NEXUS

A LINGUISTIC TECHNIQUE
FOR
PRECOORDINATION

R. A. Benson
Convair Division of General Dynamics
Graduate Student
San Diego State College
San Diego, California

Abstract

A method for automatically precoordinating index terms was devised to form combinations of terms which are stored as subject headings. A computer program accepts lists of auto-indexed terms and by applying linguistic and sequence rules combines appropriate terms, thereby effecting improved searchability of an information storage and retrieval system.

A serious failing exists in many indexing systems in that index terms authorized for use are too general for use by technically-knowledgeable searchers. A search conducted using these terms frequently produces too many documents not specifically related to the users' requirements. An indexing method using the language in which the document was written corrects this failing, but eliminates the generality of the previous approach. A compromise between indexing generality and specificity is offered by NEXUS precoordination which combines specific terms into subject-headings, eliminating improper coordination of terms when matching search requirements with document term sets.

NEXUS examines the suffix morpheme of each input term and determines whether or not the term should be a member of an index term combination or precoordination. If insufficient evidence is present to make such a

determination, a sequence rule goes into effect which combines terms based on their syntax.

A variety of corpora was used to test and develop the NEXUS precoordinator. Data bases consisting of legal information, computer program descriptions and NASA linear tape system documentation were used. More variety was present in the NASA documents which made the results of the application of NEXUS to this collection more significant than the others. Also, a fuller battery of rules was developed by this time, increasing the power of the program.

Summary

NEXUS is a research project which is concerned with input processing of natural language for information retrieval.

The computer program used to do this task consists of linguistic rules that operate on the suffix portions of printed words, and the order of these words as they appear in a sentence.

NEXUS accepts lists of index terms that have resulted from the application of an auto-indexer program to titles and abstracts. These term lists are processed by NEXUS in order to form combinations of terms which are stored as subject headings. Such subject headings or precoordinations aid the searcher in finding information when they are used in a bibliographic printout. As opposed to coordinate-indexed printouts, consisting of lists of individual terms and the accession numbers of the source documents, those printouts of NEXUS-processed terms contain word combinations that have been precoordinated, saving time and increasing accuracy for the searcher.

It must be stressed that NEXUS operates on general rules. There are occurrences in language that are not coverable by this method. Storage by

individual terms is effected in conjunction with NEXUS so that nothing is missed because of rule exceptions.

Comparison tests have been run using the full NEXUS program, a partial application of the program using sequence rules (SEQS), and human analysis of the same data. Although falling short of human analysis in some respects (except for consistency), the NEXUS approach is more effective than SEQS in producing effective combinations.

Although some suggestions are made for applying this technique along with a possible output format for a bibliographic application, the chief value of this effort, however, has been to further study those aspects of language that are amenable to computerized analysis for the purpose of improving input and output functions in information retrieval.

SECTION 1

Introduction

Of all the various operations of an information retrieval system, the input function is the most important. The decision of what to store to best represent the contents of a document involves predicting to a degree how this representation will be looked for by a user. If a user is not conversant with a subject he must be led into it by familiar, more general routes. If a user is conversant with a subject and is perhaps a contributor to its literature himself he will be after specific details which he will request, preferably in the language of his discipline. This dichotomy of users probably exists, to some extent, in any information retrieval situation. It is the intent of such research as NEXUS to help alleviate this paradox by permitting access to information by both general and specific indexing accomplished by machine.

The indexing process is discussed in this paper starting at the point where it first becomes necessary. The qualifications for an expert indexer are then enumerated, and the activity of the indexer is examined. Generalized and specific indexing are compared and, finally, a suggestion is made for converting the results of specific indexing into generalized subject headings, which is the purpose of the NEXUS programs.

Operational tests have been conducted during the stages of developing this approach, and a variety of data was used to allow testing across different types of information.

Comparison tests were made using the full set of NEXUS rules vs. only the sequence rule, SEQS. The intent was to find out how more effective the program works using suffixal morphemes to combine terms than to merely connect words that follow one another in sequence.

The NEXUS-generated subject headings can be used in bibliographic printouts to aid in locating desired information. Combinations of terms prepared in this way avoid the occurrence of incorrect coordinations of terms which sometimes happens when individual terms are coordinated by the user.

SECTION 2

Indexing for Information Retrieval

An individual is faced with the prospect of maintaining a growing collection of documentation. The documents in this collection contain information that will answer frequently asked questions. When the collection consists of a few documents, this individual can read them all and be prepared to answer these questions. But, as the amount of documents increases, he will be forced to find some method of recording clues to the information found in each document. These clues will have to be stored separately from the documents, on a list or perhaps on file cards, so that the maintainer of the documents

can scan them easily. When he is asked a question, instead of trying to remember which document or documents have the answer, he goes to his list of clues, and then selects the documents from the collection. The number assigned to each group of clues is the same as the number on the document.

Let us assume that most or even all of the questions asked of this individual are predictable. He is then in the fortunate position of being able to look for specific answers to specific questions as he records the clues from each incoming document. He can then arrange the list of clues in whatever order is most convenient for him. He can arrange the clues by frequency of questions asked, he can classify the clues by hierarchical relationship, by chronology or by any other convenient method that might best or most quickly answer these stock questions.

In some very fortunate cases, a collection of documentation consists of documents that have been specifically designed to answer questions. Each document is constructed with a consistent number of information or data blocks and the contents of these blocks vary to a predictable degree. The recording of information clues (we may as well now refer to this function as indexing) then becomes a simple task.

Collections of technical papers, the most common type of information collections, do not lend themselves to similar handling. One can predict only to a very small degree, what questions will be asked of such a collection. Therefore, the indexer must select clues from each document based on his speculation of what questions will be asked in the future. It would seem that we are now getting a vague picture of what an indexer looks like. He is able to pick up any highly technical paper, most of which are at the forefront of their disciplines (otherwise why should they be published?), to understand the content of this document so expertly that he can predict the questions that

5

will be asked and then answered by this document, and then to record the clues to its contents in such a manner that they will lead a searcher directly to this segment of recorded knowledge at some unknown future date. This astute person must certainly possess knowlege equivalent to advanced degree level in numerous scientific disciplines, he must have working knowledge of many of the world's languages, surely he must possess an advanced degree in Library Science (more popular - Information Science), and the knowledge of practical economics to such an extent that he can subsist comfortably on six to seven thousand a year (the going rate for indexers). Armed with such a formidable background this individual would render better service, at least to himself, by doing the research and writing the paper himself.

Obviously, the indexing function must be performed by someone less qualified than the individual described above.

In a normal library atmosphere, the area usually given responsibility for the important endeavor of maintaining documentation collections, there is a traditional way to process such material. Indexing is performed using such aids as subject-heading lists or thesauri. The documentalist/librarian use of the term, thesaurus, refers to a dictionary-order list of approved indexing terms, similar to a subject-heading list.

The indexer, in the above-mentioned environment, scans a document, tries to figure it out the best he can, and then selects terms from these approved lists that he thinks best describe the document. Sometimes this works, sometimes not. After all, the indexer cannot be expected to be expert in all technical fields. Anyway, the resulting terms that are the clues to the document's content are generalizations of this content. It goes without saying, if a researcher is writing about a new usage of holography in pathological x-ray applications, this document surely has something to do with photographic techniques in medicine. If holography is not an approved term,

6

it will eventually be added to the list when approved. In the meantime, it cannot be used, of course. But the term, x-ray, has been around long enough to be acceptable, and the searcher can hunt around at a higher (more general) level until he locates the document.

The point is, such approved term lists are designed to aid the partially knowledgeable library user (or library worker) who does not know the technical vocabularies of special disciplines well enough to use them intelligently. The use of generalized terms stems also from the attempt, on the part of librarians, to store their reading materials in related clumps within a library. This is understandable in a public library or even in a book collection of a technical library. A user wants a book on computer programming, so he goes to the section of books that contains programming books. However, if he wants to know the latest published research on a particular programming technique he will find it in document or journal article form. He will know, in his own terminology, what he wants at a considerably more specific level than "computer programming," or say, than the approved Library of Congress subject heading, "Electronic digital computers - Programming."

