[*International Conference on Machine Translation of Languages and Applied Language Analysis*, National Physical Laboratory, Teddington, UK, 5-8 September 1961]

THE IDENTIFICATION OF NESTED STRUCTURES IN PREDICTIVE SYNTACTIC ANALYSIS*

by

MURRAY E. SHERRY

(Air Force Cambridge Research Laboratories, Air Force Research Division (ARDC), United States Air Force
Laurence G. Hanscom Field, Bedford, Massachusetts)

1. INTRODUCTION

THE automatic syntactic analysis of natural languages has been the prime field of endeavor for investigators in the field of automatic language translation for the past several years, and also has received the attention of investigators with such other interests as automatic computer coding systems and information retrieval. In automatic language translation it has been assumed that a syntactic analysis of the source language would, by necessity, precede a semantic analysis. This order of analysis is based on the apparent difficulty of performing a semantic analysis as opposed to a syntactic analysis, it being generally acknowledged that the semantic problems are overwhelming compared to the syntactic difficulties. Thus, to achieve a satisfactory automatic translation, it seems essential that a syntactic analysis be sufficiently powerful to determine adequately the structure of sentences, distinguish sentences from nonsentences, and provide guarantees that sentences have been analyzed correctly.

Where a grammatical structure can be analyzed in terms of continuous constituents, all of the various proposed automatic syntactic analysis methods produce good results. However, many languages cannot be described entirely in terms of continuous constituents, and the difficulty in analyzing such languages is invariably the handling of discontinuous constituents. Some of these languages, including both Russian and English, have a property that can be utilized to simplify the analysis problem.

In English, if a sentence is interrupted by a phrase or a clause, the embedded phrase or clause will be completed before the main clause is resumed. This embedded phrase or clause is considered to be nested within

*This work was carried out at Harvard University with the assistance of the staff of the Computation Laboratory and was supported in part by the National Science Foundation.

the main clause. Thus, the clause "who came to dinner" is nested in the sentence: "The man who came to dinner ate heartily.", whereas the unnested string of words: "The man who came ate heartily to dinner", is a questionable sentence at best. Another structure, the phrase "to dinner" is nested within the subordinate clause. A level of nesting or depth of nesting can be assigned to every phrase and clause in a sentence. Thus, "The man ate heartily" is at the first level, "who came to dinner" is at the second level, and "to dinner" is at the third and deepest level. Both the phrase "to dinner* and the clause "who came to dinner" can be analyzed merely in terms of continuous constituents. However, a more powerful scheme is necessary to analyze structures such as the discontinuous clause "The man ... ate heartily".

The concept of nesting has received the attention of several investigators recently. Alt[1] has discussed the problem of assigning numerical values to clauses and phrases within a sentence. Yngve[2] and Sager[3] have also used the nesting concept when discussing, respectively, the synthesis and analysis of English sentences. Sager uses the terminology of "depth of parenthesization" instead of "depth of nesting" since she conceives of an approach whereby a pair of parentheses is placed around every identifiable nested structure.

The work on predictive syntactic analysis grew out of studies on a syntactic analysis technique by Rhodes[4], the formalization of the syntax of the Lukasiewicz parenthesis-free notation by Burks, Warren and Wright, on the linguistic model of Chomsky[6], and on Oettinger's theory of syntactic analysis of certain artificial languages. A comprehensive description of the work on predictive syntactic analysis that was subsequently carried out at Harvard University can be obtained from additional reports by Sherry[8,9], Bossert[10] and Isenberg[11].

In this paper the techniques for identifying the nested phrase and clause structures in a predictive syntactic analysis program for the Russian language will be discussed. The mechanism by which nested structures are identified is considered after an introductory section, in which the main aspects of predictive syntactic analysis are outlined. Several Russian sentences are included in Section 3 for illustrative purposes. To explain several concepts that are common both to Russian and to English, English examples have also been included.

2. PREDICTIVE SYNTACTIC ANALYSIS

The method of predictive syntactic analysis is based on the premise that

a Russian sentence can be scanned from left to right, and that at any point in this process it is possible both to determine the syntactic structure of the word under scrutinization based on predictions made during the analysis of the word to its left, and to predict the syntactic structures which will be encountered to the right of the word being scrutinized.

