

VU Research Portal

SIFT, a hybrid retrieval engine for providing help from technical computer manuals

Sutcliffe, R.F.E.; Hellwig, P.; Vossen, P.J.T.M.

1995

document version

Publisher's PDF, also known as Version of record

[Link to publication in VU Research Portal](#)

citation for published version (APA)

Sutcliffe, R. F. E., Hellwig, P., & Vossen, P. J. T. M. (1995). *SIFT, a hybrid retrieval engine for providing help from technical computer manuals*. Vrije Universiteit.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

E-mail address:

vuresearchportal.ub@vu.nl

SIFT, a Hybrid Retrieval Engine for Providing Help from Technical Computer Manuals.

Richard F. E. Sutcliffe*, Peter Hellwig%,
Piek Vossen&, Donie O'Sullivan*,
Liam Relihan*, and Annette McElligott*

University of Limerick*
University of Heidelberg%
University of Amsterdam&

Department of Computer Science*
and Information Systems
University of Limerick
Limerick, Ireland
+353 61 333644 Ext 5006
+353 61 330876 (FAX)
sutcliff@ul.ie

Abstract

How can linguistic meanings be represented in a form which is computationally tractable? We have developed a paradigm which allows both word and sentence meanings to be captured using distributed patterns, that is, vectors and matrices. These representations can be generated automatically. Word meanings are produced from machine readable dictionaries by a family of taxonomic traversal algorithms. Sentence meanings are then constructed from parse trees by using the distributed lexical representations together with semantic case information.

In the LRE-2 SIFT project we are developing a system for text retrieval from Technical Manuals using the above technology. The system takes as input a query relating to the content of the manuals. This is converted to a distributed pattern which is then "swept" over the patterns corresponding to units of the text looking for a good match. Salient portions of text are then shown to the user for perusal.

We have just completed the construction of SIFT-1, a version of the system in which the distributed representation of each textual unit is constructed using lexical but not grammatical information.

The work is interesting because: (1) Tractable meaning representations are generated automatically, and (2) Conventional symbolic NLP techniques are combined with a subsymbolic semantic representation scheme to form a robust information retrieval system.

1 Introduction

Computer manuals are, by common consent, very inconvenient to use. While they contain a wealth of useful information it is often virtually impossible to find the facts required. How can the usefulness of such manuals be improved?

One approach to this problem involves the construction of a specialised help system based on the information within the manual. Such a system can process queries in a natural language and provide answers which are directly related to those queries. Well known systems of this type are the Unix Consultant (Wilensky et al., 1988) and the SINIX Consultant (Hecking et al., 1988). While such programs are sophisticated they must be laboriously constructed for each and every application domain. In addition they tend to be fragile and are very difficult to update when changes need to be made to the information contained in them.

Another approach is to construct an Information Retrieval (IR) system from the documents comprising the manual. Such systems effectively use keyword matching techniques of varying sophistication and can be built automatically. One of the most successful of such approaches is the Vector Space Model (VSM) of Salton (1971). The main advantages in this case are automatic construction and robustness in performance. However, no real understanding of the text is involved, and so such systems can only have a limited performance.

In SIFT (Selecting Information From Text) we are investigating a different approach which aims to combine the benefits of the paradigms discussed above while avoiding their shortcomings. The essence of our idea is that a text can be represented as a set of distributed patterns. Each pattern can capture in gist the meaning of a textual unit such as a word, sentence or paragraph. At the same time the patterns allow the VSM to be used as the retrieval mechanism.

The efficacy of this approach depends on being able to generate reasonably accurate representations automatically. This is achieved by combining two techniques. To generate patterns for individual words we have developed an algorithm which can traverse a taxonomic hierarchy derived from an electronic dictionary. To create a pattern for a sentence or verb phrase we combine these patterns using a set of semantic cases. A robust bottom up parser based on syntagmatic relations is used to analyse phrases within each input text in order to accomplish this. Information both for constructing the distributed representations and for building the parser is derived from a database of lexical information derived from several sources including the Longmans Dictionary of Contemporary English and the Princeton WordNet.

