PAUL L. GARVIN

A linguist's view of language-data processing

Language-data processing is a relatively new field of endeavor. As with all new fields, the exact area it covers is not completely defined. Even the term *language-data processing* is not yet generally accepted, although its use is increasing.

The definition of the field of language-data processing given here includes the application of any data-processing equipment to natural-language text—that is, not only the application of computing machinery, but also the application of the less powerful punched-card and tabulating equipment. It may even be reasonable to say that a purely intellectual procedure for the treatment of language data, which by its rigor and logic attempts to simulate, or allow for, the application of data-processing equipment, is a form of language-data processing, or at least a data-processing approach to language analysis.

From a linguist's standpoint two purposes can be served by language-data processing:

The first of these is linguistic analysis, which will ultimately and ideally include the use of data-processing equipment to obtain analytic linguistic results.

The second purpose is the use of language-data processing in information handling, where linguistics is auxiliary to the major objective. Here language-data processing is of interest for applied linguistics. It is also of interest as an area in which the usefulness and perhaps even the validity of analytic linguistic results can be tested.

Concretely, language-data processing for linguistic analysis will primarily include *automatic linguistic analysis* or at least automatic aids or automatic preliminaries to linguistic analysis. Language-data processing for information handling includes such fields as machine translation, information storage and retrieval (if based on natural language), automatic abstracting, certain intelligence applications, and the like. All these activities can be summed up under the heading of *linguistic information processing*. The two aspects of language-data processing are related in that the results of the former can be utilized in the latter. Sometimes this is not only desirable but necessary.

The area of linguistic information processing can be divided into two major subareas: (1) machine translation; and (2) information retrieval, automatic abstracting, and related activities, all of which may be summarized under the heading of *content processing.* There are two criteria for this division. One is the degree to which the results of linguistic analysis are considered necessary for the purpose. In machine translation none of the serious workers in the field will deny the usefulness of linguistic analysis or linguistic information; on the other hand, a number of approaches to information retrieval and automatic abstracting are based on statistical considerations, and linguistics is considered a useful but not essential ingredient.

Another criterion by which to distinguish between the two subareas of this field is more interesting from the linguist's standpoint. This is the manner in which the content of the document is to be utilized. In machine translation the major objective is to recognize the content of a document in order to render it in another language. In content processing the recognition is only the first step. The principal objective here is to evaluate the content in order to process it further for a given purpose. This evaluation requires the automatic inclusion of some kind of relevance criterion by means of which certain portions of the document can be highlighted and other portions can be ignored. The criterion for such an evaluation in the case of information retrieval seems to be the comparability of each particular document to those of a related set; common features and differences can serve as a basis for an index in terms of which information can be retrieved in response to a request. In automatic abstracting, various portions of the document are compared and on the basis of their relative significance are retained or omitted from the condensed version.

It is apparent that the evaluation of content poses a somewhat more complex problem for the investigator than its mere recognition. It is not surprising, therefore, that the linguistic contributions to content processing have so far been much less conclusive than the contributions that linguists have made to machine translation. It is on the other hand equally apparent, at least to the linguist, that a linguistic approach has an important contribution to make to content processing, especially if a product of high quality is desired.

It is also worth noting in this connection that important negative opinions have been voiced with regard to both aspects of language-data processing. N. Chomsky, whose approach to language was presented earlier by Stockwell in "The Transformational Model of Generative on Predictive Grammar," takes the strong position that a discovery pro cedure—that is, a fixed set of rules for the discovery of relevant elements is not a realistic goal for a science such as linguistics. This implies, of course, that automatic linguistic analysis also is an unreasonable proposition. Y. Bar-Hillel, a well-known symbolic logician and a philosophical critic of language-data processing, takes an equally clear-cut position. In a survey of machine translation in the United States conducted on behalf of the Office of Naval Research, he made the well-known statement that fully automatic, high-quality machine translation is impossible.[1] He has voiced a similarly negative view in regard to other aspects of practical language-data processing.[2]

Needless to say, in spite of the need for objective criticism and an awareness of the difficulties involved, a more positive attitude toward the field is a prerequisite for active participation in the research.

The present discussion concerns the problems of language-data processing in both senses as related to the assumed properties of natural language. We shall follow the problem through the system by going from input to internal phase. We shall not consider the mechanics of the output, since at present linguistics has little or no contribution to make to this question. More detailed attention will be given to those areas of the field which are not covered elsewhere in this volume.

AUTOMATIC SPEECH RECOGNITION AND CHARACTER RECOGNITION

All present-day language-data-processing activities use the conventional input mechanisms of punched card, punched-paper tape, magnetic tape, or the like. This is generally considered to be a major bottleneck in terms of practicality since the cost, especially for the large quantities of input that are desirable for eventual production, is prohibitive. A good deal of effort in various places is therefore directed toward automating the input.

