A Cooperative Paradigm for Fighting Information Overload

Daniel Gayo-Avello, Darío Álvarez-Gutiérrez, and José Gayo-Avello

Department of Informatics, University of Oviedo, Calvo Sotelo s/n 33007 Oviedo (SPAIN)
{dani, darioa}@lsi.uniovi.es

Abstract. The Web is mainly processed by humans. The role of the machines is just to transmit and display the contents of the documents, barely being able to do something else. Nowadays there are lots of initiatives trying to change this situation; many of them are related to fields like the Semantic Web [1] or Web Intelligence. In this paper we describe the Cooperative Web [2] that can be seen as a new proposal towards Web Intelligence. The Cooperative Web would allow us to extract semantics from the Web in an automatic way, without the need of ontological artifacts, with language independence and, besides of this, allowing the usage of browsing experience from individual users to serve the whole community of users.

1 Introduction

Although the Web provides access to a huge amount of information it is not a perfect information retrieval mechanism. Search engines perform a really useful task but we can say that they are toping out since they provide a view of the Web quite poor to get a more powerful use. This claim can seem exaggerated but if we take into account two latest initiatives from the main search engine –Google Answers[1] and the First Annual Google Programming Contest[2]– it is clear that, implicitly, Google admits that state of the art techniques have reached their limit and "something more" is needed.

However, and for the moment, users continue using search engines that provide only a lexical view of the Web and force them to browse hundreds of documents for an increasing time until they find the piece of information they are looking for.

Besides of this, the current Web shows a problem as serious as its lack of semantics: each time a user browses the Web, he opens a path which could be useful for others and, in the same way, other users can have yet followed such path and have found its worth or its uselessness. However, all that experimental knowledge is lost.

Such situation is flawed and something is required to provide intelligence and semantics to the Web. We think that this is possible in an automatic way, transparent to the user, and language independent by using software agents and computational biology algorithms. Through this paper we will show in which way we think this task could be accomplished.

[1] http://answers.google.com/answers/faq.html#whatis
[2] http://www.google.com/programming-contest/

A. Yazici and C. Şener (Eds.): ISCIS 2003, LNCS 2869, pp. 228–235, 2003.

2 The Web as an Information Retrieval System

2.1 Traditional Information Retrieval Systems

The pair Web + search engines can be considered as a text retrieval system. So it is suitable for being assessed in terms of two measures, recall and precision. Both measures should ideally tend to 100% but in real systems the task of balancing them falls on the user at the moment of making his queries with more or less detail.

Likely, the main problem remains in the use of keyword-based queries. It is well known that the probability of two users employing the same keyword to refer a unique concept is below 20% [3]. So, if the query keywords are only looked for in the HTML META tags or in the document title the results are quite poor and if the search is performed using free text from the documents the recall is larger but at the expense of a serious lack of precision [4].

Such lack of precision lies, mainly, in the ambiguity of the words, even in well defined domains [5]. It is the meaning given to a word, more than the word itself, what allows us to distinguish relevant from irrelevant documents and what would permit to resolve queries that should return documents which do not include all the keywords but synonymous or semantically close words.

2.2 First Search Engines

The first main goal of the Web was to avoid the loss of information intrinsic to a large organization as well as making easier the access to available information. The initial proposal [6] suggested to develop the Web starting from a semantic ground by using concept nodes, which would be linked from the documents, and shows the drawbacks of using keyword-based information retrieval systems.

However, the Web was finally developed in a much simpler way, very similar to traditional hypertext, thus, making it an artifact designed to grow very quickly but with no document retrieval mechanisms.

In 1994 the first search engines appear: ALIWEB [7], WebCrawler [4] and Lycos [8]. These search engines showed clearly that links directories were not enough to find information but they raised new problems. On one hand, each search engine used a different Web exploring technique so the created databases were also different forcing the users to try their queries with several search engines. On the other hand, most of the returned documents had little relevance.

