

XXI. MECHANICAL TRANSLATION\*

V. H. Yngve	D. A. Dinneen	E. S. Lowry
Carol M. Bosche	G. H. Harman	J. D. McCawley
Elinor K. Charney	Muriel S. Kannel	T. More, Jr.
Ann Congleton	E. S. Klima	W. K. Percival
J. L. Darlington	Gisela Laurent	A. C. Satterthwait

A. GENERATIVE GRAMMARS WITHOUT TRANSFORMATION RULES

A phrase-structure grammar has been written which generates roughly the same set of sentences generated by the most comprehensive transformational grammar<sup>1</sup> of English with which we are acquainted.

Chomsky and others have argued that a grammar consisting of a set of phrase-structure generation rules, along with a very simple rule of interpretation, which assigns structural descriptions to sentences on the basis of the manner of generation,

---

\* This work was supported in part by the National Science Foundation (Grant G-24047).

(XXI. MECHANICAL TRANSLATION)

is inadequate for giving a full grammatical description of sentences in English. On the basis of these arguments, many grammarians have chosen to write grammars using transformational generation rules, rules of considerably more mathematical power than phrase-structure rules. However, as Chomsky would be the first to point out, the arguments that he has given do not show that no adequate grammar for English may be written which uses phrase-structure generation rules.

The argument against the use of phrase-structure grammars is threefold. First, such grammars will be quite long, complex, ad hoc, and therefore difficult to write. Second, grammatical description in terms of parsing alone is not complete.<sup>2</sup> Third, phrase-structure grammars cannot exploit or explain certain general features of particular languages<sup>3</sup>

In order to circumvent these difficulties the following suggestion has been made by Chomsky<sup>4</sup>:

(1) To the phrase-structure rules of the generative grammar add rules that are essentially more powerful (transformational rules).

(2) Add interpretation rules to give added structural description where certain transformational rules have or have not been used.

We have found that it is not at all necessary to introduce transformational rules to circumvent these difficulties, and, in fact, there are certain advantages in not doing so. We restrict the generation rules to phrase-structure rewrite rules of the sort described by one of us.<sup>5</sup> We retain the parsing interpretation for these generation rules.<sup>6</sup> We augment this interpretation by using a notation for the abbreviation of the phrase-structure generation rules. This abbreviated notation makes use of subscripts of the kind that have been provided for the purpose in the COMIT computer programming language.<sup>7</sup> Grammatical relations beyond those disclosed through parsing analysis are explicated in terms of derivations in the abbreviated notation. Finally we introduce an "evaluation procedure" for choosing between equivalent sets of generation rules. Our evaluation procedure involves a criterion of simplicity which enables us to exploit (and thereby explain) grammatical regularities in a given language.

The phrase-structure grammar that has been written by one of us (Harman) has been written in the form of a computer program that can produce sentences chosen at random from the set generated by the grammar. Examination of the sentences that are produced aids in eliminating errors in the grammar. This grammar generates nearly the same set of sentences as does the transformational grammar on which it is based. The only differences involve a few points at which the transformational grammar appeared to be in error.

We are now in a position to compare the two grammars from the point of view of the threefold argument that has been given against phrase-structure grammars. We have compared the lengths of the two grammars and find them to be of approximately the same

size: one reasonable method of comparison shows the transformational grammar to be shorter, another equally reasonable method shows the phrase-structure grammar to be shorter. As for complexity and ease of writing, it would appear that the phrase-structure grammar is easier to write because the rules, being unordered, are relatively more independent. Neither of the grammars can be said to be ad hoc.

The arguments that phrase-structure grammar would be incomplete and not able to exploit and explain certain general features of particular languages is not borne out in this case. Our phrase-structure grammar provides for and explains adequately all of the features of English provided for and explained adequately by the transformational grammar. It is able to do this by virtue of the additional interpretation provided by the subscript notation that also provides the compactness over an unabbreviated form of the rules.

In other words, arguments for the introduction of transformational generation rules, on the grounds that one type of grammar using phrase-structure generation rules lacks explanatory power, can be met.

Arguments that phrase-structure generation rules lack the mathematical power needed seem incoherent, at least at present. At any rate, the additional mathematical power of the transformation rules was not needed in the very sophisticated transformational grammar of English which we used for a comparison.

The result that a phrase-structure grammar appears to be adequate for English is also of great practical interest to those attempting to handle natural language by machine.

G. H. Harman, V. H. Yngve

#### References

1. An improved version of the transformational grammar given by R. B. Lees, *The grammar of English nominalizations*, Part II, *Int. J. Am. Ling.*, Vol. 26, No. 3, July 1960.
2. N. Chomsky, *Syntactic Structures* (Mouton and Company, The Hague, second printing 1962), Chapter 5.
3. *Ibid.*, Chapter 8.
4. N. Chomsky, *Syntactic Structures*, *op. cit.*
5. V. H. Yngve, A model and an hypothesis for language structure, *Proc. Am. Phil. Soc.* 104, 444-466 (1960).
6. These rules differ somewhat from those discussed by Chomsky, *op. cit.*, p. 41, footnote. The exact form of these rules is unimportant as far as our argument here is concerned.
7. V. H. Yngve, A programming language for mechanical translation, *Mechanical Translation* 5, 24-41 (1958).