To accommodate the two usual manifestations of the sign components of languages—the phonetic and the graphic—efforts toward automating input proceed in two directions: *automatic speech recognition* and *automatic character recognition.* The purpose is to design a perceptual device which will be capable of identifying either spoken or written signals for transposition into the machine code required by a computer. Needless to say, this objective is of considerable interest to the linguist, and linguistics has a significant potential contribution to make, especially in the case of speech recognition.

The problem in speech recognition is one of identifying, within the total acoustic output of the human voice or its mechanical reproduction, those elements which are significant for communication. The difficulty

[1] Y. Bar-Hillel, “The Present Status of Automatic Translation of Languages,” in F. L. Alt (ed.), *Advances in Computers,* New York, London, 1960, pp. 91 — 163.

2 Y. Bar-Hillel, “Some Theoretical Aspects of the Mechanization of Literature Searching,” in Walter Hoffman (ed.), *Digital Information Processors,* Interscience Publishers, New York, 1962, pp. 406-443.

of the problem becomes apparent when one realizes that most phoneticians agree that only a small portion of the total energy in the human voice output (some claim as little as 1 per cent) is utilized for purposes of the linguistic signal proper. The remaining energy serves as a signal for such nonlinguistic elements as the identification of the sex and individuality of the speaker, his state of health (whether or not he has a cold), his emotional state, and a large number of other behavorial indices. Thus, the acoustic power available for purposes of speech recognition appears to be rather small. The significant fact here is that this small percentage tends to be masked by the rest. The second difficulty is that the natural vocal signal is semicontinuous; that is, the number of physically observable breaks in the continuity of the stream of human speech is much smaller than the number of discrete elements into which the signal may be decomposed in either alphabetic writing or linguistic analysis.

The problem in phonemic analysis is one of transposing a semicontinuous natural signal into a series of discrete elements. To give an example, a short utterance such as *time* is revealed by acoustic instruments to consist of essentially two distinct physical portions: a burst following a pause representing the element */t/*, and a set of harmonic elements extending over a given period and with no observable major interruptions. In terms of phonemic analysis, on the other hand, the utterance *time* is usually interpreted to consist of four discrete units: the phonemes */t/*, */a/*, */y/*, and */m/*. The method by which the phonemic analyst arrives at this decomposition of the continuous span into its presumed underlying components is one of comparison, based on his assumptions about the dimensional structure of language. The */t/* is isolated by comparing *time* to *dime;* the */a/* is isolated by comparing *time* with *team;* the */y/* is isolated by comparing *time* to *town;* and the */m/* is isolated by comparing *time* to *tide.* The analyst gains his initial knowledge of the speech signal from his interpretation of what he hears. Not until an initial description of the elementary discrete units has been obtained does the analyst proceed to investigate the structure of phonemic fused units of a higher order of complexity such as syllables or the phonemic analogs of orthographic words.

It is not unreasonable to suppose, on the other hand, that an ideal speech-recognition device may deal directly with the semicontinuous *phonetic stretches* that are observable in the stream of speech, and that may turn out to correspond, roughly or precisely, to the fused units of phonemic analysis. The preceptual mechanism which would constitute the first component of such a device would then have to meet two objectives: first, to recognize pertinent points of interruption in the stream of speech in order to find the boundaries of the phonetic stretches; and second, to recognize, within the total energy spectrum of the human vocal signal, those particular acoustic features of each stretch that are relevant to the transmission of the spoken message.

The present state of speech recognition resembles the early stages of phonemic analysis in the sense that experiments so far have been largely limited to machine perception of short isolated stretches comparable to the short isolated examples elicited by an analyst in the beginning of his work. Just as in phonemics these short examples are used to determine an initial inventory of vowels and consonants, so the present speech-recognition work on short stretches is directed toward an identification of vocalic and consonantal features. Even in this limited framework, progress has so far resulted in the identification of only some of the gross acoustic features, such as the break between syllables, the friction component of certain consonants, and the voicing component of vowels and certain consonants. More refined identifications can be expected as acoustic research progresses, and an adequate capability for identifying isolated phonetic stretches is quite conceivable.

Little attention has been given so far to machine recognition of the interruptions in the continuity of the stream of speech that linguists call *junctures*. Linguistic and acoustic research on the phonetic characteristics of junctures will undoubtedly make significant contributions to this aspect of speech recognition.