SIFT is based on a previous prototype, PELICAN (Sutcliffe, 1991). PELICAN was intended to retrieve appropriate help files in response to a query about Unix. The figure below shows output from the system. The system demonstrated the efficacy of using distributed representations for retrieval but the parser was very simple and the semantic lexicon was created by hand. SIFT addresses these shortcomings.

The rest of the paper is structured as follows. Firstly, we describe the general architecture of the SIFT systems and describe how they operate. We then turn to the role of distributed representations in the project, defining what these are and how they can be created automatically for words, phrases and sentences in the text. Finally we outline the operation of SIFT-1 and summarise progress made on the project so far.

I would like to remove a directory.

1	rmdir	(10)	0.990290
2	rm	(9)	0.887207
3	rsh	(8)	0.737982
4	pwd	(7)	0.716869
5	ls	(1)	0.686783
6	mv	(4)	0.653809
7	mail	(6)	0.650262
8	more	(3)	0.629431
9	cd	(2)	0.621570
10	rmail	(5)	0.567892

I wish to change directory.

1	cd	(2)	0.894543
2	ls	(1)	0.828476
3	mv	(4)	0.718574
4	rmdir	(10)	0.717317
5	more	(3)	0.702810
6	pwd	(7)	0.692261
7	rsh	(8)	0.689359
8	rmail	(5)	0.654914
9	mail	(6)	0.617851
10	rm	(9)	0.614234

Example queries processed by PELICAN

2 Operation of the SIFT System

The SIFT system consists of two principal components. The first, the document processing component takes as input an SGML tagged computer manual and associates with the different sections, subsections and individual sentences of that manual distributed patterns capturing the meaning of those textual units. The second, the query processing component takes as input a user query about the material covered in the document and produces as output a list of pointers to text portions within it which are ordered by relevance to the query.

The document processing component operates by parsing the text of the manual utterance by utterance and then converting each of these into a distributed pattern which captures its meaning. Additional distributed patterns are also associated with headings which can be analysed in the same way as the text itself. Finally, these semantic patterns are linked back to portions of the original text from which they are derived.

Query processing is accomplished by parsing the input query and hence converting it into a distributed pattern, and then performing a match of this with each of the patterns

linked to the text during the document processing stage. The comparison is readily done because the patterns are vectors, and the result is a list of portions of the original text, ordered by their relevance to the query.

In summary, therefore, the system operates as follows. You start with a query, for example concerning installation of the Lotus Wordprocessor AmiPro. The query must be formulated as an English statement such as ‘installing AmiPro’ or a question such as ‘how do I install AmiPro’ and then typed into SIFT. The system then searches for portions of the AmiPro reference manual which deal with installation and produces a list of these ordered by their relevance to the query. Finally, the user can browse through the sections indicated and in so doing find the answer to their question.

3 Distributed Representations

3.1 What is a distributed representation

A distributed representation is one in which the information to be captured is distributed over a large set of units. For example, the meaning of a word can be captured as a set of <feature,centrality> pairs. Within this paradigm the meaning of the word AmiPro could be defined as [<wordprocessing,0.8>, <computer,0.7>, <file,0.3>]. In such a representation, no feature provides complete information about the concept defined. Taken together, however, the set of features provides an outline definition of the concept. In general, such a representation is equivalent to a vector in an n-dimensional space, since if each feature in a system is listed exhaustively in every representation, the features can be omitted entirely. Vectors can be processed in many ways which are not applicable to structured information such as lists. For example, two vectors can be compared very readily by the use of, say, the dot product. The use of distributed patterns thus allows the comparison of a pair of meanings to be reduced to a simple arithmetic operation.

3.2 Constructing Distributed patterns for individual words

We need to construct for each noun or verb sense in a lexicon a semantic representation consisting of a set of feature-centrality pairs. The features are semantic attributes each of which says something about the concept being defined. The centrality associated with each feature is a real number which indicates how strongly the feature contributes to the meaning of the concept. The use of centralities allows us to distinguish between important and less important features in a semantic representation. By scaling the centralities in a particular noun-sense representation so that the sum of their squares is one we can use the dot product operation to compute the semantic similarity of a pair of concepts. A word compared to itself always scores one while a word compared to another word is always less than or equal to one. This is equivalent to saying that each word representation is a vector of length one in an n-dimensional space, where n is the number of features which are used in the lexicon as a whole.