2.3 Modern Search Engines

Kleinberg [9] stated that documents' relevance requires human assessment. But, could this understanding of relevance be algorithmically computed? To perform this task, Kleinberg defined the concepts of "authority" and "hub".

An "authority" is a heavily linked document. If each link to a document is considered a vote for that document then such a link would indicate the relevance of the document from a human point of view since it was established by a person. By analyzing the text employed in the links to the target document and in the document itself it can be set for which topics the document is an authority. On the other hand, a hub is a document that links to many authorities.

Later, Page et al introduce the PageRank algorithm [10], based on the Kleinberg algorithm but with some new ideas to assess the relevance of documents and which turned out to be the kernel of the Google search engine. With this algorithm each document receives a score that shows the document's relevance. To compute this score the different links have different weights and the PageRank flows from one document to documents linked from this one. Thus, very linked documents would get top PageRanks and, this is the novelty, not very linked documents but from authoritative ones would inherit top PageRanks. Doing this, the search engine gets higher recall although only knowing something about their contents and the user interests could reject many of the return documents for a query.

2.4 Fighting against Information Overload

Thus, more than lack of precision the problem of the Web is information overload. A problem almost as old as the Internet since the users have suffered it with e-mail and especially with USENET posts.

For the last decade there have been many proposals to alleviate such a situation –[11], [12], [13], [14], or [15] to only quote some of them– Several ones were proposed specifically for some of the previously mentioned services while others wanted to filter any kind of information from the Internet. The technologies employed in this task were mainly three, combined or on its own: software agents, collaborative filtering and content-based recommendation.

Such initiatives did not have great success; this is not very shocking since most of them forced the user to evaluate the obtained results in order to improve the system performance. In fact, the users rarely are willing to do this extra work [14] since they consider it bothersome, so, it is imperative to use only implicit feedback [15]. On the other hand, the approaches that took into account the contents only processed automatically extracted keywords something that provides poor solutions.

3 The Semantic Web

Part of the problem can be attributed to the continuous attempts of adapting techniques to perform text retrieval in local systems to an ever-growing worldwide system. Such attempts do not perform in the expected way and, although the exploitation of hyperlinks has allowed better search engines, the increasing number of documents hinders the precision of the results and keeps the users under a flood of information.

In 1998 Tim Berners-Lee started to outline the Semantic Web. The main idea is to mark up the documents on the Web with "semantic tags" that would provide metainformation about the tagged text.

In a sense, this idea is quite similar to the use of "concept nodes" described in the original Web proposal [6] and now again the crux of the matter is the way to provide such semantic tags and state the relationships between them. To perform this task ontologies and ontological languages are being used.

Other approaches were proposed before the Semantic Web itself and have contributed greatly to it. Some of the most relevant have been: SHOE and Exposé [16], WebKB [17] and Ontobroker [18]. All of them showed similar characteristics to the ones that the Semantic Web, according to Berners-Lee, would need: a system to state asserts (RDF), a model to define new properties and relationships (RDF Schema) and a logic layer (inference and queries).

Later, a more elaborated version [1] of the Semantic Web was introduced; in this one ontologies take a leading role similar to the one played in above proposals.

3.1 Information Retrieval and Limitations of the Semantic Web

The Semantic Web is not widespread enough as to provide search engines comparable to those from traditional Web. However, some solutions have been proposed, such as Metalog [19], SquishQL [20] or SiLRI [21] that allow queries on RDF data and RQL/Sesame [22] that allows queries on RDF and RDF(S) data by means of a functional language.

Projects such as DAML, SHOE or OIL rely on some kind of query language that, in turn, relies on one or more ontologies. Because of that, in spite of differences on syntax or architecture, the Semantic Web "search engines" can be seen as a kind of inference engines which accept queries expressed in terms of one or more ontologies and return as results objects belonging to such ontologies.