Assuming that a perception mechanism can acquire the capability of recognizing both the boundaries and the characteristic features of the phonetic stretches of normal speech, there still remains a significant phase of speech recognition which goes beyond the perceptual. From the stretches that have been recognized, the complete device must in some way compute a linguistically relevant input for the internal phase of the data-processing system; that is, the speech-recognition device, after having identified phonetic stretches on the basis of their perceptual characteristics, must transform them into strings of linguistic signs—morphemes. For practical purposes the device might have to transform sound types not into morphemes, but into printed words or their binary representations.

Consider the problem in terms of an immediate application of speech recognition: the *voicewriter*. This device is intended to transmit the spoken message to a typewriter to obtain as output a typewritten version of the message.

It is clear that a good many of the pecularities of a typewritten document, even not counting problems presented by orthography in the narrower sense, are not directly contained as vocal signals in the spoken message. These details would include paragraphing, capitalization, and punctuation. In dictation such features of the document are either left to the secretary or indicated by editorial comments. Thus, even assuming a functioning perception mechanism, some provision would have to be made for details of this type—for instance, a capacity for receiving and executing verbal orders similar to the editorial comments for the secretary.

The problems posed by the orthography in the narrower sense—that is, the actual spelling conventions for particular words—vary to the extent that the writing system deviates from the spoken form of the language. It becomes a translation problem, comparable to the problem of translating by machine from one language to another. The same spoken form may well correspond to more than one written form, and this ambiguity then has to be resolved by context searching, which is a syntactic operation analogous to its equivalent in machine translation. Thus, assuming that the perception mechanism has identified a phonetic stretch */riyd/*, the voicewriter may have to represent it in typescript as either *reed* or *read*, depending on whether it occurred in a sentence dealing with *a reed in the wind* or a sentence dealing with *reading*.

An additional context-searching routine, comparable to the "missing-word routines" used in machine translation for dealing with words that are not found in the machine dictionary, will probably have to be included for the identification of "poor" phonetic stretches—that is, those that do not have enough signal strength or are not pronounced clearly enough to be recognizable by the perception mechanism.

At the present state of the art is appears that it may not be necessary to go through a three-step computation sequence from phonetic stretches to phonemes to morphemes or written words, but that a direct computation of morphemes or written words from phonetic stretches can be envisioned. This computation would be carried out by means of a dictionary of phonetic stretches stored in memory, to be processed by appropriate ambiguity-resolution and missing-form routines. (This conception of the linguistic aspects of the speech-recognition problem stems from Madeleine Mathiot, personal communication.)

The problem of character recognition is by comparison somewhat less complex, because—unless one thinks of a device for recognizing handwriting—the visual input into the device is discrete—that is, it can be expected to consist of separately printed or typed letters or characters. Thus, the very difficult speech-recognition problem of recognizing the boundaries of stretches within a semicontinuous signal does not exist for character recognition. On the other hand, the problem of recognizing what particular features of the signal—in terms of strokes, angles, curves, directions, and the like—are relevant to the function of the character is similar to the problem of recognizing the linguistically relevant acoustic features of speech. A further advantage of character recognition is that no computation is required to give orthographic representation to the visual signal, since the signal is orthographic to begin with. Linguists have generally given much less thought to the structure of writing systems in terms of their differentiating characteristics than they have to the phonological structure of speech and its relevant distinctive features. Thus, the linguistic contribution to the field of character recognition has been quite trivial so far. It seems that the recognition of relevant properties of shape such as the ones enumerated above is closely related to the problem of recognizing visual shapes in general, and therefore is less closely related to linguistics than is the problem of speech recognition.

Where linguistics can make a contribution is in the recognition of poorly printed or otherwise unrecognizable characters, for the gaps in the recognition string will have to be filled by a context-searching routine similar in principle to that required for speech recognition.

One of the fundamental difficulties in the area of character recognition seems to be variety of fonts that are used in ordinary print and typing. Devices which are limited to a single font—particularly if that font has been specifically designed to facilitate the operation of the device—are now in an operational stage. On the other hand, devices which can deal with a multiplicity of fonts, particularly fonts with which the device has had no prior “experience,” are still in their infancy. Some experiments have already yielded data about those characteristics which different fonts have in common and on which a common recognition routine can be based. Work is in progress on the particular perceptive mechanisms which could optimally serve to recognize these characteristics. In this respect, the field of character recognition appears to be closer to practical results than the field of speech recognition.

AUTOMATIC LINGUISTIC ANALYSIS

From a linguist's standpoint, the internal phase of language-data processing involves two types of activities: automatic linguistic analysis on the one hand and linguistic information processing on the other.

An automatic linguistic-analysis program is here defined as a computer program which, given as input a body of text, will produce as output a linguistic description of the system of the natural language represented by the text. A corollary capability of such a program will be the capacity for deciding whether or not a given input indeed constitutes a text in a natural language.

