ALLOY : An Amalgamation of Expert, Linguistic and Statistical Indexing Methods

Leslie P. Jones	Cary deBessonet	Sukhamay Kundu
Computer Science Department	Southern University Law School	Computer Science Department
Louisiana State University	Baton Rouge, Louisiana	Louisiana State University
Baton Rouge, Louisiana		Baton Rouge, Louisiana

Abstract

In this paper we report progress on the development of ALLOY, a system that simplifies automatic document indexing and retrieval by combining techniques from several different approaches: expert, linguistic and statistical. The system is being designed to allow a panel of experts to create an ALLOY system for a given field by providing the necessary input that ALLOY needs to automatically index documents and to set up a convenient user interface. The input provided by the experts includes a hierarchy of concepts and an expert dictionary. The amount of information that the panel must provide for given field is considerably less than the amount required to build a complete thesaurus or knowledge base about that field.

1. Introduction

Classical keyword retrieval [Salton83] using statistical methods to match documents and queries has been used and studied for many years. In recent years, researchers have been considering the possibility of using linguistic and expert systems approaches to document indexing and retrieval, frequently in some combination. A linguistic approach attempts to use knowledge of the English language to interpret documents and queries and thereby match them effectively (see, e.g., [Fagan87], [Croft87], [Dillon83], [Smeaton83]). An expert systems approach attempts to incorporate the knowledge of an expert in a subject area into a system that users can easily use to satisfy their individual retrieval needs (see, e.g., [Croft86], [Tong87a], [Tong87b] and [DiBenigno86]). Linguistic methods and expert methods can be combined by using expert knowledge in the interpretation of the documents and queries. Two of the major problems encountered in the operation of setting up an expert system for document retrieval in a field are:

a) the large volume of information that must be stored
b) the ambiguity of topics and their relationships [Salton87].

In our work, we attempt to make it easier to incorporate expert knowledge into a system for retrieving documents in a specific field. This is accomplished by providing linguistic and statistical tools to aid in the operation of setting up an indexing system that is based on knowledge about the field. In particular, we believe that automatic indexing and retrieval can benefit substantially from expert knowledge in a field without having to "understand" the entire field (as suggested by [Croft87]). Also, by establishing an environment that can be used to specify systems about many different subject areas, we can provide a consistent user interface that makes it easy for a user to move from one field to another.

C 1988 ACM 0-89791-274-8 88 0600 0191 $ 1,50

In recent years, there has been considerable activity in the area of building expert system shells. The shell of ALLOY has two parts: a simple language for the expression of a conceptual hierarchy about a field and a syntax for the expression of an expert dictionary about that field. An expert panel sets up an ALLOY system for a given field as follows. First, the panel defines a conceptual hierarchy consisting of a set of concepts linked by "subconcept" relationships. Concepts are not regarded as sets; they are merely subject areas such as "information retrieval" or "relevance feedback". The concepts form a tree with one concept at each node. The most general concept is at the root and the offspring of a concept are its subconcepts. The usual tree terminology will be applied when discussing the conceptual hierarchy. The hierarchy may also have additional links between concepts defined by "cross-references". Specifically, a cross-reference may be given from a concept C_1 to a concept C_2 if and only if neither C_1 nor C_2 is an ancestor of the other. Each concept and cross-reference is given a name. The ALLOY system is then be able to index documents by determining how strongly they relate to each of the concepts in the hierarchy. The first implementation of ALLOY is being designed to index documents, but not actually store them. Later versions will store documents and provide for standard keyword retrieval at the nodes in the hierarchy.

It is absolutely essential that each leaf in the hierarchy have distinctive terminology naturally associated with it. For each leaf in the hierarchy, the panel provides a collection of representative documents that contains the terminology typical of the leaf. This terminology consists of keywords and phrases that are often used in documents about the leaf. ALLOY then compiles the words that are used in the union of the collections and the panel provides information regarding the stems and possible parts of speech of these words for the expert dictionary. The form of the dictionary, and its use in identifying key phrases, is discussed extensively in [Jones88a]. ALLOY employs the dictionary to study each collection and to extract the keywords and phrases used in the collections. This associates a list of words and phrases with each leaf in the hierarchy. The lists are compared, with automatic help from ALLOY, to determine the words and phrases that best categorizes each leaf in the hierarchy. These words and phrases are then be used to index documents as they are entered into the system. The sets of keywords and phrases can be augmented at any time, and ALLOY provides a module to help with this operation.