Our algorithm for constructing the representations is based on two well-known observations. Firstly, a word definition in a dictionary provides attribute information about the concept (‘a mastiff is a LARGE dog’). Secondly a word definition also provides taxonomic information about the concept (‘a mastiff is a large DOG’). We use the former

to derive attributes for our representation, and the latter to obtain other definitions higher up in the taxonomy from which further attributes can be obtained. In assigning centralities to features, we use the same value for each attribute added at a particular level in the taxonomic hierarchy, and we reduce the value used as we move up to higher levels. This corresponds to the intuition that a feature which is derived from a definition which is close to the word of interest in the taxonomy contributes more to its meaning than one which is derived from a more distant definition.

So far we have implemented and tested the basic extraction algorithm on nouns derived from the Merriam Webster Compact Electronic Dictionary (Sutcliffe, 1992) and the Irish-Irish An Foclóir Beag. A MKI traversal on nouns yielded good results from the Princeton WordNet (Sutcliffe, O'Sullivan and Meharg, 1994) while the MKII traversal which incorporates feature unification works for both nouns and verbs (Sutcliffe, O'Sullivan, Slater and Brehony, 1994). The performance of this lexicon has recently been tested against human subject data with good results (Sutcliffe, Vossen, Sharkey, O'Sullivan, Slator and McElligott, 1994).

A MKIII traversal of WordNet is under development. This incorporates a more sophisticated process of feature unification which allows us to control the feature set size as well as being able to make representations either more or less distributed (Sutcliffe, 1991a).

4 Progress

So far, the SIFT-1 system has been completed and is currently undergoing refinement and testing. SIFT-1 uses the lexicon of distributed representations derived from WordNet but incorporates no syntactic processing. Instead, an utterance representation comprises a list of <word representation, syntactic category> pairs. Syntactic categories are determined using the Brill tagger (Brill, 1992). We have developed an algorithm for matching a pair of such utterance representations which is based around the best match of a particular <word, category> pair in utterance one with a word of the same syntactic category in utterance two.

In order to use a manual with SIFT-1 it must be converted into an appropriate format. We have developed a filter which converts a native AmiPro binary file into SGML using the SIFTREP DTD designed for this purpose. In the second stage, the SGML text is scanned, tagged and disambiguated. A stream of utterance representations can then be produced which are stored in the SIFT-1 RDAFT database. Each utterance has a unique World Wide Web (WWW) Uniform Resource Locator (URL) associated with it so that the WWW mechanism for identifying document portions can be exploited in SIFT.

SIFT-1 is used via a customised mosaic interface with SIFT-1 itself acting as a WWW server. Once a query is entered id passed to the server for processing. The query is tagged, disambiguated and converted to a list of <word, category> pairs as mentioned above. This is then matched with the utterance representations of portions of the manual text, leading to the production of an ordered list of matches which is returned to the interface for display. The user then selects a particular section of the manual from the list. This is converted into a request for the document with the corresponding URL to

be shown. SIFTREP is designed so that it can be converted on-the-fly into HTML, the DTD used for WWW documents. Finally, the document appears on the screen and can be studied by the user.

In summary therefore, SIFT-1 is a full text retrieval system which operates by matching the semantic representation of an input query with the representations of text utterances in the manual. While SIFT uses ideas derived from connectionist research it is not a true connectionist system. Likewise it is not restricted to conventional language engineering technology either. We believe that hybrid approaches of this kind are a very promising way of addressing the intractable problems which the task of building robust and useful NLP systems imposes.

5 References

Brill, E. (1992). A simple rule-based part of speech tagger. Proceedings of the Third Conference on Applied Natural Language Processing, ACL, Trento, Italy, 1992.

Hecking, M., Kemke, C., Nessen, E., Dengler, D., Gutmann, G., & Hector, G. (1988). The SINIX Consultant - A Progress Report (Bericht Nr. 28). Sarbruecken, Germany: Universitaet des Saarlandes, FB 10 Informatik IV, Lm Stadtwald 15, D-6600, Sarbruecken 11.

Hellwig, P. (1980). PLAIN - A Program System for Dependency Analysis and for Simulating Natural Language Inference. In L. Bolc (Ed.) Representation and Processing of Natural Language (271-376). Munich, Germany, Vienna, Austria, London, UK: Hanser & Macmillan.