As discussed earlier in "The Definitional Model of Language," we can conceive of the system of a language as an orderly aggregate of various kinds of elements, each of which has a finite and typical set of cooccurrence possibilities with regard to other elements of the system. The elements are of different functional types and orders of complexity, as exemplified by such elements of written English as letters, syllables, words, or phrases. These elements recur in texts in a regular way, so as to form *distribution* classes in terms of shared cooccurrence characteristics.

Here the purpose of linguistic analysis is to specify the nature and boundaries of the various types and orders of elements, as well as to describe the cooccurrence patterns serving as the criteria for the definition of the distribution classes, and to list the membership of these classes. The former aspect of linguistic analysis is often termed *segmentation*, the latter is called *distributional analysis.*

Linguistic segmentation is the first step in the analysis of raw text—that is, spoken messages recorded from native informants. Segmentation procedures are based on the relation between the form (i.e., the phonetic shape) and the meaning (in operational terms, the translation or possible paraphrase) of the message. Their mechanization thus would require the comparative processing of two inputs—one representing the phonetic shape of the raw text and the other its translation or paraphrase.

A program designed for a single rather than a dual input hence cannot be expected to accomplish segmentation. We can therefore suggest that the initial inventory of elementary units not be compiled automatically, but that the automatic processing of the text for purposes of linguistic analysis use a previously segmented input consisting of units already delimited. This could be a text segmented into morphemic segments by a linguistic analyst or, the more practical alternative, a text in conventional spelling with orthographic word boundaries marked by spaces, and punctuation indicating certain other boundaries. The type of linguistic analysis to be performed on this previously segmented input would then be one of classifying the elementary input units on the basis of their relevant cooccurrence properties. That is, automatic linguistic analysis would essentially be a distributional analysis by a computer program.

The intended output of such a distributional analysis program would be a dictionary listing of all the elements (for instance, all the printed words) found to recur in the input text, with each element in the listing accompanied by a grammar code reflecting the distributional description of the element in terms of the distribution class and subclass to which it belongs. Since the purpose of the program thus is to produce a grammar-coded dictionary listing, it is logically necessary to require that the program itself initially contain no dictionary or grammar code, but only the routines required for their compilation.

The basic question of distributional analysis is: does unit *a* occur in environment *b*? This question can be answered by a computer program. The problem is primarily one of specifying automatically what units *a* the question is to be asked about, and what environments *b* are to be considered in arriving at an answer.

In ordinary linguistic analysis, informant responses are evaluated and text is examined "manually" in order to arrive at distributional descriptions by using the diagnostic contexts which are discussed in “The Delinitional Model of Language.” The difficulty of informant work, as all linguists know, is the element of subjectivity inherent in the use of a human informant. This subjectivity is maximized by using one's own self as an informant; it may be minimized by circumscribing the test situation very narrowly and by using a variety of informants, as well as other controls. However, as the questions become more sophisticated, the informant's responses become more and more difficult to control and his memory becomes less and less reliable. Thus, even in ordinary linguistic analysis one reaches a point where informant work has to be combined with the study of text.

The basic difficulty in the use of text for purposes of linguistic analysis is that large samples are required. This is understandable if one takes into account the inverse ratio of the recurrence of elements to the size of sample: The less frequently an element recurs, the larger the sample required in order to study its distributional properties. Data-processing equipment allows the processing of very large bodies of text using the same program. At the present time, lack of speech- or character-recognition devices is the greatest practical bottleneck requiring considerable expense at the input end for keypunching or related purposes.

From the linguist’s standpoint, these difficulties are balanced primarily by the advantage of the increased reliability of data-processing equipment and the possibility of attaining a rigor hitherto not customary in the field. Once costs can be brought down, there is the promise of an ultimate operational capability for processing much larger samples of language than the linguist can ever hope to examine manually. Finally, even without access to extensive programming and computer time, a partial implementation of automatic analysis can be expected to yield interesting results.

A fully automatic distributional analysis program can be looked upon as a heuristic rather than a purely algorithmic problem. A. L. Samuel has set forth some of the characteristics of an intellectual activity in which heuristic procedures and learning processes can play a major role. As applied to the problem of playing checkers, these are as follows:[3]

> 1 The activity must not be deterministic in the practical sense. There exists no known algorithm which will guarantee a win or draw in checkers, and the complete exploitations of every possible path through a checker game would involve perhaps 10^{40} choices of moves which, at 3 choices per millimicrosecond, would still take 10^{21} centuries to consider.
>
> 2 A definite goal must exist—the winning of the game—and at least

[3] A. L. Samuels, “Some Studies in Machine Learning Using the Game of Checkers,” *IBM Journal of Research and Development,* pp. 211-212, July, 1959. Quoted by permission.

