# Autonomous Filter Engine Based on Knowledge Acquisition from the Web

Luigi Lancieri, Pierre Agostini, Nicolas Saillard, Samuel Legouix
*Centre National d'Etudes des Télécommunications France Telecom*
*{luigi.lancieri, pierre.agostini, nicolas.saillard, samuel.legouix}@cnet.francetelecom.fr*

## Abstract

*For a long time filtering data has been a very important matter. With the growth of Internet this question has become more and more pressing. Everyone knows that finding information is not always very easy, because of the large amount of data that the network contains. More generally, our concern here is not only to find what is interesting to download but also to avoid what is unsuitable. One of the different ways recently explored to address the question of filtering and rating information is to apply the studies done in the field of linguistic and artificial intelligence to the web. In this context, we would like to show that it is possible to build an easy going and low cost tool to compare information. This tool will use automatically extracted and selected Web contents to build a specialized knowledge database that can easily be adapted to any subject in any language.*

## 1 Methodology

We first describe how the semantic database will be used to compare a page, randomly downloaded from the Web by one user, to a specific theme. We give here a brief overview of the model used; the complete version [1] is available for more details. The basic principle in analyzing and characterizing page content is to correlate the words spatial proximity to their semantic proximity This means that words that are close (in the text) are statistically supposed to have a close meaning. Progressively analyzing a high quantity of page helps to build a semantic network and the representative matrix that formalizes the inter-word connectivity. This matrix can be considered as a representation of *n* (total words) vectors in an *n* dimensional vectorial space. As we will see, the coefficients of one vector are defined automatically by a learning process. In these conditions, an estimated value of the semantic link between two words can be formulated as the Euclidean distance between the two vectors associated with these words. A semantic representation of one page can be represented by its barycentre. This can be obtained by computing the resulting vector from the words' vectors of the page pondered by their occurrences in the page. One interesting point regarding this model is that it is possible to characterize and compare all elements of information (words, user's profile[1], Web pages,...) in the same algebraic space. This method has some similarities with Salton or Dumais' works [2].

We need now to compare a human ranking ability to the automatic ranking system. In order to do that, we "manually" download from the Web a set of 60 HTML pages that more or less matches a specific subject (e.g. " car"). Each page has a comparable size, and only one contains one time the word "car" in a set of about 60 000 words. This set of pages was ranked according the human feeling level of relevancy with the subject "car". Each person from a group of six was asked to rate the set of pages without knowing the results of the others (this took about 3 hours per tester). We also computed the system evaluation of the 60 pages by comparing the vector barycentre of each page to test with the thematic vector "car" extracted from the knowledge database. Remember that this vector is made up of the weights of all words linked to the word "car".

In the global architecture the Filter engine is associated with a proxy cache and a semantic engine (knowledge database and access interface) in a local area network. Users get connected to the Internet through the cache and almost all the pages that the users download go in the cache. In our case, the filter engine will act in the L.A.N as a virtual user. In order to collect pages, we use an access module made up of a regular search engine service (e.g. altavista) connected with a module that allows pages whose addresses are sent back by the search engine to be downloaded. The access module is at first fed by a user who gave 3 or 4 key words (bootstrap) basically related to the theme that will be used to build the knowledge database. The access module automatically feeds the proxy cache with thousands of pages that are more or less consistent with the chosen theme. The decision module will sort these pages and feed the semantic engine with the most pertinent ones.

We know, by experience, that the first pages returned by a search engine have a good level of consistency. So, even if we can not compare these pages with the semantic database (it is empty at the beginning), we can accept them and start to fill the database. Progressively, key words will appear in the database and will be used to compare the next pages. A soon as the needed key word (e.g "car") appears, we extract its representative vector.

We also select the less no null weighted word in this vector and we extract its corespondent vector from the database. We compute the "reference" distance between these 2 vectors that correspond to the most related and most non-related theme (with "car"). To make the decision to accept a new page, we compute the distance between this page and the vector "car" and if this distance is up to half the "reference" we accept the page. As the database grows, we compute a new reference that becomes more and more reliable.