The predictions are stored in a prediction pool, a linear array that behaves somewhat like a pushdown store. New predictions are always entered at the top of the prediction pool, and the predictions are tested starting at the top of the pool and proceeding downward.

Many of the predictions used in the experimental program are named for classical grammatical terms, such as subject prediction. All of these classifications are explicitly defined within the context of the experimental program. These definitions need not coincide with the classical grammatical definitions, but they resemble the classical definitions closely.

The process of predictive syntactic analysis consists of two cycles, a testing cycle and a predicting cycle.

During the testing cycle the predictions are tested against the information about the arguments of words that are obtainable from a dictionary. Since the lexical properties of words do not always define a unique argument, a set of alternative arguments must be considered. Thus, "waters" has two alternative arguments, /noun, plural/ and /verb, third person, singular, present tense/.

Every time that an alternative argument can fulfill a prediction, an intersection takes place. The preferred argument is the alternative argument of the first intersection in a test sequence. A prediction is fulfilled when it results in the first intersection. A fulfilled prediction is removed or wiped from the prediction pool. In a test sequence all the alternative arguments of a word are tested against all the predictions in the pool in their respective orders, such that each prediction, in turn, is tested against the set of alternative arguments. All intersections occurring subsequent to the first intersection are listed in hindsight for future reference, while the preferred argument is recorded as the temporary analysis for the given word.

After the testing cycle has been completed, the predicting cycle starts. This consists of updating the prediction pool (1) by wiping fulfilled predictions, (2) by modifying predictions already in the pool, and (3) by adding new predictions to the top of the pool as dictated by the word just analyzed. In this manner, a noun assigned the preferred argument of subject

would cause (1) the subject prediction to be wiped from the pool, (2) the predicate head prediction to be modified, so that only a predicate that agrees with the subject in person, number and gender can be accepted, and (3) two new predictions, a compound subject and a noun complement, to be entered at the top of the new pool. The compound subject is predicted because the noun was selected as the subject; the noun complement, a prediction of a genitive noun phrase, is predicted by every noun regardless of its preferred argument.

A number is assigned to every prediction in the pool. This number acts as a reference to the preferred argument that initiated the given prediction. In this manner, when the sentence is analyzed, not only is a preferred argument assigned to every word, but also a linkage to the word initiating the prediction is established. To continue with the same example, if the word following the noun subject is a genitive noun, the text number of the noun subject is attached to the preferred argument of the genitive noun. In this manner the noun complement can be identified as a dependent on the subject.

There are a number of words or other forms that either can never be predicted or can be predicted only sometimes. Examples of such forms are adverbs, prepositions, and commas. Adverbs occur both to the left and to the right of the words that they modify. In a left to right pass, adverbs can only be predicted if they occur to the right of the words they modify. Thus, an adverb preceding an adjective or a verb cannot be tied to the dominant structure since the dominant structure has not yet been identified. Likewise, if a prepositional phrase does not follow immediately after the word it modifies, it is a difficult matter at best to predict the phrase. A comma is even worse in this respect since it can be found after almost any word in a sentence. However, it is true that if two commas are used to isolate some structure in a sentence, the second of the commas may be predicted by the first.

When a word that cannot be predicted is encountered during a testing cycle, it must nevertheless be accepted in some sense, subject to later revision. Since there is no prediction in the pool, no finite number can be assigned the unpredicted word to indicate the linkage. Rather, an "infinite number" is assigned to the unpredicted word, and in the terminology of predictive syntactic analysis, the word is "accepted by infinity".

It is necessary to indicate the distinction between the infinity classification and the arbitrary choice classification, the only other non-grammatlcally oriented classification in predictive analysis. A word, such as a noun, that does not fulfill any prediction during a testing cycle is

automatically assigned to the arbitrary choice classification. This classification, by definition, excludes all words that can be accepted by infinity. In the present program it is hypothesized that all nouns should be predictable whereas all prepositions need not be. Thus, although a preposition, if otherwise unpredicted, can come from infinity, a noun that is otherwise unpredicted is labelled an arbitrary choice.

One of the requirements for the identification and analysis of a complete sentence is that every word in the sentence fulfill a prediction. Thus, a completely analyzed sentence can contain words fulfilled by infinity, but it cannot contain any words which have been labelled arbitrary choice.