Thus, the Semantic Web depends heavily on ontologies, because of that many efforts are being made to provide semi-automatic generation of ontologies [23] and automatic semantic markup of documents [24].

This dependence is also the cause of the two main limitations of the Semantic Web. On one hand the creation of huge ontologies with large numbers of classes and relationships between them will require, at least in the short time, human supervision [23]. On the other hand, the ontologies developed up to now and the queries for which the Semantic Web best performs are more metasemantics than semantics (e.g. it is possible to build an ontology to allow finding papers from a specific author but quite difficult to model an ontology to fit all the topics which could appear in the papers).

In short, the Semantic Web will make the access to information much easier in well-defined environments such as corporate intranets [25] but it would be really difficult to apply the same techniques to the Web as a whole.

4 The Cooperative Web

As we can see, approaches to fight against information overload are not suitable to help the user in his information searches on the Web. Some of them require explicit feedback from the user to evaluate the retrieved information. On the other hand, most of the proposals cannot process documents with language independence. As for the Semantic Web, it will play a vital role in well-defined domains but it is difficult to apply it to the whole Web in an automatic way.

Thus, we propose a possible, and complementary, solution in order to contribute towards the Web Intelligence, the so-called Cooperative Web:

"The Cooperative Web is a layer on top of the current Web to give it semantics in an automatic, global, transparent and language independent way. It does not require explicit user participation but implicit feedback that would be acquired by software agents. The Cooperative Web relies on the use of concepts and document taxonomies, both of them can be obtained with no human supervision from free text."

Many researchers involved in the Web Intelligence field share similar views and proposals. Nishida introduces the concept of "virtualized ego" [26], a kind of software agents quite similar to the ones proposed for the Cooperative Web. Han and Chang also explain the need for automatic building of documents taxonomies [27]. Cercone et al state the future relevance of recommender systems and software agents in the Intelligent Web [28].

4.1 Concepts vs. Keywords

The retrieval of information using keywords has many drawbacks. The use of ontologies can improve precision in some cases. However, developing ontologies to support any conceivable query on the Web would be insurmountably hard.

There is a middle point: the use of concepts. A concept would be a more abstract entity (and with more semantics) than a keyword. It would not require complex artifacts such as ontology languages or inference systems. A concept can be seen as a cluster of words with similar meaning in a given scope, ignoring tense, gender, and number. For instance, `(actor, actress, artist, celebrity, star).`

Concepts would be useful if they could be automatically generated and processed as keywords. We think that techniques such as Latent Semantic Indexing [29] or concept indexing [30] could serve this task.

4.2 Documents Taxonomies

The Cooperative Web would use the whole text of the document without using any markup as the source for semantic meaning. How could this be done without the need to "understand" the text? A document can be seen as an individual from a population. Among living beings an individual is defined by its genome, which is composed of chromosomes, divided into genes constructed upon genetic bases. Alike, documents are composed of passages (groups of sentences related to just one subject), which are divided into sentences built upon concepts.

Using this analogy, it seems clear that two documents are semantically related if their ”genome” is alike. Big differences between genomes mean that the semantic relationship between documents is weak.

We think that it is possible to adapt some algorithms used in computational biology to the field of document classification. Similar individuals or species have similarities in their genetic codes so it is possible to classify individuals and species into taxonomies or dendrograms without the need to know what every gene "does".

Such dendrograms cluster different species in “categories” which provide useful information to understand species evolution and confirm (or rebut) the Linnaean classification system.

This system establishes the taxonomic groups basing on noticeable attributes in the living beings, that is the phenotype. Dendrograms, however, build the groups from the genotype, which in turn influences the phenotype. Because of this, categories obtained in an automatic way from the species DNA can be very similar to other categories built by a human.

In the same way, documents could be classified into taxonomic trees depending on the similitude found in their “conceptual genome”. The important thing about such a classification is that it would provide semantics (similitude at the conceptual level between documents or between documents and user queries) without requiring the classification process to use any semantics. In fact, it should be able to cluster documents in categories similar to the ones that a person would build.