The user is provided with a browsing facility that allows movement through the conceptual hierarchy and along the cross-references. The system also supports the inclusion of additional explanatory information at the nodes in the hierarchy. At each node the user is able to see the following information:

a) An explanation of the concept of the node
b) The (name of the) parent concept of the node
c) The subconcepts of the node
d) All concepts cross-referenced from the node
e) All concepts cross-referencing the node
f) A list of keywords and phrases that relate to the node
g) A list of references to documents that relate to the node

When the system is augmented to store documents, the user will be able to browse the conceptual hierarchy and then apply classical keyword retrieval at the nodes by searching the list of documents associated with a node for combinations of words and/or phrases. The first implementation does not use cross-references for indexing purposes; they are present only to help the users. Our first major application area will be in the retrieval of legal documents.

This paper outlines our plans for the ALLOY system. Section 2 describes an existing module of ALLOY, INDEXD, that is used to find keywords and phrases within a document. INDEXD is capable of a certain amount of syntactic analysis when provided with an English dictionary containing information on word stems and parts of speech. Section 3 deals with the operation of defining the conceptual hierarchy and cross-references. An simple example of a hierarchy is given. Section 4 describes the use of INDEXD to analyze the collections of representative documents, selected by the expert panel, to determine sets of keywords and phrases that typify the concepts in the hierarchy. Section 5 describes the operation of using INDEXD and sets of keywords and phrases to automatically index a large body of input documents. Sections 3, 4 and 5 contain mathematical formulas for weighting of words, phrases and documents; these formulas will be the subject of further study as more of ALLOY is implemented and tested. Section 6 presents a brief summary and collection of acknowledgements.

2. The INDEXD program

One of the most important components of ALLOY is an automatic phrase indexing program called INDEXD. The retrieval aspects of this program are discussed extensively in [Jones88a] while the algorithms and data structures of the program are discussed in [Jones88b]. This program locates repeated phrases in a document, gathers statistical information about them, and ranks them according to an estimate of their value as index phrases. Phrases are ranked in such a way that frequently occurring phrases that contain several frequently occurring words are given a high ranking. INDEXD incorporates a dictionary for stemming, preweighting of words and validation of syntax of output phrases. Sample output of INDEXD is given below. A large dictionary has already been built for INDEXD. It contains many common English words and legal terms. We believe that a standard English dictionary can be developed, and modified to accommodate specific subject areas. The problem of preweighting words in the dictionary to emphasize those that have great power to distinguish among concepts is discussed in Section 4.

It is well known (see, e.g., [Salton83]) that the mere fact that a single word occurs frequently in a document does not alone mean that the word is a good content indicator for the document. For obvious reasons, certain words have no value as content indicators. Other words may distinguish between documents in some contexts but not in other contexts. INDEXD is an efficient program that finds keywords and phrases within a document and gathers statistical information about them. Our techniques for assigning values to the keywords and phrases provide results that are promising in the sense that phrases with high values express concepts that are consistent with prior knowledge of the content of the document. INDEXD appears to be a good tool for compiling statistical information about a document or group of documents and thus is an important component of the ALLOY system.

The linguistic analysis performed by INDEXD makes use of a dictionary which can be adapted to new subject areas by simple addition of information. This dictionary consists of a long series of entries of the following form:

a) a word stem, s
b) a preweight for the stem
c) a set of lists, with each list of the following form:

    i) a part of speech, p
    ii) a list of words having stem s and part of speech p

The preweight of a stopword is 0, and the preweight of any other word is 1 until further analysis, described below, has been performed. INDEXD has a built-in set of valid phrase forms, each defined by a sequence of parts of speech (e.g. adverb adject noun). A phrase is a sequence of words such that some choice of parts of speech for the words produces one of the valid sequences stored in INDEXD. After completely reading an input document, INDEXD assigns a value to each word and phrase in the document as follows. The value of a word is the frequency of occurrence of the stem of that words times the preweighted value of the stem. The program assigns a value to a given phrase as follows. Let W be the sum of the values of the words in the phrase; words that appear more than once in the phrase are included only once in W. Let F equal the frequency of occurrence of the phrase in the document and let N equal the number of distinct non-stopwords in the phrase. Then the value of the phrase is defined to be WF^2N^2. (This choice of value formula will be the subject of further research.) INDEXD also provides a list of words found in the document but not included in its dictionary. The value of a word in the output of INDEXD is the value of the word as a phrase of length one.

