WebTribe: Implicit Community Clustering by Semantic Analysis

Damien Leprovost
Le2I - CNRS Lab.
University of Bourgogne
Dijon, France
damien.leprovost(at)u-bourgogne(dot)fr

***Abstract*—Since the advent of Web 2.0, any user becomes a content provider through personal websites, posts on wikis and forums, recommendations, annotations, etc. In this paper, we propose a method to analyze the interests of users based on their publishing activities, by positioning them into a semantic graph. We describe the WebTribe system that allows to extract topic information from collaborative websites and to query the resulting clusters of users.**

Keywords-implicit communities; semantic distance; topic graph.

I. INTRODUCTION

The overflowing data produced by collaborative websites (forums, wikis, etc.) requires new analysis tools. Now, any Web user is no longer a simple reader, but a content provider who publishes information on the network: he is able to share his opinion. Such new data offers new opportunities, and must be analyzed.

In these circumstances, the indexing methods proposed by traditional systems such as user profiles ([1], [2]) may suffer limitations. Indeed, the description of a person's activities, whether by itself or by others, is often simplistic: users are reluctant to spend a precious time filling their profiles. User profiles do not define their precise interests, from the strongest to the more tenuous one, as manifested by the user's activities. Furthermore, profiles often static and can not be updated at any time. We therefore rely on an implicit definition of user interests to detect his/her activities properly.

Our goal in this paper is to identify implicit communities, that focuses on specific topics. Members of these communities are not necessarily aware of their membership, or even of the existence of the community. Indeed, what a user seeks is not necessarily in contact with him. In this sense, implicit communities are strongly apart from communities as they exist in social networks.

The paper is organized as follows: we present the architecture of our system in Section II. Section III defines the semantic topic graph that we construct. Section IV describes how the user is integrated into the graph and the graph querying possibilities. We present the system milestones in Section V. Section VI sums up the related work and we conclude in Section VII.

II. ARCHITECTURE

We briefly present each analysis step of the WebTribe system, and will explicit them in following section. Figure 1 presents the flowchart for our proposal.

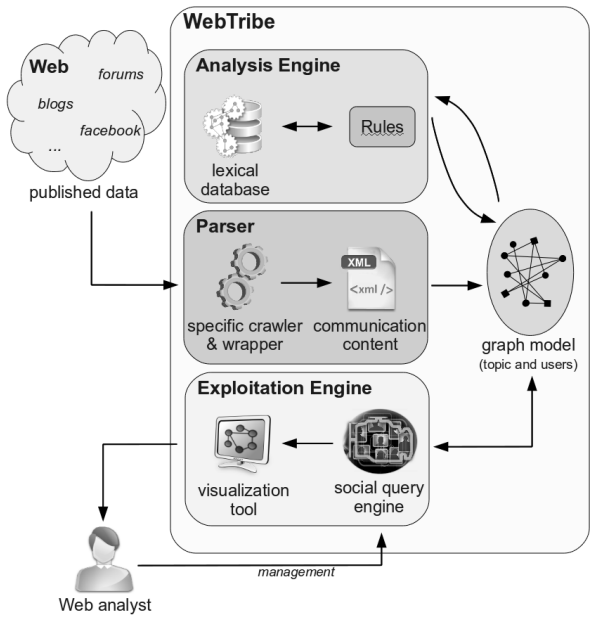

Figure 1. Architecture used for community clustering.

WebTribe has for input various published data on the web, and is managed by a Web analyst, who controls the system. WebTribe is structured around a graph model, and has three internal layers : the *Parser*, which extracts content from given sources; the *Analysis Engine*, which interprets the meaning of content; and the *Exploitation Engine*, which builds communities according to parameters and wishes of the Web analyst.

In the first step, a web analyst provides a list of topic used to build a topic graph, as the basis of our analysis. This list is the lexical database to be used by the *Analysis Engine* to find related content on the analyzed documents. This graph will be potentially pruned for non-relevant topics, and used for semantic user positioning.

In parallel, the *Parser* collects various publications (posts, etc.) from various sources selected by the web analyst, and

 ISBN: 978-1-61208-104-5

associates the publication with its author.

Then, the *Analysis Engine* extracts for each publication its main topics, and quantifies the *publication attractivity* by topics, as the degree of importance of each topic evaluated in the publication. By analyzing all publication found for one author, we are now able to compute the *user attractivity* of this author. Using a Web-based semantic distance computing method (see Section IV), we evaluate the distance between topics and locate the user inside this topic graph.

Finally, querying the system through *Exploitation Engine* now means to compute a sub-graph of our results, including users who validate a closeness constraint given by the query, based on previous computed semantic distance.

The system is equipped with a query language and visualization tools that allow the Web analyst to explore sets of users.