Why not use the terms the researcher uses? Well, they are not controlled, you might say. A term might be in vogue today that is turned into something else tomorrow. You will clutter up your list of document clues (index) with variations of the same term. You may find some words that mean the same thing. The truth of the matter is, that the actual synonym is not as common as you might think. Slight variations of meaning exist in many words that seem to be synonymous with others. These slight variations may turn out to be highly significant in many contexts. If the words of actual technical jargon are used, some later editing may be in order, it is true, due to the high volatility of language in fast-moving technology, but

7

the documents will be accessible to people who know this language, without translation for the benefit of the middle-man.

The knowledgeable searcher knows this language. He uses it every day, and keeps up with its variations. The collection of documents is for his benefit, not for the convenience of the library worker.

What impact does all this have on documentation? And specifically on indexing?

Let us assume, for just a minute, that we do not have a crew of super-intelligent people for indexers. Instead, we have a few competent clerical workers well-enough educated to spell words properly. They can't do foreign languages, so let us, necessarily, eliminate those documents for the present. But they can read titles; they know an author from a date; they can identify an abstract. Within the latter, they are able to tell what words are being used to describe some esoteric subject although they are unable to define the meanings of those words.

If these people know enough to get them this far, they can eliminate the function words (is, an, the, but, etc.) from the content words (holography, pathology, thorax, etc.), copy down these latter content words and, in effect, perform indexing. This is indexing at the specific, not the general level. To generalize these terms one would have to know that holography is related to photography, pathology is related to medicine, thorax is related to anatomy, and so on. We don't expect that much sophistication from our clerical workers. We really can't afford to pay for that much knowledge. Actually, we don't want them to know that much. It could bias their indexing.

This is exactly the way the KARDIAK[1] automated bibliography on artificial heart research was produced. Now that it has been released (almost three years ago) and has received some acclaim throughout the world of medical research (e.g., Harvard Medical School, National Library of Medicine,

National Institutes of Health, etc.), we in Technical Data Systems (better known as IS&R), the compilers of this useful work, are still unable to use it! Why? Because we are not, nor should we be, conversant with the terminology used to index it. We just don't know that much about the technical specialty of cardiac medicine. When we are asked to demonstrate how KARDIAK works, we must use a standard search of two terms; "Ebstein" and "Anomaly." During the production of this, Dr. Shafer, Artificial Heart Study Program leader, introduced us to Dr. Grey, an eminent cardiac specialist from India. At this point, KARDIAK was 50% compiled. We had about a thousand entries and had produced an interim version. Dr. Grey was asked by Dr. Shafer to pose a question to this half-KARDIAK. He thought for a moment and then asked us if we had anything in the bibliography on "Ebstein's anomaly." For all we knew of this phenomenon, it could as well have been "Einstein's anachrony." KARDIAK was queried with these terms, however, and produced a sufficient quantity of answers, to our relief and to the pleasant surprise of Dr. Grey. (He later asked for copies.) Anyway, we now use this same query as a test query of the system, because we don't have the sophistication to ask anything else.

We should add, however, for justice's sake, that if the KARDIAK were on "Information Science" instead of "Cardiac Medicine", the situation would surely be reversed.

The thesis, so far, has hopefully convinced the reader that it is possible to index highly technical collections cheaply and accurately without super-intelligent, universal men wielding the indexer's pencil. But we are still faced with the problem of some cross-discipline communication. We cannot query a collection on "Cardiac Medicine", and they cannot query a collection on "Information Science." Now then, how do we go about communicating to one another through the medium of a general-information collection? That

9

is, how do we do this without getting too general and paying the price for this generality?

KARDIAK, once again, has given us a clue to how this may be done.

As we were feeding KARDIAK the terms selected by our clerk/indexer, some of these terms kept recurring; recurring with such frequency that our computer program could not hold them all in storage. That is, there was not enough room set aside to hold all the document numbers with which these terms were associated. The number of these terms was small, only seven in all, but the number of documents that used these seven terms was extensive. Because of the physical impossiblity of storing all these document numbers, these terms were rejected for storage. Oddly enough, perhaps serendipitously enough, if you will, these were the terms that generally described the collection:

Artificial

Heart

Cardiac

Valve

Extracorporeal

Blood

Circulation

We have here, then, the general terms to describe the KARDIAK collection, and we have them delivered automatically. If we were to decide that we must have subject headings to communicate in a general fashion to other less knowledgeable searchers, in this case to ourselves, these are doubt- less the best candidates. Just for practice, let's make subject headings out of this list:

Artificial Heart

Cardiac Valve

Extracorporeal Blood Circulation.

We don't need an approved list of terms. We couldn't have found one, nor known how to use one, if we had had one. It has been said, "Let the documents themselves generate their own terms."[2] One step further, let the terms rejected because of over-frequency be combined as subject-headings. These combinations can then be used as general descriptors for the particular collection.

The KARDIAK is a closed collection. That is, it was produced for a specific purpose, it served its purpose, and it is now a static piece of documentation history. Of course, it can always be picked up at a later date and be added to; but we don't foresee this happening at the present time. This is all leading up to the fact that there is any amount of manipulation one can perform on a static collection that cannot be done on a growing one. When a collection is constantly being added to, one must figure out a way to maintain control of it as it develops. If the collection is specialized enough, the term rejection factor, mentioned above, will still appear. But, as the collection grows, we certainly must increase our storage capacity of the ratio of document numbers to terms. This ratio probably remains the same, but we can't say so for sure unless we do some research on it. This is an area for further work with which we are not principally concerned in this report.

What we would now like to suggest is an interim feature: an aid to indexing and searching that is in between a free, specific, individual key word system and a generalized, controlled subject-heading system. We have already shown an almost algorithmic way of doing indexing. The clerical worker identifies a title and an abstract, and separates content words from function words. The function words are then copied down, or in the case of KARDIAK, are keypunched directly on punched paper tape. It's easy to

imagine a machine doing essentially the same operation, and this is what we have done.

A program was written similar to one described in previous research[3] which, using a function word deletion list, scans lines of text and records the content words that are in the original syntactical order of the text. Such a method resembles the well-known KWIC indexing system. These remaining content words can be used as index terms for searching the collection on a specific level. They can be stored on tape with each term added to previously stored usages of the term by recording the document number under that term. Or, in the case of a first-time usage of a term, a new entry on tape is made. Now, so far, this is essentially what was done by the clerical worker. But now we have avoided her occasional human errors, and since her human judgment was previously discouraged we have lost very little, and have gained a great deal in speed and accuracy.

At this point, let's switch over to the searching function. The searcher knows the terms he is looking for, if he knows the technical specialty concerned. His query will be couched in these same terms. Therefore, he proceeds in his search of the collection by combining terms and looking for coordinating document numbers. (This follows no matter if he is doing it manually, such as with KARDIAK, or whether a computer search is made.) One element is missing, however, and that is syntax. He must presume that the hits he comes up with are of terms arranged in the same syntactical order as his search query. In other words, he is attempting to regenerate sentence order. This is successful much of the time, but then again there are times that it doesn't work.

If we had our clerical worker again, we could show her some lines of text and ask her to combine words that bear relationship to one another. If she did a good job of making combinations, some of this missing syntax

would be recovered. Let's take a title, for example: "Applications of Linguistic Experiments to the Industrial Community." Our clerk would probably make the following combinations:

"Applications" (not combined)

"Linguistic Experiments" (combined)

"Industrial Community" (combined).