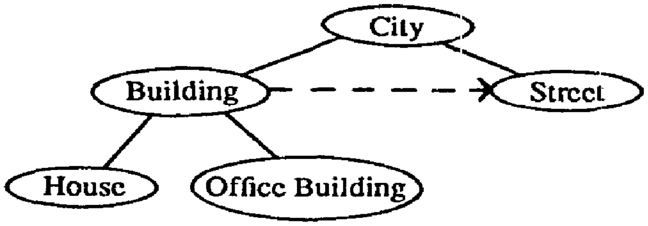
For demonstration purposes, INDEX was run on Book 3, Title 7 of the Louisiana Civil Code [Louisiana] with a maximum phrase length of four. The resulting file of phrases was then sorted into decreasing order of phrase value. A portion of the sorted output is given in Figure 1. The segment of output gives a clear indication of the subject matter of the document: it deals with the contract of sale. Further experiments with INDEXD are reported in [Jones88a] and [Jones88b]. INDEXD is currently being augmented to make it better able to cluster related phrases and to remove redundancy among phrases. Finally, INDEXD is also being augmented to produce a probabilistic model of the input document that determines the likelihood that any word will be followed by any other word.

3. The Conceptual Hierarchy and Cross-references

The first step in preparing an ALLOY system for use in a particular subject area is the specification of the conceptual hierarchy. We will discuss the handling of a conceptual hierarchy, both from the point of view of the panel and from the point of view of a user. The definition of a hierarchy called "City" in a prototypical form of the ALLOY script language for hierarchies is given below.

```
concept("City", "Large collection of people, their dwelling and work places")
concept("Building", "Structure that shelters people and material goods")
subconcept("Building", "contained in", "City")
concept("House", "Dwelling for one or a few families")
subconcept("House", "is a small", "Building")
concept("Office building", "Building containing corporate offices and businesses")
subconcept("Office building", "is a large", "building")
concept("Street", "Paved surface for transportation")
subconcept("Street", "lies in", "City")
xref("Building", "has address on", "Street")
```

The hierarchy, with cross-references, may be expressed diagrammatically as follows:



An ALLOY user would then see the following information about "Building" when at that node:

Building:	Structure that shelters people and material goods
Parent:	Building contained in City
Subconcept:	House is a small Building
Subconcept:	Office building is a large Building
Xref to:	Building has address on Street
Xref from:	

Accessible concepts:

City	(parent)
House	(subconcept)
Office building	(subconcept)
Street	(xref to)

The user will then select one of several operations from a menu: move to an accessible node, see keywords and phrases associated with node, see references to documents associated with node, get help or quit.

4. Preprocessing of Representative Documents

Assume that the panel has developed a hierarchy and that concepts $\{L_1, L_2, \ldots, L_N\}$ are the leaves of the hierarchy. We attempt to establish the terminology typical of each leaf in the following way. An initial group, **D**, of documents is chosen in such a way that the experts agree that

a) Each document $D \in \mathbf{D}$ belongs strongly to exactly one of the leaves.
b) Each leaf is represented by several documents that cover the breadth of the subject.

In reference to condition a), we expect that each document is devoted almost entirely to one leaf. If necessary, existing documents about several concepts can be subdivided into pieces which satisfy condition a). It is not our purpose to index the documents in **D** at this point; rather, we are trying to analyze them to gain information that will be used for indexing later documents. In reference to b), we note that it is not necessary to cover every theory or opinion about a concept; it is sufficient to discover the terminology typically associated with the concept. The documents in each concept are concatenated into one super-document for the concept. The program INDEXD is then run on the concatenation of all the super-documents to see if any words exist in the super-documents which are not already in the dictionary. Any such words are added to the dictionary.