III. TOPIC GRAPH

A. Choosing topics

Our method aims to group users according to their affinities with defined topics. The Web analyst has to define which major topic are relevant for the analysis of his system. We call this topic list the *lexicon* of the system. The goal is to have enough topics to cover all of users. But having too many topics is not desirable either, unnecessarily burdening the system. We propose, at the end of this section, a method for pruning topics so that only useful topics remain.

Example 1: The Web analyst of a car fan forum submits the following lexicon: `Ferrari`, `Porsche`, `tuning`, `petrol`, `dealership`, `engine` and `fuel`.

B. Topic graph

Once defined all system topics, we have to organize them. To put them all into a weighted semantic graph, we use a Web-based semantic distance computing method [3], to evaluate the semantic distance between a term x and a term y. This method is well suited for our approach, because it does not extract the semantic distances from predefined ontologies, but from the Web content (through what Google sees, which seems to be the best viewpoint available). Since we intend to bring together users based on their activity on the Web, this method seems very appropriate to our context.

Therefore, the semantic distance between x and y is defined as follow:

$$\text{DIST}(x,y) = \frac{max\{\log\ f(x), \log\ f(y)\} - \log\ f(x,y)}{\log\ M - min\{\log\ f(x), \log\ f(y)\}},$$

where $f(\lambda)$ is the frequency of the term, and M the total number of indexed terms. Using Google, $f(\lambda)$ means that the number of results to the "λ" query, and M the number of documents indexed (estimated at 1 trillion).

This expression calculates the lowest probability of $x|y$ and $y|x$, where $|$ means conditional probability, using a negative logarithm to increase the difference significance, and standardized by a division to solve scale problems.

Finally, our topic graph is a complete graph with topics as vertices. An edge between topics t_i and t_j is annotated by their distance.

Example 2: With previous lexicon, WebTribe computes the following semantic distances:

	fuel	engine	dealership
Ferrari	1,3478	1,6431	1,0418
Porsche	1,1140	1,4399	0,9475
tuning	1,3064	1,4529	0,7161
dealership	0,8998	1,1027	-
engine	1,0774	-	-

	tuning	porsche
Ferrari	1,3010	0,4195
Porsche	1,1301	-

Table I
COMPUTED DISTANCES USING EXAMPLES' LEXICON

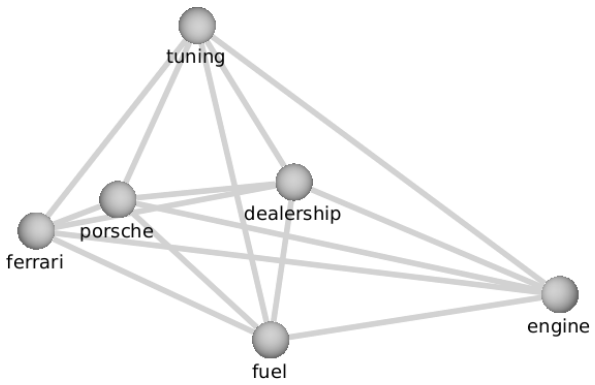

Figure 2. Example of Topic Graph

C. Pruning topics

The resulting graph of topics is a complete graph. In order to be as relevant as possible, but also be easily used, it must be as small as possible. Indeed, the more the number of topics is small compared to the number of content analyzed, the more shades of distances between users have an interesting meaning.

For these reasons, we prune topics considered non-relevant. A topic is not relevant when it is too close to another semantically. That is, when distance is smaller that a threshold δ_s given by the Web analyst. When this happens, the topic with the lowest frequency is removed. In other words, the remaining topic is considered to represent a concept encompassing the pruned topics. This reduces the graph size, making it more relevant, and its use more efficient. It also allows a feedback to the system owner, notifying him of the irrelevance of some of the topics he has chosen.

 ISBN: 978-1-61208-104-5

Example 3: We use the previous lexicon (see Example 1). Topic `petrol` has been pruned, considering its proximity with `fuel` lower that the threshold δ_s.

IV. User attractivity & Querying

A. User Data Acquisition

From the source of data we analyze, we retrieve the various publications of system users. The source system may be a website, a blog, a social network, a online newspaper allowing comments, or any platform allowing users to publish content. This operation can usually be done by a wrapper specifically designed for the given source, including a specific parser and outputting data in a normalized format. One can also rely on classical API to extract information such as the Facebook API.

B. Publication attractivity

For each analyzed content, we look for extract main topics and for each one, define the publication attractivity by topics. We use the previously pruned lexicon $(t_1, \ldots, t_n)$ containing all topics relevant to search. If there are n topics in the lexicon, we can figure that each topic t is assigned a dimension in vector space, then the lexicon is the basis of a n-dimensional hypercube. Every publication p may be thought as a topic vector in this space, so $\bar{p} = (p_{t_1}, \ldots, p_{t_n}) \in \mathbb{R}^{+^n}$.

To determine the topics addressed in a publication, we use a derivative work of Das et al. [4]. This method involves analyzing the document with five different algorithms to determine with a simple majority if the text contains a feeling about the topic (positive or negative), or not relevant at all. As we consider the interest and not the opinion, we interpret both feelings as a positive vote as interest. Based on it, we build a vector for each publication.