These term combinations aid in restoring syntax, to some degree, where the free terms might be recalled out of order; for example, something like "Linguistic Community" or "Industrial Experiments" or "Community Experiments," all of which are entirely misleading in regard to the actual meaning of the title. Now, for the clerk to do term combining correctly, she uses some simple rules. The most obvious rule is that of sequence. There are other rules used that are not so obvious, even to her, because she may not know she is using them. These rules have to do with linguistics, specifically suffixal morphology. This is to say that the suffixal morphemes of the words in this title are giving her clues about the relationship of one word to another. In other words, the presence of one of a group of particles at the end of a content word in a line of text will give a clue to its relationship to the next content word. Of course, the next word in sequence must be examined for the presence of a final particle, as well. Let's take "linguistic experiments" as an example. The two words are in sequence in the text line, even though this is not an absolute indication that they should be combined. The suffixal morpheme of "linguistic" is "-ic," an adjectival ending. And since there is no punctuation following "-ic," this indicates the proximity of some next entity to be modified, some noun form coming up. In our example it is "experiments." But, if the suffixal morpheme of "experiments" were "-al" instead of "-s," and there is still no following punctuation, we would have a clue that we don't yet

have a noun form to be modified. We have two adjectives stacking up, and the next following word may be the noun form we have been waiting for. However, the "-s" morpheme is most likely acceptable enough as a noun plural ending, and the combination "linguistic experiments" is a valid one.

The application of such rules by our clerical worker is automatic because she does all these operations following the rules that are built into her knowledge of the language. She might possibly be able to explain the process but it is so obvious and natural to her that she might not be able to.

To do this function by machine ia another matter. We must not only explain the process, but we must also instruct the computer precisely what to do and in what order to do it. And also, unfortunately, we must put up which occurrences of letter constructions that look like a legitimate suffixal morpheme, such as the plural "-s", but are actually not; constructions which would be immediately obvious to our clerical worker.

Succeeding sections of this report will outline the method used (NEXUS) to precoordinate terms during the automatic indexing process.

All programming of this research task was accomplished by James C. Moore and G. E. Sullivan, of Department 591-0, in FORTRAN II. The computer used was the CDC 160G.

<div align="center">SECTION 3</div>

NEXUS I

The inspiration for NEXUS came from a particular collection compiled by IS&R on legal literature.

The indexing was done by an individual highly trained in law but who had never done any previous indexing. His indexing consistency, to begin with, was slightly erratic in that he occasionally repeated terms in bound form that he had already noted down in free form. However, as he progressed through the collection of 1742 documents his indexing became more stabilized.

<div align="center">14</div>

Each document was given an accession number. The index terms, usually six or seven of them, were listed under the number. The indexer wanted retrieval by date at some future time, so he used the year the document was published as an index term in every case.

The output of this project was a KARDIAK-type (or "busted-book", as it is known in IS&R) manual index, which was produced by computer. The terms were sorted alphabetically and the document numbers of the documents indexed by the term listed beneath each term in ascending order.

Precoordination of these terms would have aided the searcher, in the way previously indicated, as a time-saver and a syntax safeguard. This would have prevented the searcher from erroneously hooking together terms that actually were not related.

To begin with, the unsorted sets of index terms were used as input to NEXUS. NEXUS was first put together in a very rudimentary form. The dates were isolated and the criteria for precoordination were based on (1) sequence, (2) "-ed" suffixal morpheme in the first position, and (3) "-s" suffixal morpheme in the second position. The flow chart for NEXUS I, with the aforementioned legal collection in mind, is shown in Figure 3-1. The first step (1) is to examine the first term in the document term set under initial examination. If the first term is a date (2), we don't want to couple it with another term, so we leave it as a single term and move on to the next word (3), if there is one. The next word is examined as a first word (4), and if it is not a date, it is tested (5) for a final plural morpheme, "-s". If it does end with "-s", a preceding word is looked for (6). If no preceding held word exists, the term is printed as a single term (7). If the term does not have an "-s" ending, it is held for pairing (8) with the next word in the set (9).

If this held word is the last in the set, it is also (7) printed as a single term. But, if there is a next word (10), the next word is examined and (11)

START

(1) LOOK AT FIRST WORD

(2) IS IT A DATE ?

(3) IS THERE A NEXT WORD ?

(4) GO TO NEXT WORD AND TREAT AS 1ST WORD

(5) END WITH "-S"?

(6) IS THERE A PRECEDING WORD?

(7) PRINT AS SINGLE TERM

(8) HOLD FOR PAIRING WITH NEXT WORD

(9) IS THERE A NEXT WORD ?

(10) LOOK AT NEXT WORD

(11) IS IT A DATE ?

(12) PRINT HELD 1ST WORD

(13) END WITH "-ED" ?

(14) PRINT HELD 1ST WORD AND HOLD THIS WORD FOR COUPLING

(15) PRINT HELD WORD WITH THIS WORD AS COUPLING

(16) DOES PRECEDING WORD END WITH "-S"?

(17) PAIR UP AS PART TWO OF COUPLING & PRINT

(18) GO BACK ONE WORD

(19) END WITH "-S"?

(20) PRINT THIS WORD WITH EACH WORD ENDING WITH "-S"?

(21) IS THERE A NEXT RECORD ?

(22) GET NEXT RECORD
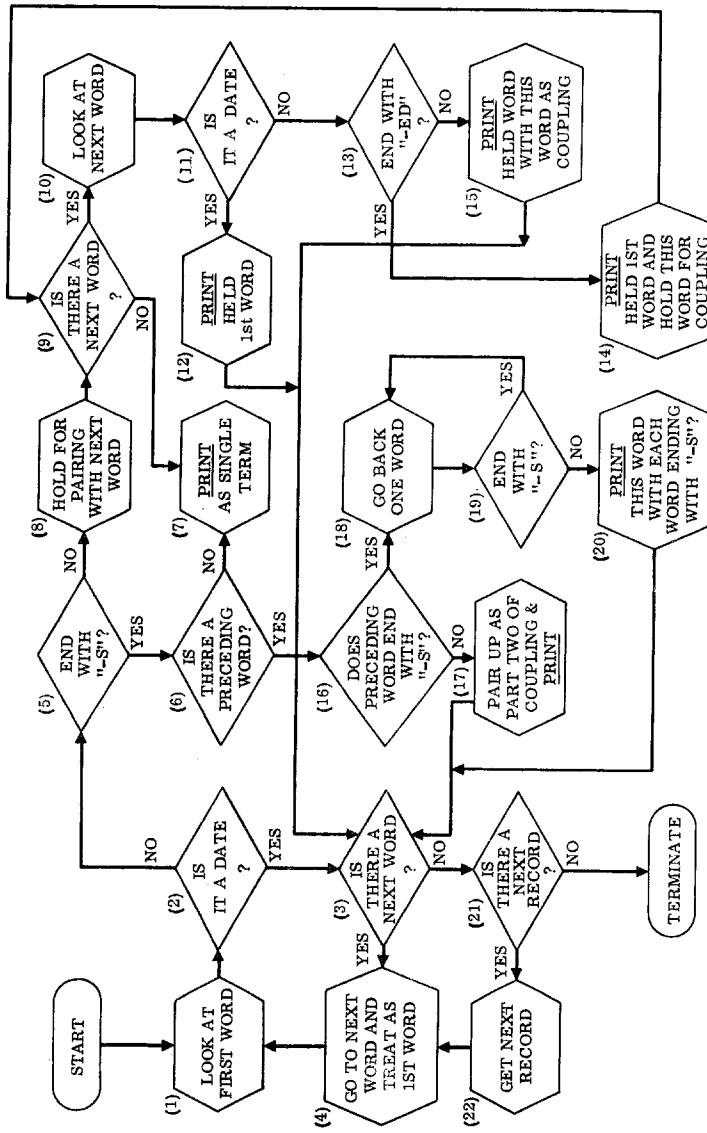
TERMINATE

YES   NO

Figure 3-1. NEXUS I Flow Chart

16

tested for being a date. If it is a date, it is printed (12) as a single term. If not, it receives a test for "-ed" (13) as the final morpheme. This morpheme can only be allowed with the first word of a pair (unless, of course, it is the last term in the set; in which case it is printed alone). If "-ed" is present, the held first word (14) is printed by itself, and the "-ed" word is held for first-position pairing. If "-ed" is not present, the held word is printed with this word (15) as a coupled pair.

