

# Generating Better Concept Hierarchies Using Automatic Document Classification

Razvan Stefan Bot, Yi-fang Brook Wu, Xin Chen, Quanzhi Li  
Information Systems Department  
New Jersey Institute of Technology  
{rsb2, wu, xc7, ql23}@njit.edu

## ABSTRACT

This paper presents a hybrid concept hierarchy development technique for web returned documents retrieved by a meta-search engine. The aim of the technique is to separate the initial retrieved documents into topical oriented categories, prior to the actual concept hierarchy generation. The topical categories correspond to different semantic aspects of the query. This is done using a 1-of-n automatic document classification, on the initial set of returned documents. Then, an individual topical concept hierarchy is automatically generated inside each of the resulted categories. Both steps are executed on the fly at retrieval time. Due to the efficiency constraints imposed by the web retrieval context, the algorithm only uses document snippets (rather than full web pages) for both document classification and concept hierarchy generation. Experimental results show that the algorithm is able to improve the quality of the concept hierarchy presented to the searcher; at the same time, the efficiency parameters are kept within reasonable intervals.

## Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval – *clustering, information filtering*; I.2.7 [Artificial Intelligence]: Natural Language Processing – *text analysis*.

**General Terms:** Algorithms.

**Keywords:** Information retrieval, document classification, automatic classification, manual classification, concept hierarchy.

## 1. SYSTEM DESIGN

The goal of the prototype system presented throughout this section is to improve the quality of the concept hierarchies generated for web returned documents. The approach is to augment automatic concept hierarchy generation by adding a pre-classification step. During this step, web returned documents are assigned to a set of 13 predefined and 1 “others” categories dynamically; meaning that, based on the content, returned documents for a query could be separated into as few as 1 or as many as 14 categories. The classification task is of type 1-of-n, meaning that each web document can only be assigned to one single category. After that, an individual concept hierarchy is generated for each of the active categories. The rationale for adding the pre-classification step is to obtain a set of topical oriented categories (more homogeneous categories). Our hypothesis is that the quality of automatically generated concept

hierarchies is better for homogeneous document sets than those for heterogeneous document sets. Therefore, the steps of the algorithm are: **Step 1:** retrieve web documents in response to a query from several sources; **Step 2:** pre-classify these documents into a set of manual predefined categories, using classifiers built from an internet positive sample of the Google Directory; **Step 3:** extract noun phrases from classified documents for each active category, and automatically generate an individual concept hierarchy within each of the active categories.

### 1.1 System Architecture

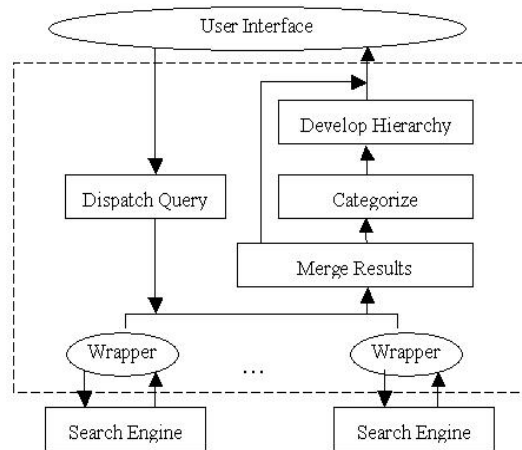


Figure 1. System Architecture

Figure 1 illustrates the system architecture. Step 1 consists of the query dispatch module, the wrapper and the results merge modules. Step 2, the pre-classification module, is depicted as “Categorize.” Step 3, the automatic concept hierarchy generation is depicted as “Develop Hierarchy.” One can notice the streamlining of pre-classification and concept hierarchy generation.

### 1.2 Document Acquisition

Step 1 of the algorithm is document acquisition. Our system, named Highlight, has a meta-search engine architecture. Documents are collected from several search engines: Google, Teoma, MSN Search, AltaVista and AllTheWeb. The “Dispatch Query” module (See Figure 1) is responsible for dispatching the query to all the search engines listed above. All retrieved document snippets are merged together, to form a unified document list. The merging operation consists of duplicates removal and document ranking. To rank document snippets, a relevancy indicator is computed. The relevancy indicator for a

document snippet is defined as: the average of the rank orders of the document snippet, in all sources.