If the analysis proceeds merely as described, the size of the prediction pool will expand linearly with the number of analyzed words in a sentence. As each word in a sentence is analyzed, one prediction is fulfilled and subsequently wiped from the prediction pool. However, after each analysis new predictions are added to the pool, and on the average, more than two new predictions are added for each analyzed word. Thus, the prediction pool can grow to enormous proportions, especially if an unusually long or complex sentence is being analyzed.

It is known that when a prediction is made, it can be fulfilled only within a certain span of words. For example, if a verb is expected to follow the English word "to", the verb must be found immediately after "to" (excluding split infinitives). Otherwise, "to" is a preposition and not the head of a verb infinitive. In a similar vein, it is possible that a prediction made early in the analysis of a sentence cannot be fulfilled until much later in the sentence. Thus, in the sentence: "The man who came to dinner ate heartily", the predicate of the main clause cannot be fulfilled while the subordinate clause, "who came to dinner", is being analyzed.

Since extraneous predictions only increase the possibility of error by increasing the number of possible intersections, it becomes obvious that the burden of predictive syntactic analysis is to accomplish the following goals: (1) provide a prediction for every grammatical structure that might occur, based on an a priori expectation about the structure of sentences in general and on the analyzed words; (2) wipe all predictions that remain unfulfilled after it is known that they no longer can be fulfilled; and (3) identify the predictions that temporarily cannot be fulfilled due to the structure under analysis.

3. SENTINELS IN THE PREDICTION POOL

Consider the set of predictions in ***fig. 1 as*** an initial set of predictions to be placed in the prediction pool before the analysis of a sentence is started. These predictions are based on the hypothesis that a sentence can either start with the main clause, a subordinate clause, or a phrase introduced by a gerund. A further hypothesis made for this example is that every clause contains an explicit subject and an explicit predicate. The first prediction takes care of a sentence that starts with a gerund introducing a phrase such as: Держа шляпу в руке, он начал ГОВОРИЬ. The next five predictions are utilized if the sentence starts with a subordinate clause, and the remaining four predictions refer to the main clause of the sentence. The subject and predicate predictions are entered into the pool twice, since if the sentence starts with a subordinate clause, the second set of predictions will remain unfulfilled until the main clause is identified. With the sentence: Когда она ушла, он сел на стул, она and ушла, respectively, fulfill the top subject and predicate predictions, leaving the second subject and predicate pair to be fulfilled by он and сел.

This example shows how quickly the number of unfulfilled predictions in in the pool can multiply. If a sentence being analyzed starts with the main clause, the first six predictions can never be fulfilled. A mechanism is needed to wipe such unneeded predictions from the pool.

The end wipe, a sentinel, is inserted into the prediction pool to separate the predictions representing the different identified nested structures of the sentence under analysis (***fig. 2).*** (It is a moot question whether or not such a sentinel can be considered a prediction.) The end wipe is always placed below all the predictions in the pool that represent a nested structure. In that manner, it is tested after the other predictions that represent the nested structure. If there has been no previous intersection before an end wipe is tested, all the predictions located above the end wipe, as well as the sentinel itself, are wiped from the pool. If there has been a previous intersection, no action takes place, and the predictions following the end wipe are tested for hindsight in the usual manner. It is assumed in making a scan of the prediction pool that, if none of the predictions of the nested structure being analyzed can be fulfilled, the nested structure is complete. If the end wipe is tested with no record of a previous intersection, the action taken is tantamount to the hypothesis that the analysis of the nest is complete and the testing process has to revert to the structure represented by the predictions located below the sentinel.

Two end wipe sentinels have been added to the prediction pool of ***fig. 1*** to give the pool of ***fig. 2.*** If a sentence starts with a subordinate clause that is headed by a conjunction or a relative pronoun, there is no need to keep the gerund prediction, since it can no longer be fulfilled. Likewise, if the sentence starts with the main clause, the first six initial predictions can be wiped from the pool. In the event that a sentence is started with either a phrase or subordinate clause then, when the main clause is reached (and the entire structure of the phrase or clause has been identified), none of the predictions located above the second end wipe should be fulfilled. That end wipe, together with all the remaining predictions located above it, can then be wiped from the pool.