Once all words in the super-documents have been added to the dictionary, the program INDEXD is run on each super-document individually to provide a list of phrases and their values. (We will regard a word as a phrase of length 1.) The list of phrases for the super-document corresponding to concept L_i is denoted S_i. Each list S_i is trimmed by the panel. First, they delete phrases with low values as computed by INDEXD. Next, the experts delete any other phrases that they believe are superfluous. A superfluous phrase could be a phrase that has a valid syntactic interpretation as determined by INDEXD, but does not have any reasonable semantic interpretation. The values of phrases as determined by INDEXD are then used to define new phrase values called "inverse frequencies". Specifically, the inverse frequency of a phrase, p, within a list, S_i, is

$$\frac{\text{value of p in } S_i}{\sum \text{values of p in all } S_j\text{'s}}$$

Clearly the inverse frequency of a phrase in a super-document is 1 if and only if that phrase occurs only in that one super-document. For a given phrase, the sum of the inverse frequencies of that phrase taken over all the super-documents is 1. The inverse frequency of a phrase in a super-document is then used as the value of that phrase in the super-document. A threshold value is selected, and phrases with value less than this threshold are deleted. The quality of the initial hierarchy and choice of representative documents is evaluated at this time by looking at the values of the phrases in each list S_i. If one or more S_i's exist which have very few phrases with high values, the choices should be re-evaluated.

Finally, the panel examines each list, S_i, and determines whether or not S_i can be partitioned into subsets, each of which indicates a subconcept of L_i. If S_i cannot be split in this way, the phrase list associated with the node L_i is merely S_i. If the phrase list is partitioned, a new concept name is created for each subset in the partition, and the subsets are associated with their respective concepts names. (A new leaf is created in the hierarchy for each new concept name.) At this point, each leaf has an associated phrase list. Each non-leaf node in the hierarchy is given an associated phrase list consisting of the union of all phrase lists for nodes below the given node in the hierarchy. The result is a hierarchy represented in the desired form. Finally, the dictionary is modified by assigning preweights to words. The preweight of a word, w, is taken to be

$$\frac{1}{\text{number of } S_i\text{'s containing w}}$$

or 0 if w is in no S_i.

5. Indexing of New Documents

After the operations of Section 4, the dictionary contains the necessary information to find key phrases in documents within the area of the panel's expertise. We describe the operation of indexing new documents. The operation should be tested on the original input documents and on a selected group of new documents. Let the leaves in the hierarchy be denoted $\{L_1, L_2, \dots, L_M\}$ and their respective words lists be denoted $\{S_1, S_2, \dots S_M\}$. The sets S_i are put together as follows. A set S is formed consisting of pairs of the following form:

(phrase p, list of pairs (leaf L_i with $p \in S_i$, inverse frequency of p in S_i))

When a new document, D, is entered on the ALLOY system, INDEXD is run on D, resulting in a set of phrases, S_D. The phrases are then put through a separate module of ALLOY that produces a vector

$$(v_1, v_2, \dots, v_M)$$

in which each number v_i satisfies $0 \le v_i$ and represents the relative extent to which document D belongs to the concept L_i. When ALLOY is in use, a user will be able to see this vector whenever the reference to document D is recovered. Each v_i is computed as follows. (The vector will be normalized at the end so that its components sum to 1.) The set S is searched for each phrase in S_D. Each time a phrase p in S_D is found, each v_i corresponding to a concept in the list of leaves for that phrase is incremented by the quantity

(value of p in S_D computed by INDEXD)*(inverse frequency of p in S_i)

When all phrases have been processed, the vector is normalized by dividing all of its components by the sum of its components.

6. Summary and Acknowledgements

We have presented plans for a document indexing and retrieval system called ALLOY. The system combines expert, linguistic and statistical methods to provide a general environment for the specification of a system for retrieval of documents within a specific subject area. To set up an ALLOY system in an area, a expert panel provides a conceptual hierarchy and a collection of documents relevant to the leaves in that hierarchy. The ALLOY system then provides aid to the panel in the form of statistical and linguistic tools for the analysis of the collections and for the automatic indexing of new documents entered on the system. A standardized user interface is provided that allows a user to move among different subject areas without having to learn different interfaces. A component of ALLOY, called INDEXD, has already been implemented and an example its output was provided. The rest of ALLOY is currently under development and testing.