> one criterion of intermediate goal must exist which has bearing on the achievement of the final goal and for which the sign should be known. . . .
>
> 3 The rules of the activity should be definite and they should be known. . . .
>
> 4 There should be a background of knowledge of the activity against which the learning progress can be tested.
>
> 5 The activity should be one that is familiar to a substantial body of people so that the behavior of the program can be made understandable to them. . . .

The above criteria seem to be applicable to automatic linguistic analysis as well, paraphrased as follows:

1 Linguistic analysis is not deterministic in the practical sense. There exists no known algorithm which will guarantee success in linguistic analysis, and the complete exploitation of every possible combinatory criterion might involve an equally astronomical number of steps as the number of moves to be explored in a checkers algorithm.

2 A definite goal does exist—a detailed distributional statement—and criteria can be formulated for intermediate goals that have bearing on the achievement of the final goal. These would be the broader distributional statements from which the ultimate, more refined classifications can be derived. Unlike checkers, the final goal can not be formulated as simply.

3 The rules of the activity are definite and can be formulated. This, of course, presupposes that one accepts as a basic assumption the possibility of linguistic discovery procedures. The procedures discussed in "The Definitional Model of Language" are those of substitution and dropping; they can be made computable, and they may be introduced into the heuristic linguistic-analysis program after certain necessary preliminary steps have been completed. Other equally computable procedures can be formulated.

4 There is, of course, a background of knowledge of the activity against which the machine results are tested: Linguistic analysis is, or can be made into, an established field and machine results can be compared to human results.

5 Although the activity of linguistic analysis is not one that is familiar to a substantial body of people, its results nonetheless can be compared to the intuitive behavior of an entire speech community, and the behavior of the program can be explicated in terms of this observed intuitive behavior.

It is thus possible to envision a computer program which will process the initially segmented text by applying to it a variety of linguistic an alytic: procedures, and will evaluate the results of each trial on the basis of certain built-in statistical or otherwise computable criteria. The program, operating in an alternation of such trial and evaluation routines, can be expected to accept certain trials and reject others on the basis of these criteria. The results of the initial tests performed by the program can then be utilized for the automatic formulation of additional tests leading to a more refined classification, until the potential of the program is exhausted and the output can be printed out for inspection by a competent linguistic evaluator.

Such a program will be particularly interesting for the analysis of languages in which word classes—that is, parts of speech—are not easily definable and where conspicuous formal marks of syntactic relations are either absent or infrequent. Examples of such languages are Chinese and English.[4]

MACHINE TRANSLATION

Let us now turn from automatic linguistic analysis to linguistic information processing. The activities in the latter field can be divided into two major categories: machine translation on the one hand and content processing on the other.

Prior to machine translation, descriptive linguists were mostly concerned with the formal features of language and considered linguistic meaning only to the extent to which it has bearing on formal distinctions. In translation on the other hand—both human and machine translation—meaning becomes the primary subject of interest. Relations between forms are no longer dealt with for their own sake; they are now treated in terms of the function they have as carriers of meaning. Meaning is granted an independent theoretical existence of a sort, since it is only by assuming a content as separate from the form of a particular language that one can decide whether a passage in one language is indeed the translation of a passage in another language: they are if they both express the same, or at least roughly the same, content; they are not if they do not.

In the process of translation the expression of the content in one language is replaced by the expression of an equivalent content in another. To mechanize the process, the recognition of the content in its first expression, the *source language,* must be mechanized; then the command can be generated to give the same content another linguistic expression in the *target language.* A machine-translation program must therefore contain a *recognition routine* to accomplish the first objective, and a

[4] For the detailed discussion of a proposed program of automatic linguistic analysis, see Paul L. Garvin, "Automatic Linguistic Analysis—a Heuristic Problem", *1961 International Conference on Machine Translation of Languages and Applied Language Analysis,* vol. 2, pp. 655- 669, London, 1962.

command routine to accomplish the second. Since the command routine presupposes the recognition routine and not conversely, a "recognition grammar" of this sort is more essential for purposes of machine translation than a "generative grammar."

For recognition of the content of the source document, the machine-translation program has to take into account, and can take advantage of, the structural properties of the language in which the content is originally expressed. In a sense, one structural property has already been accounted for by the nature of the input: the two levels of structuring, the graphemic and the morphemic, are utilized in the input by sensing the text letter by letter and recognizing spaces, punctuation marks, and special symbols. The graphemic input then has to be processed for morphemic recognition: the program has to ascertain what content-bearing element is represented by each combination of letters—that is, printed word between spaces—that has been sensed at the input. In order to effect this identification, the program can and must draw on the other two sets of levels of natural language: the two levels of organization, and the levels of integration.