4.3 Collaboration between User Agents

The Cooperative Web intends to employ user browsing experience, extracting useful semantics from it. Each user in the Cooperative Web would have an agent with two main goals: to learn from its master (developing a user profile without explicit feedbak), and to retrieve information for him.

Thus, having each user attached to a profile, it is possible to assign to each pair `(profile, document)` a utility level. Having an agent for each user it would be responsible for deciding that utility level. In order for this utility valuation to be really practical, the utility level should be determined in an implicit way (just by observing users’ behavior, without querying them). The utility level should also be assigned to individual passages within a document, and not to the document as a whole.

4.4 Information Retrieval

The agent would have two different ways to perform information retrieval: to find information satisfying a query formulated by the user, or to explore in the background on his behalf to recommend him unknown documents.

To perform both tasks we want to employ two well-known techniques: Collaborative Filtering (CF) and Content-Based Recommendation (CBR). If the agent uses CF it would recommend the user documents that have obtained a high utility level from users with his same (or similar) profile. On the other hand, if the agent uses CBR it would retrieve documents that would be conceptually related with the user profile, with a query or with an initial document, without priorizing the utility level.

The first system would allow “queries” similar to the following ones: (1) “Find documents related to `star`”. Because of the ambiguity of the term, the agent should not provide results but more terms related to the initial one depending on the different topics to help the user to refine the query. (2) “Find documents related to this sentence/paragraph/document”. The user would select a piece of text and the agent would find conceptually related information.

The recommender system would work in a different way. It would be, mainly, a personal assistant that would help the user by performing tasks such as information retrieval on behalf of the user or unsolicited recommendation of interesting not previously visited documents.

5 Conclusion

We have described a proposal to provide Web Intelligence: the Cooperative Web. We have compared it with the Semantic Web and older proposals to avoid information overload in the Internet.

In more detail we have describe the information retrieval techniques that the Cooperative Web could provide. If they are compared with modern search engines we think it is clear that our proposal would obtain less but more relevant results since it would employ conceptual taxonomies.

References

1. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American, 284 (5) (2001) 34–43
2. Gayo-Avello, D., Álvarez-Gutiérrez, D.: The Cooperative Web: A Complement to the Semantic Web. Proc. of 26th Annual International Computer Software and Applications Conference. Oxford, England (2002) 179–183
3. Furnas, G.W., Landauer, T.K., Gómez, L.M., Dumais, S.T.: The vocabulary problem in human-system communication. CACM, Vol. 30, No. 11 (1987) 964–971
4. Pinkerton, B.: Finding what people want: Experiences with the WebCrawler. Proc. of the Second International World Wide Web Conference. Chicago, IL, USA (1994)
5. Krovetz, R., Croft, W.B.: Lexical Ambiguity and Information Retrieval. ACM Transactions on Information Systems, Vol. 10, No. 2 (1992) 115–141
6. Berners-Lee, T.: Information Management: A Proposal `http://www.w3.org/History/1989/proposal.html` (1989)
7. Koster, M.: ALIWEB: Archie-Like indexing in the Web. Computer Networks and ISDN Systems, Vol. 27, No. 2 (1994) 175–182
8. Mauldin, M.L, Leavitt, J.R.R.: Web agent related research at the Center for Machine Translation. Proc. of the ACM Special Interest Group on Networked Information Discovery and Retrieval (ACM-SIGNIDR-V). McLean, VA, USA (1994)
9. Kleinberg, J.M.: Authoritative Sources in a Hyperlinked Environment. Proc. of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms. San Francisco, CA, USA (1998)
10. Page, L, Brin, S. Motwani, R., Winograd, T.: The PageRank Citation Ranking: Bringing Order to the Web. Stanford Digital Libraries Working Paper (1998)
11. Morita, M., Shinoda, Y.: Information filtering based on user behavior analysis and best match text retrieval. Proc. of the 17th Annual International Retrieval. Dublin, Ireland (1994)
12. Maes, P.: Agents that Reduce Work and Information Overload. CACM Vol. 37, No. 7 (1994) 811–821
13. Lieberman, H.: Letizia: An Agent That Assists Web Browsing. Proc. of the 14th International Joint Conference on Artificial Intelligence. Montreal, QC, Canada (1995)
14. Starr, B., Ackerman, M.S., Pazzani, M.: Do-I-Care: A Collaborative Web Agent. Proc. of the ACM on Human Factors in Computing Systems. Vancouver, Canada (1996) 273–274