The authors wish to thank Donald Kraft and Bert Boyce for their helpful conversations regarding the content and exposition of this paper. This work was partially supported by grant LEQSF(86-87)-UNEXP-3 from the Board of Regents of the State of Louisiana. The authors also wish to thank the Louisiana State Law Institute for its support.

References:

[Croft86]
Croft, W. B., "User-specified domain knowledge for document retrieval", ACM Conference on Research and Development in Information Retrieval, Pisa, Italy. 1986.

[Croft87]
Croft, W. B. and David D. Lewis, "An Approach to natural language processing for document retrieval", Proceedings of the Tenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New Orleans, 1987.

[DiBenigno86]
DiBenigno, M. Kathryn, George Cross and Cary deBessonet, "COREL - a conceptual retrieval system", Proceedings of the ACM Conference on Research and Development in Information Retrieval, Pisa, Italy. 1986.

[Dillon83]
Dillon, Martin and Ann Gray, "FASIT: A fully automatic, syntactically based indexing system", Journal of the American Society for Information Science, Vol. 32, No. 2, pp. 99-108. 1983.

[Fagan87]
Fagan, Joel, "Automatic phrase indexing for document retrieval: an examination of syntactic and non-syntactic methods", Proceedings of the Tenth Annual International ACMSIGIR Conference on Research and Development in Information Retrieval, pp. 91-101. New Orleans. 1987.

[Jones88a]
Jones, Leslie, Edward Gassie and Sridhar Radhakrishnan, "INDEX: the statistical basis for an automatic conceptual phrase indexing system", to appear in the Journal of the American

Society for Information Science.

[Jones88b]
Jones, Leslie, Edward W. Gassie and Sridhar Radhakrishnan, "A portable repeated string-finder", submitted for publication.

[Lousiana]
Louisiana Civil Code, 1986 Edition, West Publishing Company. 1986.

[Salton87]
Salton, Gerald, "Expert Systems and Information Retrieval", ASCM SIGIR Forum, Vol. 21, No. 3-4. 1987.

[Salton83]
Salton, Gerald and Michael McGill, "Introduction to Modern Information Retrieval", McGraw-Hill, Inc. 1983.

[Smeaton86]
Smeaton, Alan, "Incorporating syntactic information into a document retrieval strategy: an investigation", Proceedings of the ACM Conference on Research and Development in Information Retrieval, Pisa, Italy. 1986.

[Tong87a]
Tong, Richard, Lee Appelbaum, Victor Askman and James Cunningham, "Conceptual information retrieval using RUBRIC", Proceedings of the Tenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New Orleans. 1987.

[Tong87b]
Tong, Richard, Clifford Reid, Gregory Crowe and Peter Douglas, "Conceptual legal document retrieval using the RUBRIC system", Proceedings of the First International Conference on Artificial Intelligence and Law, Boston, Massachusetts. 1987.

Illustration

9035874	sale
6411440	buyer
4419314	seller
3355344	time of the sale
2434536	right of redemption
1643264	thing
1379952	price
1113300	contract of sale
938800	delivery of the thing
930800	possession of the thing
929200	vices of the thing
912000	reduction of the price
653568	dissolution of the sale
563712	value of the land
511600	case of eviction
396000	case
371952	part of the seller
367575	right
361375	action
336816	payment of the price
335232	part of the thing
328176	restitution of the price
281394	sales
274500	sale of immovables
274500	sale of immovable
272250	liability of seller
259632	rise to the redhibition
239400	case of eviction
221400	seller is bound
219312	necessary for the general
209250	reduction of price
197325	vices of body
178848	ordinary contract of sale
176448	part of the buyer
175744	obligations of the buyer
174272	neglect of the buyer
169856	price of the sale
165669	purchaser
164096	object of the sale
163008	sale of the danger
163008	date of the sale
146688	preservation of the thing
145664	supplement of the price
145600	diminution of the price

Figure 1