The two levels of organization, those of selection and arrangement, are represented in the program by the *machine dictionary* and the *translation algorithm* respectively. It is obvious that in order to produce a non-ridiculous translation, a program must contain not only a dictionary but also an algorithm. The function of the algorithm is dual: it must select from several possible dictionary equivalents that which is applicable to the particular sentence to be translated; it also must achieve the rearrangement of the words of the translation, whenever this is necessary in order to give the appropriate expression to the content of the original. To make possible the generation of these selection and rearrangement commands, the algorithm must be capable of recognizing the syntactic and other conditions under which these commands are necessary and appropriate. For this recognition to be effective, the levels of integration of the language—that is, the fused units of varying orders of complexity-have to be taken into account. Fused units have to be identified as to their boundaries and functions. The details of this problem are discussed later in "Syntax in Machine Translation."

In an early theoretical paper on machine translation by this author[5] the statement was made that "The extent of machine translatability is limited by the amount of information contained within the same sentence." Since the sentence is the maximum domain of necessary linguistic relationships, a translation algorithm based on fixed linguistic rules appears to be limited to this domain. Later experience has shown that such

[5] Paul L. Garvin, "Some Linguistic: Problems in Machine Translation," *For Roman Jakobson,* 's-Gravenhage, 1956, pp. 180-186.

deterministic rules are, however, not the only possible translation rules. In order to deal with relations across sentence boundaries, it is necessary to assume that in addition to deterministic rules, *probabilistic rules* can be found, the span of which is not limited to the sentence. To make the distinction clear: a deterministic rule is one which under one ascertainable set of conditions comes up with a *yes* branch, under another set of conditions with a *no* branch; a probabilistic rule is one which bases its decision on a tabulation of a set of factors and branches off into *yes* or *no* depending on relative percentages rather than absolute conditions. Broadly speaking those translation decisions which are based on primarily grammatic conditions—that is, conditions of the cooccurrence of linguistic forms—will be largely deterministic. Decisions that are based on other conditions will be largely probabilistic. It is clear again that in terms of the actual design of a program, deterministic routines should be given precedence over probabilistic ones.

CONTENT PROCESSING

As indicated above, the field of content processing differs from machine translation in that it requires not merely the rendition of the content of the document, but its evaluation according to some relevance criterion. Evaluation in turn implies comparison of elements in terms of this relevance criterion; such a comparison then presupposes some orderly classification within the frame of which units can be selected for their comparability. The principles of classification will be discussed further below.

At several points in the flow of an information-retrieval or automatic abstracting system one may reasonably speak of the processing of the content of natural-language messages. At the input of an information-retrieval system is the user’s *request for information.* If this request is phrased in natural language, it will have to be processed for transmittal to the internal phase. Systems in which the request is either stated in some *standardized language* or is reformulated manually, will not require language-data processing at this point.

The internal phase of a retrieval system consists of information from which portions relevant to the request are selected for display at the output end. The ordering system used for the storage of this information can be called a system of *indexing,* since it is comparable in purpose—though not necessarily in structure or efficiency—to the index of a file or library. This indexing system can be prepared manually, in which case only the actual searching operations within the memory are automated; if it is not prepared manually—that is, if the system is equipped for *automatic indexing*—natural language has to be processed. In this case it is the natural language of a series of documents, the informational content of which is to be stored in the memory. Needless to say, systems are possible and have been devised in which neither the formulation of the request in machine language nor the storage in indexed form is done by machine, but such systems—although of unquestioned utility for a number of purposes—are of no interest in the present framework.

Automatic abstracting by the nature of the process uses documents in natural language at the input, and the system must therefore be capable of recognizing content indices in natural-language text in order to yield at the output the required condensed representation.

In all the above it is again possible to divide the automatic process into a recognition phase and a command phase. If this is done it becomes apparent that the automatic processing of the natural language of informational requests, the automatic processing of natural-language documents for indexing, and the automatic processing of natural-language documents for abstracting all fall under the same heading of being recognition operations, with similar requirements for a recognition routine. Where they differ is essentially in their command routines. The command routine for the processing of the informational request will have to include commands for translation into a search language to be used during the search. In automatic indexing the command routine will have to consist of commands for the storage of portions of documents in appropriate memory locations corresponding to the index terms under which they can be retrieved during the search. In automatic abstracting the command routines will have to consist essentially of *accept* and *reject* commands for individual sentences, if—as is the case at the present state of the research—abstracting is in effect an activity of extracting. It is possible that a future automatic abstracting system may be capable of rewording the sentences extracted for retention in the abstract by generating natural-language text on its own in order to approximate more closely human abstracts, which have certain characteristics of continuity and readability that are absent in mere extracts. This particular final phase of automatic abstracting is the one area of language-data processing in which at the present we can visualize a genuine practical way of sentence generation by machine. This is, of course, as yet for the future, but it may turn out to be an important area of application for some of the efforts of linguists today in the formulation of generative grammars.