Hinton, G. E. (1981). Implementing semantic networks in parallel hardware. In G. E. Hinton and J. A. Anderson (Eds) Parallel models of associative memory (pp. 161-187). Hillsdale NJ: Lawrence Erlbaum Associates.

Hinton, G. E., McClelland, J. L. and Rumelhart, D. E. (1986). Distributed Representations. In D. E. Rumelhart and J. L. McClelland (Eds) Parallel Distributed Processing: Explorations in the microstructure of cognition. Volume I: foundations (pp. 77-109). Cambridge MA: MIT Press.

Salton, G. (Ed.) (1971). The SMART Retrieval System - Experiments in Automatic Document Processing. Englewood Cliffs, NJ: Prentice Hall.

Smolensky, P. (1987a). A method for connectionist variable binding (Tech. Rep. CU-CS-356-87). Boulder, CO: University of Colorado, Department of Computer Science, February 1987.

Smolensky, P. (1987b). On variable binding and the representation of symbolic structures in connectionist systems (Tech. Rep. CU-CS-355-87). Boulder, CO: University of Colorado, Department of Computer Science.

Smolensky, P. (1990). Tensor Product Variable Binding and the Representation of Symbolic Structures in Connectionist Systems. Artificial Intelligence, 46(1-2), 159-216.

Sutcliffe, R. F. E. (1991a). Distributed Subsymbolic Representations for Natural Language: How many Features Do You Need? In M. F. McTear and N. Creaney (Eds.) Proceedings of the 3rd Irish Conference on Artificial Intelligence and Cognitive Science, 20-21 September 1990, University of Ulster at Jordanstown, Northern Ireland. (pp. 279-305). Berlin, FRG, Heidelberg, FRG, New York, NY: Springer-Verlag.

Sutcliffe, R.F.E. (1991b). Distributed Representations in a Text Based Information Retrieval System: A New Way of Using the Vector Space Model. In A. Bookstein, Y. Chiaramella, G. Salton & V. V. Raghavan (Eds.) The Fourteenth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, Chicago, Il., October 13-16, 1991 (pp. 123-132). New York, NY: ACM Press.

Sutcliffe, R.F.E. (1992). Constructing Distributed Semantic Lexical Representations using a Machine Readable Dictionary. In Proceedings of AICS-92 - The Fifth Irish Conference on Artificial Intelligence and Cognitive Science, University of Limerick, 10-11 September 1992. London, UK, Berlin, FRG, Heidelberg, FGR, New York, NY: Springer-Verlag.

Sutcliffe, R. F. E., O'Sullivan, D., & Meharg, F. (1994). A Lexicon of Distributed Noun Representations Constructed by Taxonomic Traversal. Proceedings of the 15th International Conference on Computational Linguistics, (COLING'94), Kyoto, Japan.

Sutcliffe, R. F. E., O'Sullivan, D., Slater, B. E. A., & Brehony, T. (1994). Traversing WordNet to Create Optimised Semantic Lexical Representations. In Proceedings of the Seventh Annual Irish Conference on Artificial Intelligence and Cognitive Science (AICS'94), Trinity College Dublin, 8-9 September, 1994.

Sutcliffe, R. F. E., Vossen, P., Sharkey, N. E., O'Sullivan, D., Slator, B. E. A., & McElligott, A. (1994). A Psychometric Performance Metric for Semantic Lexicons. Proceedings of International Workshop On Directions Of Lexical Research, 15-17th of August, 1994, Beijing, China.

Vossen P. (1990). A Parser-grammar for the Meaning Descriptions of the Longman Dictionary of Contemporary English. (Tech. Rep. NWO, project no. 300-169-007). Amsterdam, Netherlands: University of Amsterdam.

Vossen, P. (1991). An empirical approach to automatically construct a knowledge base from dictionaries. Paper to be presented at the Euralex conference, Tampere, 1992. Also available as ACQUILEX Working Paper no. 25, Esprit BRA-3030, Amsterdam.

Wilensky, R., Chin, D. N., Luria, M., Martin, J., Mayfield, J., & Wu, D. (1988). The Berkeley UNIX Consultant Project. Computational Linguistics, 14(4), 35-83.