 2]—available in several European languages—enable the dictation of reports directly to a computer. Reports are immediately created in a computer-readable format and can be saved in the patient record of a medical database. They are then available for subsequent re-reading, modification, etc.

Current speech recognition systems are using phoneme and word statistical models [3]. These models are referred to as n-grams grammars—bigrams and trigrams being the most current. A n-gram model considers a sequence of *n* words and yields a statistical ranking for the next word to come knowing the $n - 1$ first words. Words' statistics are obtained from text corpora and trained on huge quantity of data. The n-gram grammars are combined with the results of a word recognition device. They attempt to fit their probabilistic models to what is actually spoken and decoded by the speech recognition device. They produce very good recognition rates provided the environment is not too noisy and the user cooperative.

An other analysis must be undertaken to understand what has been uttered: a question, a statement, a command, etc. A parser produces a tagging of all the words and phrases (constituents) of the sentence with part-of-speech and syntactic category labels. Parsing is a first step to elaborate the representation of a sentence's meaning [4, 5]. Subsequently, a semantic interpretation will enable to analyze texts or to implement interactive speech recognition that could answer questions.

II. PHRASE-STRUCTURE GRAMMARS

Many parsers rely on phrase-structure grammars to build syntactic representations of a sentence. These grammars use:

- parts-of-speech, such as determiner (DT) or noun (NN), to tag words,
- constituents, such as noun phrase (NP) or verb phrase (VP), to annotate phrases, and
- rules, to rewrite a left-hand side constituent into right-hand side constituents.

Rules describe constituent structures such as the sentence (S) structure: S ∅ NP VP, which consists here of a noun phrase and a verb phrase, and such as the noun phrase structure: NP ∅ DT NN, which consists here of a determiner and a noun. Text sentences are matched to the grammar rules using a parser. It results in a parse tree—the sequence of rules that have been applied—and in the labeling of constituents and words.

Phrase-structure grammars yield good results with limited text corpora and perform better on certain languages than on others. Languages here may mean English or French but also Medical English or Legal French. They are more difficult to implement on large corpora. The quality of the results depends much on the variability of word order. Medical text corpora present longer sentences than ordinary conversations and, in consequence, show more complicated syntactic structures and a greater number—greater variability—of possible phrase combinations. For these reasons, it is necessary to write recursive phrase-structure rules and to use a parser, such as a chart parser, able to deal with them.

III. THE CORPUS

The corpus collects text of reports on cancer treatment from the Centre François Baclesse, which is the anti cancer center of the region of Lower Normandy. The corpus represents all the records from year 1992 which are still active in 1994. All the reports were dictated by one of the 40 hospital's physicians using a dictation machine. They were subsequently transcripted by a medical secretary using a word processor and filed in a computer-readable format.

All the reports are free texts *i.e.* not constrained, except the header which must identify the physician, the patient, the date, etc. The corpus size totals approximately 180,000 words. The total number of different words is approximately

10,000. The average length of a sentence is approximately 8 words with a maximum of around 80 words. Texts are essentially medical descriptions. They are quite variable in length and style: some are telegraphic, others are more elaborate.

IV. A PARSING ALGORITHM

The algorithm is based on the Active Chart Parser [6] with features [7]. In consequence, it uses phrase-structure rules and allows recursive constituent definitions with part-of-speech subcategories. Recursive definitions are very useful to simplify the writing of intricate sentence structures—there are several sentences of more than 50 words.

The parser processes computer-readable texts but we modified it to deal more specifically with spoken sentences. It can operate either in a top-down or a bottom-up mode. It accepts words sequentially and parses phrases from left to right until the sentence is complete.

While parsing, the parser detects an error as soon as it occurs. It can then reject the last input word when the current sentence is no longer grammatically correct—i.e. when no syntactic rule can be applied to match the incoming word. Since many speech recognition modules propose a word list representing the N-Best hypotheses, the parser can discard the faulty word and select the second better word from this list. It can then try a new parsing.

The parser is able to process sentences lacking a few words—“holes”—by guessing the location and the category of these missing words. Notably, it can predict, in top-down mode, the categories of the next word likely to be pronounced by the speaker. The maximum number of missing words in a sentence and the maximum number of consecutive missing words is parametered when launching the parser.

The parsing results in several parse trees corresponding to all the different parsing assumptions. Arcs are labeled with part-of-speech or constituents tags. We used classical French part-of-speech tags which are similar to the English ones. On the other hand, we defined more specific types of constituents with a semantic relation. Notably, we use approximately 30 types of complements classified according to their head preposition.

V. IMPLEMENTATION AND PRELIMINARY RESULTS

The parsing algorithm has been implemented in C++. It is built with an object-oriented design which was inspired by [8]. A PC Windows 3.1 version runs into an interactive environment to facilitate debugging and to help with the lexical and syntactical category assignment. When included, “hole” processing significantly increase processing time. Results are presented for 3 sentences of variable length parsed on a PC-486 running at 66 MHz with “hole” processing turned off.

Sentence	Number of words	Parse time in seconds
Sentence 1	6	3
Sentence 2	16	5
Sentence 3	20	14

VI. CONCLUSION AND PERSPECTIVES

An ongoing effort concerns the flexibility of the parsing strategy. We are adding characteristics from dependency grammars [9]. It will enable the parser to build the complete sentence representation by the gradual attachment of the set of the parsed phrases. We will also include statistical tagging [3] characteristics to help disambiguate sentences and to speed up parsing. Finally, we plan to interface the parser with a speech dictation system such as the IBM Speech Server.

ACKNOWLEDGMENT

We would to thank Pr. Héron, Director of Centre François Baclesse, Bernard Vaudron, Information Systems Manager, and Danièle Mauger for help and support, and for providing us with the corpus.

REFERENCES

[1] R. Kurzweil, *The Age of Intelligent Machines*, MIT Press, 1990.

[2] S. Jovanovic, *Le radiologue dicte à l’ordinateur*, AIX, N° 4, 1994. (In French).

[3], K.W. Church and R.L. Mercer, *Introduction to the Special Issue on Computational Linguistics Using Large Corpora, Computational Linguistics*, Vol. 19:1, pp. 1-24, 1993.

[4] P. Nugues, P.O. ElGuedj, F. Cazenave, and B. de Ferrière, Issues in the Design of a Voice Man Machine Dialogue System Generating Written Medical Reports, *Proc. of the Annual Int. Conf. IEEE/EMBS*, Vol. 14:3, pp. 842-844, 1992.

[5] P. Nugues, C. Godéreaux, P.O. ElGuedj, and F. Cazenave, Question Answering in an Oral Dialogue System, *Proc. of the Annual Int. Conf. IEEE/EMBS*, Vol. 15:2, pp. 590-591, 1993.

[6] J. Earley, An efficient context-free parsing algorithm, *Communications of the ACM*, 13:2, pp. 94-102, 1970.

[7] G. Gazdar and C. Mellish, *Natural Language Processing in Prolog, An Introduction to Computational Linguistics*, Addison-Wesley, 1989.

[8] D. Perelman-Hall, A Natural Solution, *BYTE*, Vol. 17:2, pp. 237-244, 1992.

[9], M. Covington, Parsing discontinuous constituents in dependency grammars, *Computational Linguistics*, Vol. 16:4, 1990.