From the linguist's point of view, the major purpose of language-data processing as discussed above is the recognition of content for purposes of comparative evaluation. The program must ideally be capable of doing two things: it must first recognize an individual *content element* in the natural-language text (a single word or a relevant combination of words) ; second, it must be able to decide on some comparability criterion for each of the content elements that it has found.

In order to meet the first of these requirements, the program will have to include some features comparable in nature to those of the algorithms used in machine translation. Something like an *idiom routine* is necessary to recognize word combinations that represent single content elements, as well as provisions for the recognition of the syntactic units and relations relevant to the objective. In technical terms, the system must contain a machine dictionary equipped with a grammar code capable of calling appropriate subroutines for idiom lookup and syntactic recognition.

An additional area of the application of syntax routines to content processing has been suggested by one school of linguistics: the automatic standardization of the language of the original document. The purpose of such a set of routines would be to transform all the sentences of a document into sentences of a type exhibiting a maximally desirable structure, namely kernel sentences, as discussed earlier by Stockwell. The work now in progress under the direction of Professor Z. S. Harris at the University of Pennsylvania's Transformation and Discourse Analysis Project is, as far as I know, primarily concerned with the application of transformation theory to this objective. The aim of the work is to be able to reduce the sentence of natural-language documents to a standardized kernel shape. The assumption is that storage in this kernelized form will significantly facilitate retrieval.

To meet the second requirement, the dictionary will have to include, in addition to a grammar code, a semantic code capable of calling appropriate subroutines for content comparison and evaluation.

SEMANTIC CLASSIFICATION AND SEMANTIC CODE

In order to compare content elements to each other in terms of some relevance criterion related to the goal of the operation, whether it is processing of the request, assignment to index terms, or acceptance or rejection for the extract, these elements must be classified in terms of the content which they represent, rather than in terms of their formal co-occurrence characteristics. The semantic code will thus have to be able to refer each dictionary entry to the appropriate area of content representation—that is, to the semantic class to which it belongs. For optimal efficiency such a semantic code ought to be based on a systematic classification of content elements. Classifications of a kind can be found in existing thesauri and partial classifications can be found in synonym lists. There are two major defects in thesauri of the Roget type: one is that they usually do not contain enough of the technical terminology required for most practical content-processing purposes; the other, more serious from a linguistic standpoint, is that they are compiled purely intuitively and without adequate empirical controls, sometimes on the basis of an underlying philosophic assumption, and do not necessarily reflect the intrinsic content structure of the language which they represent. Most synonym lists have similar weaknesses.

These criticisms are based on the assumption that there may exist for each language a system of content in the same sense in which there exists the formal system that linguists deal with when they treat the various levels of a language. This assumption is not unreasonable in view of the intuitive observation that the meanings of content elements are not unrelated to each other. It is, after all, from a similar assumption of the systematic relatedness of formal elements that modern descriptive linguistics has derived its results.

It is thus possible to envision a systematization of content, or meaning, not unlike the systematization of linguistic form for which today we have the capability. It is likewise not unreasonable to assume that some of the methods which have allowed us to systematize formal linguistic relations may contribute to a systematization of content relations. The following linguistic considerations have bearing on such a systematization:

First, the basic assumption that there exists for each language a system of meanings comparable to the system of forms allows application of linguistic methods to the problem of meaning.

Second, two methodological assumptions can be made that allow the formulation of linguistic techniques for the treatment of meaning: (1) that, irrespective of theoretical controversies about the "nature" of meaning, there are two kinds of observable and operationally tractable manifestations of linguistic meaning—translation and paraphrase; and (2) that linguistic units with similar meanings will tend to occur in environments characterized by certain specifiable similarities.

The first assumption allows the formulation of techniques for semantic classification based on similarities in the translation or paraphrase of the content-bearing elements in question. In a monolingual approach, which most workers in the area of content processing would consider the only reasonable one, these would be paraphrasing techniques.

The second assumption permits the extension of linguistic techniques of distributional analysis from problems of form to problems of meaning.

In order for either type of linguistic techniques to yield significant and reliable results, the conditions affecting their application will have to be controlled and the appropriate comparison constants specified. If this is done, one may reasonably expect to arrive at a semantic classification of the content-bearing elements of a language which is inductively inferred from the study of text, rather than superimposed from some viewpoint external to the structure of the language. Such a classification can be expected to yield more reliable answers to the problems of synonymity and content representation than the existing thesauri and synonym lists.

Two directions of research can be envisioned at present: the application of a technique of paraphrasing, and the investigation of the role of context in the definition of meaning. Both lines of study can be based on prior linguistic research experience.