Let's go back to (5) where a word is tested for the presence of an "-s" final morpheme. The word does end with "-s", so we check for a preceding word (6). In this case we will get "yes" for an answer, and the next test is (16), "Does the preceding word end with "-s"? If "no" to this test (17), the word with "-s" ending is printed in the second position of a pair, with the preceding word in first position. If the answer to (16) is "yes", the function (18) is activated, which checks the word preceding and (19) checks that word for a suffixal "-s". The program loops between (18) and (19) until a non-"-s" suffixal morpheme word is found. It then (20) prints the latter word in first position, followed by all "-s"-ending words. This portion of the NEXUS I program can produce precoordinations of more than two words. The remainder of the tests and functions on this flow chart are probably self-explanatory. If the program runs through the set of terms for one document, indicated by a "no" at test (3), the next test (21) asks "Is there a next record?". If "yes", the next set of terms for a document is brought up by function (22) and the processing continues. If all document term sets have been processed, the answer to (21) is "no", and the program terminates.

The NEXUS I program processed all 1742 document sets contained in the legal information system. The results of this processing produced 4078 combinations. 3527 of these were good precoordinations. 154 times terms with "-s" suffixal morphemes were isolated and thereby avoided ambiguous

17

combinations. 397 precoordinations were unsuccessful. The latter quantity, however, was the source of further rules that will be applied to future versions of NEXUS. We knew that the development of this program would have to involve expansion of the rules step by step. So some of the bad coordinations showed us where more rules could have been applied to avoid them. Of course, some of these anomalies were unavoidable. They were merely caused by characteristics of the language with which we have to live if we are going to continue to speak English. For example, one term set listed the following terms:

Jurimetrics	Investigation
Committee	Legal
Scientific	Problems.

Because of our "-s" rule in second position only, the program isolated "Jurimetrics" instead of making the obvious (to a human) coordination, "Jurimetrics Committee." The rule must be valid for only one position, and the second position is the most common one. Continuing the sequence, "Committee" was precoordinated with "Scientific" because of the sequence rule. This is also an obvious error to a human, because of the suffixal morpheme "-ic", which is part of "Scientific." In analyzing the production, so far, "-ic" seems like a good candidate for a first-postion suffixal morpheme; so, it became one in the next version of the program. The next combination, "Scientific Investigation," turned out successfully because of sequence, but "Investigation Legal" went bad; once again because of a suffixal morpheme cue that wasn't included in the program.

This morpheme was the "-al" on the term "Legal" which was later included as a first-position rule. Finally, "Legal Problems" was produced, meeting the requirements of both sequence and "-s" rules.

18

Please bear in mind that the rules incorporated in this program can never attain 100% effectivity. Natural language won't allow it. Still, NEXUS I delivered 90% correct precoordinations, which is encouraging as the first try of an experimental program.

SECTION 4

NEXUS II

Based on the success (and the failures) of NEXUS I, an expanded version of the program was written. NEXUS II was made more effective by adding rules principally affecting first-position qualification, and one rule affecting both first and second position.

The new first-position rules included the suffixal morphemes "-al", "-ern", "-ese", "-ic", "-ive", "-ly" and "-ous". The remaining rule was one that prevented two words with "-ing" endings from being paired together.

As you may have noticed, the first-position rule, "-ous" conflicts with the second-position rule, "-s". The latter rule looks for a final "-s" only and when it finds one, qualifies the term for second position. Because of this, the "-s" test must also include a test for preceding "o" and "u". When these are present, we have a first-position rule in effect; when absent, a second-position rule.

One of the NEXUS I rules was eliminated. The rule for stacking "-s" words and attaching the first non-"-s" as a first-position word. This rule did not produce anything of value, and could possibly have contributed to ambiguity. However, a turnabout version of this rule was adopted. This rule, if it locates a sequence of first-position suffixal morphemes, will stack them up until it finds a second-position word. It then prints them all in combination. In this way, we have a method for creating strings of terms in precoordination consisting of more than two words. "Three-dimensional Holographic Techniques," is an example of a production of this kind.

19

NEXUS I contained an overlapping feature which we haven't mentioned, but which may have been obvious when we went through the "Jurimetrics, Committee, Scientific, and so on" example. The purpose of overlapping was to left-justify each term whether combined or left alone, so that it could be stored alphabetically in an IS&R system. In this way, no term is hidden from the search by reason of being forever concealed in second position in storage. We did install a jump switch in NEXUS II, so that we can eliminate overlapping, if desired. If mere subject-headings are required, overlapping is of no value; but if the option for a free-term search is needed, the overlapping feature allows storage and search exposure of each individual term.

Two very different corpora were run against NEXUS II. The first was a collection of computer program descriptions which was assembled for the Scientific Master Programming System (published as "Information Storage and Retrieval Computer Program Index, GDC-DBA68-003). The second was a series of documents from the NASA Tape System collection.

The program descriptions consisted of abstracts of what each particular program was intended to do, and how it operated. Each description also had a short name, a title, the computer language used, the name of the responsible programmer and the responsible engineer, and a set of terms used to index the description.

The abstract portion of each description was used to supply NEXUS II with material to work with. The abstracts were first processed through an auto-indexer to produce lists of terms. These lists were next presented to NEXUS II and then printed out for analysis after the term-binding operations were performed. NEXUS II was run two ways; with and without the overlapping feature.

The program worked well with this material, with one exception. The suffixal morpheme carried by the third person singular, present tense verb,

"-s", has the same physical appearance as the plural morpheme, "-s".
Since the computer can't tell the difference, there occurred some bound
terms that were somewhat less than rife with meaning; for example,
"Program Calculates", "Computes", "Program Generates", "Program Uses".
Although these odd combinations could be avoided by employing a different
writing style when producing the abstracts, we are not concerned with pre-
conditioning a corpus, rather with handling it in whatever form we happen to
find it. The above combinations can certainly be tolerated, however, since
they have no effect on the other precoordinations. Still, their value in a fu-
ture search may be predicted as slight. A NEXUS II processed record of the
computer program descriptions shows:

9916	Computes
9916	Allowable Moments
9916	Axial Loads
9916	Atlas
9916	Tank Skins
9916	Compression Capability
9916	Structures
9916	Tech Memo
9916	5 Function
9916	Ullage
9916	Hydrostatic Pressure
9916	Geometry

The second corpus processed through NEXUS II consisted of titles of
documents from the NASA Tape System. These titles were first auto-indexed
in the same way as the abstracts of the computer program descriptions. The
lists of terms derived in this way were then given the NEXUS II treatment.

21

The rules applied were:

"-s"	Final Position
"-ed"	1st Position
"-ic"	1st Position
"-ly"	1st Position
"-al"	1st Position
"-ing"	1st Position
"-able"	1st Position
"-ive"	1st Position
"-ous"	1st Position
"-ar"	1st Position
"-ary"	1st Position
"-ese"	1st Position
"-ern"	1st Position

Immediately following NEXUS II processing, the same NASA documentation was run using only a sequencing rule, without suffixal morpheme examination. The results of this modified program, SEQS, were merged in alternation with the NEXUS II results, and printed out for analysis.

The first twenty-five NASA documents resulted in 108 good precoordinations out of 124 for NEXUS II, and 73 good precoordinations out of 113 for SEQS.