Since the test to determine whether the predictions should be wiped is the previous occurrence of an intersection, when a sentinel is encountered, the infinity test should be performed before determining whether or not an Intersection has occurred. In this manner, words from infinity will always be assigned to the deepest existing nested structure that has been only partially completed. The result is somewhat similar in Yngve's sentence synthesis method where prepositions are basically kept at the same level of nesting. However, it should be kept in mind that Yngve is concerned about the depth of nesting of individual words, whereas in predictive analysis this concept is applied only to entire phrases and clauses as a whole.

While the subordinate clause in the sentence: Стул, на котором он седел, был сломан, is being analyzed, by the nesting hypothesis none of the predictions of the main clause, remaining in the pool after стул has been analyzed, can be fulfilled. Thus, there is no point in looking for the predicate of the clause whose subject is стул while на коором он седел is being analyzed. Since the scanning of the prediction pool always continues after an end wipe is located and the appropriate action taken, the sentinel does not help distinguish between the predictions of continuous constituents and the predictions of discontinuous constituents. That is, whereas the end wipe eliminates predictions once they can no longer be fulfilled, this sentinel is of no help in inhibiting the testing of other predictions, (such as the predicate prediction in the example) which cannot be fulfilled in any given scan of pool.

Likewise, in the previously mentioned sentence: Когда она ушла, он сел на стул, after когда has been identified as the relative conjunction, the prediction pool of ***fig.*** 2 looks as in ***fig. 3.*** With the end wipe as a sentinel, она can fulfil both subject predictions and ушла can fulfil both predicate predictions. The first intersection in each case results in the preferred argument and the second intersection is placed in

hindsight. Since the information in hindsight will be used for error detection and error correction, any intersections which are known to be meaningless on a given scan of the pool should not be recorded. The second intersection of both the subject and predicate predictions fall into this category since once когда has been identified, the entire subordinate clause must be analyzed before the analysis returns to the main clause.

Since only one clause can be analyzed at a time, a second sentinel, the comma end wipe, has been adopted to isolate the predictions referring to different clauses (*fig. 4*). This sentinel is inserted underneath the other predictions for a clause. The sentinel is placed in the pool both when the pool is initialized at the start of a new sentence and whenever predictions for a new clause are made. The name of this sentinel implies its origin. It has been hypothesized that subordinate clauses, as well as certain types of phrases, are isolated by commas from the rest of the sentence in which they occur; and the comma preferred argument makes the necessary predictions for a new clause or phrase.

In Russian writing this rule is followed fairly strictly. However, sentences do occur in which the commas in question have been omitted. Whether or not such sentences are "good Russian" is an academic question since their solution will be necessary for an effective syntactic analysis scheme. When such sentences are handled by a syntactic analysis, then the comma end wipe will have to be introduced when the new phrase or clause is detected. At that time perhaps a change of name of the sentinel might be in order!

When a sentence is being analyzed, there are times when it is known that a deepest nested phrase or clause is only partially identified and that the next word must belong to the same structure. At other times there are clues that perhaps the deepest nested phrase or clause has been analyzed in its entirety and either that a new phrase or clause might start or that the analysis might return to a less deeply nested grammatical structure that was only partially identified before the deepest nested phrase or clause started.

Thus, the comma end wipe must operate in two modes, which have been named the continue clause mode and the end clause mode. In the continue clause mode, the comma end wipe inhibits the testing of the predictions located below it in the pool. In that mode the prediction pool is scanned as if there were no predictions located below the sentinel. (However, the predictions below the comma end wipe are kept when the pool is updated.) In the end clause mode, the sentinel behaves as an ordinary end wipe and the predictions below the comma end wipe are scanned in the normal manner.

When ОНА from the sentence: Когда она ушла, он сел на стул, is being analyzed *(see fig. 4)*, the comma end wipe should be in the continue clause mode since there is no question that the subordinate clause is currently being identified. However, when the word after the comma, он is being analyzed, the sentinel should be in the end clause mode. At this time the analysis might return to the main clause (as it does in the example), might continue with another deeper nested structure, or might even remain in the same clause. The latter two possibilities are illustrated, respectively, by the following Russian sentences: Когда она ушла, одетая в новой шубе, он сел на стул, and: Когда она стояла, ходила или бегала, ее нога болела.

Since the basic hypothesis for this sentinel is the assumption that in Russian commas separate certain phrases and clauses from the rest of a sentence, to help the analysis of these phrases and clauses, it is natural for the comma end wipe to be in the continue clause mode at all times except immediately following the recognition of a comma. The word following a comma should be tested when the comma end wipe is in the end clause mode. Then, the analysis of the word following a comma can return to any previous depth of nesting. After that word is tested, all remaining comma end wipe sentinels in the pool are returned to the continue clause mode.