The paraphrasing technique can most usefully be applied to the study of the verbal elements of a language such as English. It can be based on the use of replacement predicates, limited in number and of the required semantic generality, which can be substituted for the original verbal elements found in the sample text that is to be processed.

The following example illustrates how such replacement forms can serve to define the potential semantic features of a given original form:

First rephrasing operation
Original statement: The induction and the confirmation of the theory depend on experience.
Original form: depend on
Replacement form: be based on
Resultant statement: The induction and the confirmation of the theory *are based on* experience.
Comparison property: semantic similarity
Semantic feature induced from operation: basic relation

Second rephrasing operation
Original statement: In all other cases, the magnitudes of the elements *m, r,* and *t* of the problem depend on the motion of the observer relative to point P_0.
Original form: depend on
Replacement form: vary with
Resultant statement: In all other cases, the magnitudes of the elements *m, r,* and *t* of the problem *vary with* the motion of the observer relative to point P_0.
Comparison property: semantic similarity
Semantic feature induced from operation: covariance

These operations have yielded a crude first-approximation semantic spectrum of the verbal element *depend on,* which can be represented in a manner suitable for adaptation to a semantic bit-pattern code:

Lexical unit	*Semantic features*		
	basic relation	*constituency*	*covariance . . .*
	be based on	consist of	vary with . . .
depend on	1	0	1

It is assumed that the successive application of paraphrasing operations to a large sample of text will serve to establish a series of semantic features for each verbal element that has occurred. On the basis of similarities and differences in their respective sets of features, the verbal elements can then be arranged in a systematic thesaurus. Such a thesaurus would have been inductively derived from the processing of text, and thus could be considered empirically more reliable.

Once a thesaurus of verbal elements is available, the nominal elements of the language can be classified semantically on the basis of their co-occurrence with semantic classes of verbal elements. Each nominal element in a text can then be assigned semantic features, depending on whether or not it has been found to occur as the subject or object of members of the various classes in thesaurus. This research can ultimately be automated, but first the detailed requirements for such a program must be worked out.

The application in content processing of such an empirically derived systematization of meaning is outlined below as it would apply to information retrieval.

As mentioned above, natural language has to be processed at two points in the flow of an ideal system: the inputting of requests, and the automatic assignment of document content to index terms. The latter further implies the systematic storage of the indexed information, based on a systematization of the terms to which the information has been assigned. The processing of the request, finally, has to be related to the ordering of the stored terms to permit retrieval of pertinent information. Thus, both language-data-processing operations are ultimately referred to the same semantic system. It is possible to envision this semantic system as a set of thesaurus heads and subheads, with the individual content-bearing elements of language classed under the lowest subheads, each of which will include under it a number of elements which for all operational purposes can be considered synonymous.

The semantic classes and subclasses subsumed under the heads and subheads will have been derived inductively by the linguistic techniques suggested above. They will have been based on a finite set of semantic features ascertained by these techniques. These features would be classified so that the broadest classes would be defined by shared features that are few in number and more general in scope, the narrowest subclasses by features that are many in number and more specific.

The criteria on which to base these semantic features and the techniques for ascertaining them could be related even more specifically to the purposes of automatic indexing or abstracting than the techniques described above. In the case of information retrieval it might be reason able to base equivalences on reference to the same subject-matter area rather than on some simple relation of synonymity based on sameness of content.

The information derived from such an analysis of the semantic system of the language can then be incorporated in the semantic code, which is part of the machine dictionary. In the indexing phase of a retrieval program, the document can be run against the dictionary; the semantic code would then assign its contents to the appropriate index terms, which can be stored in memory in the ordering derived from the semantic system on which the semantic code is based. In the request-reading phase, the request can be run against the dictionary, and the semantic code could serve to extract the index terms by means of which the required information is retrieved from storage and furnished in answers to the request.

In summary, it is worth noting that the major difference between a linguistic and a nonlinguistic approach to content processing is that the former ideally requires the inclusion of a previously prepared machine dictionary with a dual code: a grammar code and semantic code. To offset this added complexity, a linguistic approach should contribute increased accuracy and reliability.

This discussion has been limited to some of the areas of linguistic contribution, both theoretical and methodological. This is not intended to imply that techniques based on nonlinguistic assumptions and approaches, whether statistical, logical, or philosophical, are in any way considered potentially less significant. On the contrary, the rules which may be derived for content processing from a linguistic analysis of content systems might well turn out to be largely probabilistic rather than deterministic in the sense these terms are used in the above discussion of machine translation. If this is so, the linguistic classifications will indeed have to be meaningfully related to the statistical, logical, and other considerations which are now being set forth in other areas of language-data processing.