Some term sets are shown using NEXUS II, SEQS, and human analysis:

NEXUS II		SEQS		HUMAN	
9993	Development Thin-Film	9993	Development Thin-Film	9993	Development Thin-Film Space-
9993	Space-Charge	9993	Space-Charge Limited		Charge Limited
9993	Limited Triode				Triode
9993	Final Report	9993	Triode Final	9993	Final Report Mar.
9993	Mar. 1965	9993	Report Mar.		1965 – Jun. 1966
9993	Jun. 1966	9993	1965 Jun.		
		9993	1966		

22

NEXUS II		SEQS		HUMAN	
9988	Electron Impact	9988	Electron Impact	9988	Electron Impact
9988	Broadening	9988	Broadening		Broadening
	Isolated Ion		Isolated	9988	Isolated Ion Lines
9988	Lines	9988	Ion Lines		
9996	Man System	9996	Man System	9996	Man System
9996	Criteria	9996	Criteria Extra-		Criteria
9996	Extraterrestrial		terrestrial	9996	Extraterrestrial
	Roving Vehicles	9996	Roving Vehicles		Roving Vehicles
9996	Phase IB	9996	Phase IB	9996	Phase IB
9996	Lunex II	9996	Lunex II	9996	Lunex II Simula-
9996	Simulation Interim	9996	Simulation		tion
9996	Technical Report		Interim	9996	Interim Techni-
		9996	Technical Report		cal Report
9998	Bodies	9998	Bodies Maximum	9998	Bodies
9998	Maximum Lift-	9998	Lift-to-Drag	9998	Maximum Lift-
	to-Drag		Ratio		to-Drag Ratio
9998	Ratio	9998	Hypersonic Flow	9998	Hypersonic Flow
9998	Hypersonic Flow				

SECTION 5

Conclusions

As an exercise in demonstrating the difficulties encountered in handling natural language for computerized information retrieval, the NEXUS experiments have been very successful.

The intent has been to expand upon more or less standard automatic indexing techniques by reestablishing a connection between terms that, when combined, aid the searcher in retrieving a document reference from storage. We have named this process precoordination because of its relationship to coordinate index systems. In a coordinate index the searcher combines terms, looking for a common accession number, thereby indicating their occurrence together in a document description. NEXUS has an application

in precoordinating these terms, when applicable, to save time for the searcher and to ensure a correct coordination and to prevent coordinating terms that give a misleading implication. Precoordinated terms are then, in effect, equivalent to subject headings insofar as they partially express a concept in one or more words in a syntatic construction.

The comparison of NEXUS, and its several linguistically-based rules, with SEQS, and its single rule for sequential linking, has shown that NEXUS is the more efficient of the two approaches. Neither, of course, can compare with human decision power, which has the ability to employ knowledge, past experience, and heuristics. Since we are trying to approach a human intellectual activity using a machine, however, the work of a human will probably always make our results look inferior. We are limited to looking at words primarily as physical entities and then relating these physical features to semantic relationships. There is only so much to work with in English, and that much is not 100% reliable, as we have seen.

We have attempted to use a simple algorithm, and to add to it, or subtract from it, through trial and error. No doubt these rules can be expanded more than they have been, so the program is open to further additions at any time.

The NEXUS II flow chart, Figure 5-1, with a narrative explanation, follows.

The first step at (1) is to read a record, a document term set. Step (2) examines the first term in the set and if there is one, moves through the date test (3), which is a holdover from the legal data collection. Next, the program makes the first suffixal morpheme test (4). If the examined word does not end in "-s", it is held for pairing (5) and the "-ed" counter is set to zero. This counter is used for all first-position suffixal morpheme words, not just for those that end in "-ed". The counter is used to keep track of the amount

24

Figure 5-1. NEXUS II Flow Chart

25

of first-position words that accumulate before a second-position word appears, so that they can all be printed out in a string; e.g., "BINARY DIGITAL CALCULATING MACHINE".

The program then moves to (6) where a next word is looked for. If "no", the word held at (5) is printed as a single term (7) and a return to (2) is made, in turn going to (1) and the next record is begun. If (6) is "yes", the NEXUS I date check is made (8) which results in "yes" back through (7) and then (2) again, or "no", which is governed by Sense Switch 2 (9). Sense Switch 2 can be set to pass an examined work through the tests for "-ing" in first position (10) and in second position (11) in order to prevent coupling of words bearing these suffixes. These tests currently have no value because "-ing" has been established as a fairly reliable first-position suffixal morpheme and therefore must be allowed to stack up with words bearing "-ing" or any of the other * (* refers to NOTE – center of page, Figure 5-1) words. The test has been left in in case it ever appears to be of any future use.

Assuming Sense Switch 2 to be in an "on" position, a "no" answer to (8) proceeds directly to (12) where the held first word receives the first-position test for "-ed". If "yes", the "-ed" counter is incremented and the second word is passed through an "-ed" test (14). A "no" at (12) passes the program directly to (14). If (14) is "no", the second word is tested for presence of any of the other suffixal morphemes qualifying a word for first position (noted as *) (15). If (14) is "no", the second word is tested for presence of any of the other suffixal mor- phemes qualifying a word for first position (noted as *) (15). If (14) is "yes", the first word is tested for an * ending (16). A "no" at (15) moves the program to (17) where the first and second words are printed, the counter is set to zero and a flag, 2, (for later identification as a coupled pair) is placed at the end of the first and second words. This flag is externally suppressed.

26

Passing through an indexer (pointing to the last word of a combination) and moving further to (18), there is a Sense Switch 1, that controls overlapping. This is the feature that assures all terms a left-justified accessibility, by printing terms individually as well as in combinations. With the sense switch off, the program moves to (7) and the last word in the combination is printed alone. With the sense switch on, the program returns to (2) and continues through the record.

Backing up now, to (15). If a "yes" answer is made at (15), the first word is tested for * ending at (16). If "no" at (16), the first word receives an "-ed" test (19) and upon receiving another "no" at (19) the first word is printed alone at (7). If "yes" at either (16) or (19), the "-ed" counter (20) (which also counts * words), is incremented and a test for a next word is encountered at (21). If there is not a next word in the record under examination, each "-ed" (or *) word is printed individually (22) and the counter reset to zero. The program then goes back to (2). If there is a next word in the record, the date test is made (23). If "yes" on (23), the print instruction (22) is applied to all "-ed"/* words, and then back to (2). If "no" on (23), the next word is checked for "-ed", (24) and * (25). Failing both of these tests, all "-ed" and * words are printed in a string (with the last member of the string a non-"-ed"/*) (26). If either of these tests (24), (25) are positive, the program loops back through (2), increments the "-ed" counter and cycles through (21), etc., again.

Let's now go back to the first suffixal morpheme test, the last word "-s" test at (4), and assume a "yes" answer. We then must find out if it is a plural "-s", or part of an * ending, "-ous" (27). If it is "-ous", we then go to (5), and thence through the route just explained above. If it is not "-ous", but a plural "-s", we move to (28) to check for a preceding word. If there is no preceding word, the "-s" word is printed as a single word (29), and back to (2).

If there is a preceding word, the date check (30) goes into effect. If positive, the program moves to (29) and the "-s" word is printed as a single term. If "no" on (30) the test is made "Does preceding word end with '-s'?" (31) which, when "no", moves the program to (32) "Is the preceding word part of a coupled pair?". This is the reason for the flag put at the end of the 1st and 2nd words at (17).

If "yes" at (32) the program shifts to (29) where the "-s" word is printed as a single word. If "no" at (32), the program prints the preceding word with this "-s" word (33). If "yes" at (31), there is a test for a preceding word (34). If "yes" at (34), the date test (35) takes place. If "no" at (34), the program shifts to (29) and prints the "-s" word as a single word, and then goes back to (2). This also occurs when there is a "yes" answer at (35). If "no" at (35), the program goes back to (29) where the "-s" word is printed as a single word.

This is the latest version of NEXUS II. The flow chart has superfluities that haven't been removed. Many instructions could be combined to save operations. But, the intent has been to get this program operating and reported on. The flaws that are obvious are the combining of various rules that apply to "-ed" endings as well as * endings. These rules are to be treated the same. No doubt, other things could be combined to make a more efficient program.