To switch from the continue mode to the end clause mode, the comma predicts another sentinel, the comma end wipe activator, that is placed at the top of the new prediction pool. Thus, when the alternative arguments of the word following the comma are tested against the pool, this sentinel is the first one encountered. The comma end wipe activator sentinel refers to a subroutine which looks for comma end wipe sentinels in the pool and switches them from the continue clause mode to the end clause mode. After that action the comma end wipe activator sentinel is wiped and the other predictions in the pool are scanned in the normal manner. The change back to the continue clause mode can be carried out within the skeleton of the program. Every time the pool is updated (that is, after the analysis of every word) the mode can be automatically restored to the continue clause mode.

4. SUMMARY

In the present experimental predictive syntactic analysis program, a total of six sentinels are used to help analyze Russian sentences. However, the three mentioned in the last section are a representative subset that carry out the three functions required of sentinels, namely:

(1) the isolation of the predictions in the pool representative of different nested structures that have been partially identified;

(2) the wiping of predictions that no longer can be fulfilled;

(3) the modification of sentinels (or predictions) in the pool.

Another role for these sentinels is now under investigation. Several error detection devices have been introduced into the experimental program. Instead of waiting until an entire sentence is analyzed to look for errors, it seems possible to carry out the error detection in conjunction with the various sentinels present in the pool.

Prediction
Gerund
Relative Conjunction
Relative Pronoun
Subject
Predicate
Subject
Predicate
End of Sentence

A Set of Initial Predictions for the Analysis of a Russian Sentence

Fig. 1

Prediction
Gerund
End Wipe
Relative Conjunction
Relative Pronoun
Subject
Predicate
End Wipe
Subject
Predicate
End of Sentence

The Prediction Pool with End Wipe Sentinels

Fig. 2

Prediction
Subject
Predicate
End Wipe
Subject
Predicate
End of Sentence

The Prediction Pool of Fig. 2 if the First Word in a Sentence is a Conjunction

Fig. 3

Prediction
Gerund
End Wipe
Relative Conjunction
Relative Pronoun
Subject
Predicate
Comma End Wipe
Subject
Predicate
End of Sentence

The Prediction Pool with a Comma End Wipe Sentinel

Fig. 4

REFERENCES

1. ALT, F. L., "Recognition of Clauses and Phrases in Machine Translation of Languages," Report No. 6895, National Bureau of Standards, 1960.
2. YNGVE, V. H., "A Model and an Hypothesis for Language Structure," *Proc. Amer. Phil. Soc.*, 1960, *104,* No. 5 pp 444-466.
3. SAGER, N., "Procedure for Left-to-Right Recognition of Sentence Structure," *Transformations and Discourse Analysis Projects,* Report No. 27, University of Pennsylvania, 1960.
4. RHODES, I., "A New Approach to the Mechanical Syntactic Analysis of Russian," Unpublished Report, National Bureau of Standards, 1959.
5. BURKS, A. W., WARREN, D. W., and WRIGHT, J. B., "An Analysis of a Logical Machine Using Parenthesis-Free Notation," *Mathematical Tables and Other Aids to Computation,* 1954, *8,* pp 53-57.
6. CHOMSKY, N., "Syntactic Structures," Mouton and Co., The Hague, 1957.
7. OETTINGER, A. G., "Automatic Syntactic Analysis and the Pushdown Store," Symposium on the Structure of Language and its Mathematical Aspects, 567th Meeting of the American Mathematical Society, New York., April 1960, (to appear in *Proceedings* of the Symposium, American Mathematical Society, Providence, Rhode Island).
8. SHERRY, M. E., "Syntactic Analysis in Automatic Translation," Doctoral Thesis, Harvard University, 1960.
9. SHERRY, M. E., "Predictive Syntactic Analysis" (in preparation).
10. BOSSERT, W. H. "The Implementation of Predictive Analysis," *Mathematical Linguistics and Automatic Translation,* Report to the National Science Foundation No. NSF-4. The Computation Laboratory of Harvard University, Cambridge, Massachusetts, 1960.
11. ISENBERG, D., "The Predictive Syntactic Analysis Program," (in preparation).