## 2 Results

It is well known that human perception and feeling can be very different from one person to another. Each user has a personal feeling on assessment matter and consequently the difference in sensitivity concerning peripheral elements such as images or even the personal knowledge of the tested field may modify the judgment. The following table gives a view of user subjectivity regarding the ranking of the set of documents. We compute the correlation between the users' evaluation and between the users and the system evaluation. The first 6 items identify the human testers. The item "syst" identifies the filter system. The items "Aver" and "best" are respectively the average and the best combination of all the 6 human users. This table shows that the heterogeneity between the users is on the same level as between users and the filtering system. For example the correlation between user SL and PA is equal to 0,44 whereas the correlation between the system and the users vary from 0,35 to 0,55.

| LL | PA | NS | FM | SL | DV | Syst | Aver | Best | |
|---|---|---|---|---|---|---|---|---|---|
| 1,00 | 0,75 | 0,76 | 0,74 | 0,59 | 0,79 | 0,43 | 0,90 | 0,81 | LL |
| | 1,00 | 0,75 | 0,72 | 0,44 | 0,70 | 0,46 | 0,87 | 0,80 | PA |
| | | 1,00 | 0,72 | 0,53 | 0,75 | 0,55 | 0,91 | 0,95 | NS |
| | | | 1,00 | 0,49 | 0,71 | 0,53 | 0,86 | 0,90 | FM |
| | | | | 1,00 | 0,44 | 0,35 | 0,64 | 0,55 | SL |
| | | | | | 1,00 | 0,37 | 0,87 | 0,79 | DV |
| | | | | | | 1,00 | 0,53 | 0,58 | Syst |
| | | | | | | | 1,00 | 0,96 | Aver |
| | | | | | | | | 1,00 | Best |

Table 1.: Coef.. of correlation between users and system

The low part of the following graphs represent the evaluation of two testers that were chosen according to the representative divergence of their ranking. (FM, vs NS coef 0,72), the upper graph shows the human (best combination) compared to the system evaluation. (coef 0,58). We see that the global trend of the system estimation is similar to the human one. Of course, all values do not match exactly but it is not possible to have errors between relevant and non-relevant pages (extreme parts of the graph). Furthermore, let's remember that the word chosen as representative of the central theme (car) almost never appears in the tested pages.

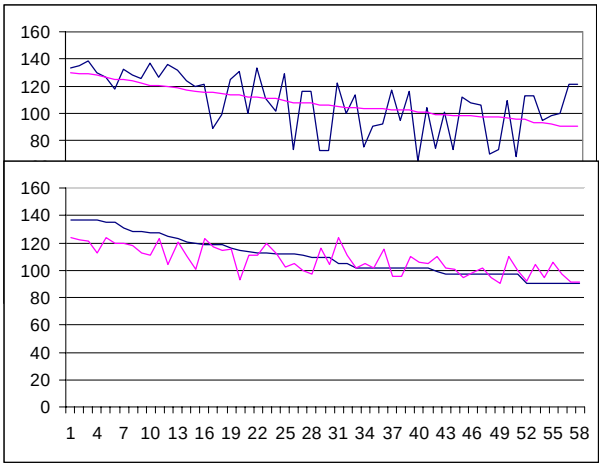


Figure1: Human vs automatic ranking

Like many systems based on artificial intelligence, the performances of this system highly depend on the learning level. For our system, learning means downloading HTML pages that more or less contain information on cars. This step was automatically done with the search engine. Feeding the system with 100, 200 to 500 pages allow progressively the best correlation coefficient (0.3 to 0,58).

Performance is also a very important question. In the experimental version, the overall computation delay (analyze of pages, distance measure and take of decision) is less than 2 seconds for each textual document on a Pentium 200 Bi processor. As a Web page is downloaded in two step, the text first and then the images, this computation can be done in the same time as image downloading that allows this system to be used for online filtering with limited extra latency impact for the users. The database needs 15 h to be completed but that can be done over working time.

## 3 References

[1] Distributed Multimedia Document Modeling
Luigi Lancieri ; In proceedings of IEEE Joint Neural Network Conference 1998

[2] Using linear algebra for intelligent information retrieval; M.W. Berry, S.T. Dumais.