A few suggestions for applying this method should be made. The previous method for auto-indexed terms has been to use them in a "busted-book" or computer-generated coordinate index. The NEXUS-generated subject headings are definitely not suitable for this type of output. The best type of output format would be something approaching what was done for the Aeromedical Evacuation Study Bibliography.[4] That was a subject-heading listing, followed by a full bibliographical entry: author, title, date, series number, and corporate author. Sample entries are shown below:

```
BLOOD PRESSURE
   ROMAN J
   HENRY JP
   MEEHAN JP
   VALIDITY OF FLIGHT BLOOD PRESSURE DATA.
   AEROSPACE MED 36,436-41, MAY 65

BLOOD PRESSURE SENSOR
   RESEARCH + TECHNOLOGY RESUME(S) FOR THE DEVEL.
    OF,
   N) DOPPLER ULTRASONIC BLOOD PRESSURE SENSOR

BLOOD VOLUME
   GRABLE E
   LURUS A
   OSOFSKY M
   THE PREDICTABILITY OF BLOOD VOLUME IN NORMAL
    AIR FORCE PERSONNEL AND THEIR DEPENDENTS.
   MILIT MED 132,114-8, FEB 67

BLOOD VOLUME
   RESEARCH + TECHNOLOGY RESUME(S) FOR THE DEVEL.
    OF,
   Q) INVESTIGATION OF BLOOD VOLUME + GAS
    ALTERATIONS MEASUREMENTS DURING AEROMEDICAL
    EVACUATION
```

This bibliography was machine processed after all input was subjected to human analysis.

The final bibliography consisted of four sections: by subject, by author, by title, and by source. The latter section was an alphabetical sort of the journals, books, papers, and manuals from which the material was taken.

A modification of this form of output has been suggested by Mr. James Moore of 591-0, who was responsible for NEXUS I and II programming. His suggestion involves a sort and printout of each auto-indexed term and beneath each such term is printed the NEXUS precoordinated set of which the term is a member. Beneath this term set would be the bibliographic entry, in entirety. Where terms are not members of a precoordinated set, they are printed alone, followed by the full bibliographic entry. A hand-generated example of this

format would appear like this:

(in "S" portion of alphabet)

SUPERSONIC
COMMERCIAL SUPERSONIC TRANSPORT
    BOEING CO.
    COMMERCIAL SUPERSONIC TRANSPORT PROPOSAL,
    A-111, AIRCRAFT DESCRIPTION.
    D6-2400-9 THE BOEING CO. 15 JAN 64

(in "C" portion of alphabet)

COMMERCIAL
COMMERCIAL SUPERSONIC TRANSPORT
BOEING CO.
COMMERCIAL SUPERSONIC TRANSPORT PROPOSAL, A-111, AIRCRAFT
DESCRIPTION.
D6-2400-9 THE BOEING CO. 15 JAN 64

(in "T" portion of alphabet)

TRANSPORT
COMMERCIAL SUPERSONIC TRANSPORT
BOEING CO.
COMMERCIAL SUPERSONIC TRANSPORT PROPOSAL,
A-111, AIRCRAFT DESCRIPTION.
D6-2400-9 THE BOEING CO. 15 JAN 64

(in "P" portion of alphabet)

PROPOSAL
PROPOSAL A-111
BOEING CO.
COMMERCIAL SUPERSONIC TRANSPORT PROPOSAL,
A-111, AIRCRAFT DESCRIPTION.
D6-2400-9 THE BOEING CO. 15 JAN 64

(in "A" portion of alphabet)

AIRCRAFT
BOEING CO.
COMMERCIAL SUPERSONIC TRANSPORT PROPOSAL,
A-111, AIRCRAFT DESCRIPTION.
D6-2400-9 THE BOEING CO. 15 JAN 64

(in "D" portion of alphabet)

DESCRIPTION
BOEING CO.
COMMERCIAL SUPERSONIC TRANSPORT PROPOSAL,
A-111, AIRCRAFT DESCRIPTION.
D6-2400-9 THE BOEING CO.15 JAN 64

Six subject entries per document reference may seem excessive at first
glance, and there may be more if an abstract is also processed, but roughly
this same approach was used for the Aeromedical Evacuation Bibliography and
was found to be helpful. Unfortunately, the same human analysis that was
employed in processing the input to that program was not completely thorough
in picking up all possible subject headings for sorting. A machine-analysis
system would not suffer from this fallibility.

<div align="center">SECTION 6</div>

Recommendations

Linguistics is becoming more and more recognized as a basic research
area in information retrieval. The problem of document analysis and index-
term selection is the most fundamental activity of all in the cycle of document-
to-storage-to-document user, which is what information retrieval really
amounts to.

No matter how sophisticated the storage medium might be, no matter
how fast the computer can sift through a data bank searching for information,
an information retrieval system is only as good as its contents.

Linguistics, as applied to information retrieval, is concerned with
improving the input function in the design of automatic indexing, abstracting
and classification methods. The kind of linquistics used in these applications
is limited to the written word or the analysis and manipulation of graphemes.
Linquistics, in a general sense, concerns itself with speech sounds, from which
a graphemic representation of a language is one step removed. If the day ever

comes that a computer can more efficiently accept the spoken word than the written word linguistics, in a fuller sense, will be found applicable. There will probably be interim improvements in methods for computer input that will predate voice input, however.

Such input devices as optical scanners and page readers may make a long-awaited appearance, for practical purposes, before people can talk to a computer in any application other than an experimental one. If there is any doubt of the superiority of the spoken word over the written as an information carrier, one merely has to read a television jingle or such a phrase as, "very interesting," heard on a popular TV program, to realize that the suprasegmental phonemes of stress, pitch, juncture and even accent in the dialect sense, completely lost in the written word, are very much present and necessary in the spoken word.

Getting back to the kind of linguistics with which we have been directly concerned, we have been devising rules for joining together two or more words to make up a phrase. The rules are activated when one or more characters (graphemes) are found at the ends of words (suffixal morphemes) that have an effect on the word's connectability to other words in a sequence (syntax). These rules work every time. There is no decision maker involved allowing a sometimes exemption to a rule. Since the rules are of a general-purpose kind, they are set up to operate on the most frequent conditions. The exceptions to these conditions that occur occasionally are merely tolerated. No attempt has been made to set up ad hoc rules to cover them. It so happens, unfortunately, that the name "Information Retrieval" is one of these exceptions and would not be produced as a combination by the NEXUS program.

Although the NEXUS method is far from perfect, even in its present state it is reasonably workable as a subject-heading generator. Its consistency of operation, of course, exceeds human processing; an advantage

32

in some respects and a disadvantage in others, as already pointed out. Research of this type is not intended to produce a panacea that will solve all natural-language-input problems, but is intended to shed a little more light on language manipulation by computer and perhaps take a few tentative steps towards a solution of these problems. Hopefully, this research has been successful to that extent.

The following pages are S-C 4020 microfilm hard copies showing a comparison between NEXUS processing and SEQS processing of NASA Linear Tape System documents.

The NASA System has been previously converted to the IS&R format[5] for more efficient information searching. The titles of a 1000-document corpus were first auto-indexed using IS&R SIMPL programming techniques. The product of the auto-indexing operation is shown in the first column on the left of each page. It consists of a list of content words remaining after the function words were deleted from the title.

The middle column is a list of the word combinations created by the NEXUS II program employing linguistic rules and sequence rules.

The SEQS column lists the combinations formed by using sequence rules alone. Here every two terms are connected as they occur in syntactical order.