### 1.3 Returned Documents Pre-Classification

Step 2 of the algorithm is the document pre-classification. In Figure 1, this process is represented by the “Categorize” module. The input of the process is the document snippets list collected during document acquisition. We are using NB classifiers that perform a 1-to-n classification task. In order to build our classifiers, we first collected a sample of the Internet. Sample documents were collected from the Google Directory (<http://www.google.com/dirhp>). The top level of Google Directory’s hierarchy consists of 16 generic categories. They are: Arts, Business, Computers, Games, Health, Home, Kids and Teens, News, Recreation, Reference, Regional, Science, Shopping, Society, Sports and World. Three categories: Reference, Regional and World, were removed from consideration due to the fact that they are very broad and non-cohesive topical categories. We then sampled each of the remaining categories to collect our representative document collection. Our sampling process collected 25% of the directory’s content, from 13 categories. To evaluate the classification accuracy of these classifiers, a 5-fold cross-validation was conducted (see Table 1).

Table 1. Classification Accuracy

Training with →		snippets	full text
Fold	# of doc	NB	NB
1	25587	0.62	0.67
2	24996	0.57	0.66
3	25178	0.62	0.69
4	24855	0.63	0.68
5	25301	0.62	0.67
Mean		0.61	0.67

### 1.4 Concept Hierarchy Generation

Step 3 of the algorithm is the automatic concept hierarchy generation. In Figure 1, this process is represented by the “Develop Hierarchy” module. After document snippets are loaded into the system, the tokenizer module separates all the words, punctuation marks and other symbols from document text to obtain the atom units. The part-of-speech (POS) tagger module is then used to assign proper part-of-speech labels to all tokens. Our POS tagger is a revised version of the widely used Brill tagger [1]. After all the words in the document are tagged, the noun phrase extractor (NPE) extracts noun phrases by selecting the sequence of POS tags that are of interests. The current sequence pattern is defined as  $\{[A]\} \{N\}$ , where A refers to Adjective, N refers to Noun,  $\{ \}$  means repetition, and  $[ ]$  means optional. The last step is the concept hierarchy construction. We revised Sanderson and Croft’s algorithm [2] previously described into a new version called probability of co-occurrence analysis (POCA) [3]. The reason behind this is that Sanderson and Croft’s method could include term pairs like  $P(X|Y)=0.8$  and  $P(Y|X)=0.9$ , which X does not subsume Y but the pair still fulfills the selection rule. This algorithm is re-defined as follows to prevent the above situation:  $P(X|Y) > P(Y|X)$ ,  $P(X|Y) \geq N$ , where  $0 < N \leq 1$ . If a term pair (X, Y) fulfills the above set of inequalities, X is the parent of Y. In our study, we used  $N=0.8$  which is the same as in [2].

## 2. RESULTS

### 2.1 Concept Hierarchy Precision Comparison

This section presents the results of the comparison between the average concept hierarchy precisions of the baseline and augmented systems. Nine Ph.D. students knowledgeable in classification/clustering were recruited to identify relationship types in the concept hierarchies. There were eight relationship types: (1) “is an aspect of”; (2) “is a part of”; (3) “is a type of”; (4) “is a child of”; (5) “is the same as”; (6) “is the opposite of”; (7) “has no relation with”; and (8) “can’t tell.” The precision of a concept hierarchy is defined as the proportion of hierarchical relationships of type 1, 2, 3, 4, and 5 out of all hierarchical relationships of the concept hierarchy.

Table 2. Hierarchy precision: baseline vs. augmented

	Baseline	Augmented
$P_{\text{concept\_hierarchy}}$	0.495	0.589 (+18.9%)

The results show that the pre-classification step generates better/cleaner input document sets for the concept hierarchy generation phase. As a result, the concept hierarchy precision substantially improves (about 18.9%).

### 2.2 Time Efficiency

It is very important to deliver a concept hierarchy as fast as possible to the user in order to avoid usability pitfalls. We obtained the following results: average searching time is 2500ms and classification and hierarchy generation average time is 300ms. Even if the total response time is larger than that of a regular search engine, the effectiveness is much higher, because the system is processing 200 documents rather than 10 at a time.

## 3. CONCLUSIONS

The results show that adding a pre-classification phase to the automatic concept hierarchy generation for a set of web-returned documents improves the precision of the resultant concept hierarchies.

## 4. Acknowledgement

This project is, in part, supported under NSF grants DUE-#0434581 and DUE-#0434998.

## 5. REFERENCES

- [1] Brill, E. (1995) “Transformation-based Error-driven Learning and Natural Language Processing: A Case study in Part-of-speech Tagging.” *Computational Linguistics* 21(4), pp. 543-565.
- [2] Sanderson, M. and B. Croft (1999). "Deriving concept hierarchies from text." *Proceedings of the 22<sup>nd</sup> annual international ACM SIGIR Conference on Research and Development in Information Retrieval*. Berkely, CA, pp. 206-213.
- [3] Wu, Y. B., C. Rakthin and C. Li (2002). "Summarizing Search Results with Automatic Table of Contents." *AMCIS 2002*, Dallas, TX: pp 88-92.