15. Balabanovic, M.: An interface for learning multi-topic user profiles from implicit feedback. Proc. of AAAI Workshop on Recommender Systems. Madison, WI, USA (1998)
16. Luke, S., Spector, L., Rager, D.: Ontology-Based Knowledge Discovery on the World-Wide Web. Working Notes of the Workshop on Internet-Based Information Systems at the 13th National Conference on Artificial Intelligence (AAAI96) (1996)
17. Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., Slattery, S.: Learning to Extract Symbolic Knowledge from the World Wide Web. Proc. of the 15th National Conference on Artificial Intelligence (AAAI98), Madison, WI, USA (1998)
18. Fensel, D., Decker, S., Erdmann, M., Studer, R.: Ontobroker: Or How to Enable Intelligent Access to the WWW. Proc. of the 11th Workshop on Knowledge Acquisition, Modeling, and Management. Banff, Canada (1998)
19. Marchiori, M., Saarela, J.: Query + Metadata + Logic = Metalog. Proc. of Query Languages Workshop. Boston, MA, USA (1998)
20. Brickley, D., Miller, L.: RDF: Extending and Querying RSS channels. ILRT discussion document. `http://ilrt.org/discovery/2000/11/rss-query/` (2000)
21. Decker, S., Brickley, D., Saarela, J., Angele, J.: A Query and Inference Service for RDF. Proc. of Query Languages Workshop. Boston, MA, USA (1998)
22. Karvounarakis, G, Christophides, V., Plexousakis, D., Alexaki, S.: Querying RDF Descriptions for Community Web Portals. The French National Conference on Databases. Agadir, Maroc (2001)
23. Maedche, A., Staab, S.: Discovering Conceptual Relations from Text. Technical Report 399". Institute AIFB, Karlsruhe University (2000)
24. Erdmann, M., Maedche, A., Scnurr, H.P., Staab, S.: From Manual to Semi-automatic Semantic Annotation: About Ontology-based Text Annotation Tools. ETAI Journal - Section on Semantic Web (Linköping Electronic Articles in Computer and Information Science), 6 (2001)
25. Fensel, D.: Ontology-Based Knowledge Management. IEEE Computer, IEEE Computer Society, Washington, D.C., 35(11) (2002) 56–59
26. Nishida, T.: Social Intelligence Design for the Web. IEEE Computer, IEEE Computer Society, Washington, D.C., 35(11) (2002) 37–41
27. Han, J., Chang, K.C.-C.: Data Mining for Web Intelligence. IEEE Computer, IEEE Computer Society, Washington, D.C., 35(11) (2002) 64–70
28. Cercone, N., Hou, L., Keselj, V., An, A., Naruedomkul, K., Hu, X.: From Computational Intelligence to Web Intelligence. IEEE Computer, IEEE Computer Society, Washington, D.C., 35(11) (2002) 72–76
29. Foltz, P.W.: Using Latent Semantic Indexing for Information Filtering. Proc. of the ACM Conference on Office Information Systems. Boston, USA (1990) 40–47
30. Karypis, G., Han, E.: Concept indexing: A fast dimensionality reduction algorithm with applications to document retrieval and categorization. Technical Report TR-00-0016. University of Minnesota (2000)