NEXUS
A LINGUISTIC TECHNIQUE
FOR
PRECOORDINATION
DEC 1968

TITLE 'GEODETIC JUNCTION OF FRANCE AND NORTH AFRICA BY SYNCHRONIZED
       PHOTOGRAPHS TAKEN FROM ECHO I SATELLITE

AUTO-INDEXED TERMS	NEXUS	SEQS
1.GEODETIC	1.GEODETIC JUNCTION	1.GEODETIC JUNCTION
2.JUNCTION	2.FRANCE	2.FRANCE NORTH
3.FRANCE	3.NORTH AFRICA	3.AFRICA SYNCHRONIZED
4.NORTH	4.SYNCHRONIZED PHOTOGRAPHS	4.PHOTOGRAPHS ECHO
5.AFRICA	5.ECHO SATELLITE	5.SATELLITE
6.SYNCHRONIZED		
7.PHOTOGRAPHS		
8.ECHO		
9.SATELLITE		

* * * * * * *

TITLE 'LIMITS OF HEAD-WAVE AMPLITUDES FOR SHORT SPREADS FROM VARIOUS CHARGE
       SIZES, BLASTING CAPS, AND 45-KG WEIGHT DROP

AUTO-INDEXED TERMS	NEXUS	SEQS
1.LIMITS	1.LIMITS	1.LIMITS HEAD-WAVE
2.HEAD-WAVE	2.HEAD-WAVE AMPLITUDES	2.AMPLITUDES SHORT
3.AMPLITUDES	3.SHORT SPREADS	3.SPREADS CHARGE
4.SHORT	4.CHARGE SIZES	4.SIZES BLASTING
5.SPREADS	5.BLASTING CAPS	5.CAPS 45-KG
6.CHARGE	6.45-KG WEIGHT	6.WEIGHT DROP
7.SIZES	7.DROP	
8.BLASTING		
9.CAPS		
10.45-KG		
11.WEIGHT		
12.DROP		

34

TITLE 'STRUCTURE AND COMPOSITION OF THE SOUTHERN COULEE, MONO CRATERS,
       CALIFORNIA - A PUMICEOUS RHYOLITE FLOW

AUTO-INDEXED TERMS	NEXUS	SEQS
1.STRUCTURE	1.STRUCTURE COMPOSITION	1.STRUCTURE COMPOSITION
2.COMPOSITION	2.SOUTHERN COULEE	2.SOUTHERN COULEE
3.SOUTHERN	3.MONO CRATERS	3.MONO CRATERS
4.COULEE	4.CALIFORNIA	4.CALIFORNIA PUMICEOUS
5.MONO	5.PUMICEOUS RHYOLITE	5.RHYOLITE FLOW
6.CRATERS	6.FLOW	
7.CALIFORNIA		
8.PUMICEOUS		
9.RHYOLITE		
10.FLOW		

* * * * * * *

TITLE 'ON THE THERMODYNAMICS OF ELASTIC MATERIALS

AUTO-INDEXED TERMS	NEXUS	SEQS
1.THERMODYNAMICS	1.THERMODYNAMICS	1.THERMODYNAMICS ELASTIC
2.ELASTIC	2.ELASTIC MATERIALS	2.MATERIALS
3.MATERIALS		

* * * * * * *

TITLE 'A FINITE DIFFERENCE METHOD FOR COMPUTING UNSTEADY, INCOMPRESSIBLE, LAMINAR
       BOUNDARY LAYER FLOWS   FINAL SCIENTIFIC REPORT

AUTO-INDEXED TERMS	NEXUS	SEQS
1.FINITE	1.FINITE DIFFERENCE	1.FINITE DIFFERENCE
2.DIFFERENCE	2.METHOD	2.METHOD COMPUTING
3.METHOD	3.COMPUTING UNSTEADY	3.UNSTEADY
4.COMPUTING	4.INCOMPRESSIBLE LAMINAR BOUNDARY	INCOMPRESSIBLE
5.UNSTEADY	LAYER FLOWS	4.LAMINAR BOUNDARY
6.INCOMPRESSIBLE	5.FINAL SCIENTIFIC REPORT	5.LAYER FLOWS
7.LAMINAR		6.FINAL SCIENTIFIC
8.BOUNDARY		7.REPORT
9.LAYER		
10.FLOWS		
11.FINAL		
12.SCIENTIFIC		
13.REPORT		

35

CORPUS OF
NASA TAPE SYSTEM
DOCUMENTS

NEXUS
A LINGUISTIC TECHNIQUE
FOR
PRECOORDINATION
DEC 1968

TITLE 'INTERPLANETARY MONITORING PLATFORM IMP III - EXPLORER XXVIII   INTERIM
       FLIGHT REPORT NO. 2

AUTO-INDEXED TERMS	NEXUS	SEQS
1.INTERPLANETARY	1.INTERPLANETARY MONITORING PLATFORM	1.INTERPLANETARY MONITORING
2.MONITORING	2.IMP III	2.PLATFORM IMP
3.PLATFORM	3.EXPLORER XXVIII	3.III EXPLORER
4.IMP	4.INTERIM FLIGHT	4.XXVIII INTERIM
5.III	5.REPORT 2	5.FLIGHT REPORT
6.EXPLORER		6.2
7.XXVIII		
8.INTERIM		
9.FLIGHT		
10.REPORT		
11.2		

*   *   *   *   *   *   *

TITLE 'ELECTROMAGNETIC SCATTERING CHARACTERISTICS OF A METEOROLOGICAL RADAR ANGEL
       MODEL BY METHODS OF PHYSICAL OPTICS

AUTO-INDEXED TERMS	NEXUS	SEQS
1.ELECTROMAGNETIC	1.ELECTROMAGNETIC SCATTERING CHARACTERISTICS	1.ELECTROMAGNETIC SCATTERING
2.SCATTERING		2.CHARACTERISTICS METEOROLOGICAL
3.CHARACTERISTICS	2.METEOROLOGICAL RADAR ANGEL	
4.METEOROLOGICAL	3.MODEL METHODS	3.RADAR ANGEL
5.RADAR	4.PHYSICAL OPTICS	4.MODEL METHODS
6.ANGEL		5.PHYSICAL OPTICS
7.MODEL		
8.METHODS		
9.PHYSICAL		
10.OPTICS		

CORPUS OF
NASA TAPE SYSTEM
DOCUMENTS

NEXUS
A LINGUISTIC TECHNIQUE
FOR
PRECOORDINATION
DEC 1968

TITLE 'EXTENSIONAL MECHANICAL PROPERTIES OF POLYESTER AND POLYETHER BASED
        POLYURETHANES

AUTO-INDEXED TERMS	NEXUS	SEQS
1.EXTENSIONAL	1.EXTENSIONAL MECHANICAL PROPERTIES	1.EXTENSIONAL MECHANICAL
2.MECHANICAL	2.POLYESTER POLYETHER POLYURETHANES	2.PROPERTIES POLYESTER
3.PROPERTIES		3.POLYETHER
4.POLYESTER		POLYURETHANES
5.POLYETHER		
6.POLYURETHANES		

* * * * * * *

TITLE 'PROPAGATION OF HARMONIC WAVES IN COMPOSITE CIRCULAR CYLINDRICAL SHELLS.
        PART I - THEORETICAL INVESTIGATION

AUTO-INDEXED TERMS	NEXUS	SEQS
1.PROPAGATION	1.PROPAGATION	1.PROPAGATION HARMONIC
2.HARMONIC	2.HARMONIC WAVES	2.WAVES COMPOSITE
3.WAVES	3.COMPOSITE	3.CIRCULAR CYLINDRICAL
4.COMPOSITE	4.CIRCULAR CYLINDRICAL SHELLS	4.SHELLS PART
5.CIRCULAR	5.PART	5.THEORETICAL
6.CYLINDRICAL	6.THEORETICAL INVESTIGATION	INVESTIGATION
7.SHELLS		
8.PART		
9.THEORETICAL		
10.INVESTIGATION		

* * * * * * *

TITLE 'MEAN MOLECULAR MASS AND SCALE HEIGHTS OF THE UPPER ATMOSPHERE

AUTO-INDEXED TERMS	NEXUS	SEQS
1.MEAN	1.MEAN	1.MEAN MOLECULAR
2.MOLECULAR	2.MOLECULAR MASS	2.MASS SCALE
3.MASS	3.SCALE HEIGHTS	3.HEIGHTS UPPER
4.SCALE	4.UPPER ATMOSPHERE	4.ATMOSPHERE
5.HEIGHTS		
6.UPPER		
7.ATMOSPHERE		

```
        CORPUS OF                        NEXUS
      NASA TAPE SYSTEM            A LINGUISTIC TECHNIQUE
        DOCUMENTS                          FOR
                                     PRECOORDINATION
                                        DEC 1968
```

TITLE 'MAGNETIC FIELD MEASUREMENTS IN INTERPLANETARY SPACE

```
AUTO-INDEXED                    NEXUS                        SEQS
  TERMS

  1.MAGNETIC          1.MAGNETIC FIELD              1.MAGNETIC FIELD
  2.FIELD             2.MEASUREMENTS                2.MEASUREMENTS
  3.MEASUREMENTS      3.INTERPLANETARY SPACE          INTERPLANETARY
  4.INTERPLANETARY                                  3.SPACE
  5.SPACE
                        *   *   *   *   *   *   *
```

TITLE 'MEASUREMENTS OF MAGNETIC PROBES DURING THE PREHEATING PHASE OF A SPINDLE
        CUSP EXPERIMENT

```
AUTO-INDEXED                    NEXUS                        SEQS
  TERMS

  1.MEASUREMENTS      1.MEASUREMENTS                1.MEASUREMENTS MAGNETIC
  2.MAGNETIC          2.MAGNETIC PROBES             2.PROBES PREHEATING
 .3.PROBES            3.PREHEATING PHASE            3.PHASE SPINDLE
  4.PREHEATING        4.SPINDLE CUSP                4.CUSP EXPERIMENT
  5.PHASE             5.EXPERIMENT
  6.SPINDLE
  7.CUSP
  8.EXPERIMENT
                        *   *   *   *   *   *   *
```

TITLE 'THE EFFECTS OF THE LAUNCH VEHICLE ON SPACECRAFT DESIGN

```
AUTO-INDEXED                    NEXUS                        SEQS
  TERMS

  1.EFFECTS           1.EFFECTS                     1.EFFECTS LAUNCH
  2.LAUNCH            2.LAUNCH VEHICLE              2.VEHICLE SPACECRAFT
  3.VEHICLE           3.SPACECRAFT DESIGN           3.DESIGN
  4.SPACECRAFT
  5.DESIGN
```

```
       CORPUS OF                      NEXUS
    NASA TAPE SYSTEM            A LINGUISTIC TECHNIQUE
       DOCUMENTS                        FOR
                                  PRECOORDINATION
                                     DEC 1968
```

TITLE 'A TWO-TEMPERATURE STATISTICAL MODEL FOR PARTICLE PRODUCTION AT HIGH
        ENERGIES

```
AUTO-INDEXED              NEXUS                        SEQS
   TERMS

   1.TWO-TEMPERATURE   1.TWO-TEMPERATURE           1.TWO-TEMPERATURE
   2.STATISTICAL       2.STATISTICAL MODEL            STATISTICAL
   3.MODEL             3.PARTICLE PRODUCTION       2.MODEL PARTICLE
   4.PARTICLE          4.HIGH ENERGIES             3.PRODUCTION HIGH
   5.PRODUCTION                                    4.ENERGIES
   6.HIGH
   7.ENERGIES
                        *   *   *   *   *   *   *
```

TITLE 'PROPAGATION OF SPHERICAL WAVES THROUGH AN INHOMOGENEOUS MEDIUM CONTAINING
        ANISOTROPIC IRREGULARITIES

```
AUTO-INDEXED              NEXUS                        SEQS
   TERMS

   1.PROPAGATION       1.PROPAGATION              1.PROPAGATION SPHERICAL
   2.SPHERICAL         2.SPHERICAL WAVES          2.WAVES INHOMOGENEOUS
   3.WAVES             3.INHOMOGENEOUS MEDIUM     3.MEDIUM CONTAINING
   4.INHOMOGENEOUS     4.CONTAINING ANISOTROPIC   4.ANISOTROPIC
   5.MEDIUM              IRREGULARITIES              IRREGULARITIES
   6.CONTAINING
   7.ANISOTROPIC
   8.IRREGULARITIES
                        *   *   *   *   *   *   *
```

TITLE 'PULSED ELECTRICAL POWER GENERATION FROM MAGNETICALLY LOADED EXPLOSIVES

```
AUTO-INDEXED              NEXUS                        SEQS
   TERMS

   1.PULSED            1.PULSED ELECTRICAL POWER GENERATION   1.PULSED ELECTRICAL
   2.ELECTRICAL        2.MAGNETICALLY LOADED EXPLOSIVES       2.POWER GENERATION
   3.POWER                                                    3.MAGNETICALLY LOADED
   4.GENERATION                                               4.EXPLOSIVES
   5.MAGNETICALLY
   6.LOADED
   7.EXPLOSIVES
```

```
        CORPUS OF                         NEXUS
      NASA TAPE SYSTEM            A LINGUISTIC TECHNIQUE
         DOCUMENTS                         FOR
                                    PRECOORDINATION
                                      DEC 1968
```

TITLE 'ELECTRICAL PULSES FROM HELICAL AND COAXIAL EXPLOSIVE GENERATORS

```
AUTO-INDEXED                 NEXUS                         SEQS
  TERMS

   1.ELECTRICAL      1.ELECTRICAL PULSES           1.ELECTRICAL PULSES
   2.PULSES          2.HELICAL COAXIAL EXPLOSIVE    2.HELICAL COAXIAL
   3.HELICAL            GENERATORS                  3.EXPLOSIVE GENERATORS
   4.COAXIAL
   5.EXPLOSIVE
   6.GENERATORS
                        *   *   *   *   *   *   *
```

TITLE 'PLASMA COMPRESSION BY EXPLOSIVELY PRODUCED MAGNETIC FIELDS

```
AUTO-INDEXED                 NEXUS                         SEQS
  TERMS

   1.PLASMA          1.PLASMA COMPRESSION          1.PLASMA COMPRESSION
   2.COMPRESSION     2.EXPLOSIVELY PRODUCED MAGNETIC  2.EXPLOSIVELY PRODUCED
   3.EXPLOSIVELY        FIELDS                       3.MAGNETIC FIELDS
   4.PRODUCED
   5.MAGNETIC
   6.FIELDS
                        *   *   *   *   *   *   *
```

TITLE 'EFFECTIVE FEEDING SYSTEMS FOR PULSE GENERATORS

```
AUTO-INDEXED                 NEXUS                         SEQS
  TERMS

   1.EFFECTIVE       1.EFFECTIVE FEEDING SYSTEMS   1.EFFECTIVE FEEDING
   2.FEEDING         2.PULSE GENERATORS            2.SYSTEMS PULSE
   3.SYSTEMS                                        3.GENERATORS
   4.PULSE
   5.GENERATORS
```

References

1. M. A. Newcomb, R. A. Benson, Technique for the Automatic Generation of Bibliographies (A Biomedical Application), San Diego, General Dynamics/Convair, 1965.

2. J. A. Sanford and Frederick R. Theriault, Problems in the Application of Uniterm Coordinate Indexing, College and Research Libraries, January 1956, v. XVIII, pp. 19-23.

3. R. A. Benson, Linguistic Experiments at Convair, San Diego, General Dynamics/Convair, 1967, pp. 5-1 to 5-3.

4. Medical Systems Analysis, Aeromedical Evacuation System, San Diego, General Dynamics Convair, 1 August 1968.

5. W. F. MacDonald, Conversion of Large-Scale IS&R Systems for General-Purpose Operations, Convair division of General Dynamics, San Diego, December 1968.