This method, using five different algorithms and a base dataset initialized by the Web Analyst, has the advantage of providing relevant and reliable results by not raising the content that does not win the majority. In other words, quality over quantity analysis of information extracted. As an interesting side effect, it also allows us to eliminate spam messages. They did not win the majority of tests, and they are simply ignored.

Example 4: Considering the previous lexicon (see Examples 1 and 3), the publication "`Review of my new Carrera`" will be mapped to:

$$p = (0, 5, 0, 0, 3, 1).$$

This means that the topic `Porsche` is considered highly relevant (Carrera is the name of car series build by Porsche). The topic `engine` is identified as a topic with average importance inside the document, and `fuel` as a minor topic. Topics `Ferrari` and `dealership` are considered non-relevant from the document.

C. User Attractivity

Based on all collected publication attractivity, we are now able to compute the user attractivity as a vector of the same type as previously. We define this u vector, with $u \in \mathbb{R}^{+^n}$, such as $u = \sum p$ with p being publication of the user.

We use a sum rather than normalizing these results, in order to maintain the independent nature of the rate of involvement. For example, if a user is the author of numerous contributions related to a given topic, normalizing the results would reduce its importance in this topic community if it publishes many documents in another independent topic. It makes no sense in this case.

D. Graph

We now have a semantic graph of the lexicon (see Section III), and a vector attractivity u per user. We translate these vectors into semantic distance, as follows:

$$\textsc{Dist}(u, t_i) = \frac{1}{\log\ u_{t_i}}$$

Finally, users are positioned on the graph, according to their attractivity.

Example 5:

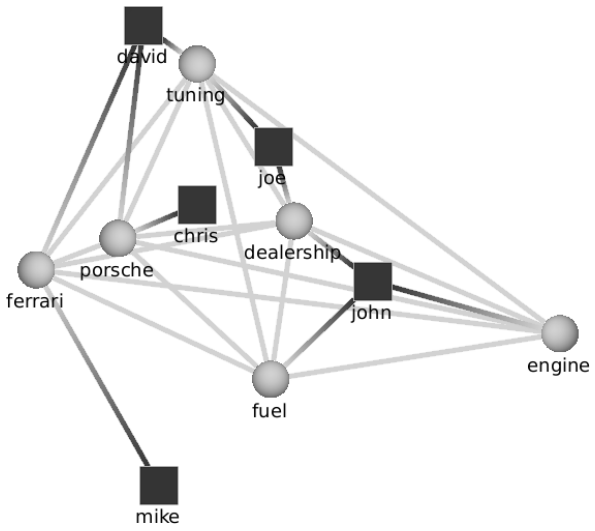

Figure 3. Example of Topic Graph after user positioning

E. Querying Communities

After users have been positioned semantically, it is possible to group them according to given parameters. By "parameters" we mean the choice of one or more subjects, with their logical operators if necessary, and a threshold.

An user u is considered as a member of the community of topic t_i, if $u_{t_i} < \delta_C$, where δ_C is a threshold set by the Web analyst.

This method allows to dynamically build communities, and adjust the threshold according to the needs of the query.

 ISBN: 978-1-61208-104-5

Example 6: According previous lexicon, the Web analyst can perform boolean queries, such as

$$(\text{dealership} \cap \text{engine}) - \text{Ferrari},$$

that selects user talking about dealership and engine, but not for Ferrari.

F. Viewing

The results of previous queries are nodes of the graph, with weighted relations between them, based on semantic distances. This allows to represent the community resulting from the query as a graph, which can be visualized by the web analyst and exploited by him.

G. Incremental Issue

1) New publication: The system is planned for a continuous crawling of the targeted sites, and a scalable analysis. For example, when a target receiving new messages, they must be added to the analysis. Because the formula of the semantic distance between a user and a topic is invertible, we do not need to store user attractivity vectors (see above). For each new publication, it is just needed to extract all of its attractiveness topic. For each topic t_i evaluated with an attractivity a, we update the semantic distance between the user u, author of the publication, and the topic t_i as follows:

$$\text{DIST}'(u, t_i) = \frac{1}{\log\ (10^{\frac{1}{\text{DIST}(u,t_i)}} + a)}$$

2) New topic: For various reasons (policy, new behaviors occurrence, etc.) the Web analyst may need to add new topics to the lexicon. Then, if the new topic is relevant (see Section III), we locate it in the topic graph as usual. After that, we have to evaluate the distance between all users and it. If the whole log of old publications is memorized, we compute the publication attractivity the new topic for each publication, and define a new semantic distance for users as usual. If we do not have archives, we approximate the new distances. As we know the semantic distances between the old topics and new one, we evaluate the distance from each user to the new topic as the value of the shortest path between them.

V. Project Milestones

We extracted several thousands of user comments to USA Today [5], an U.S. online newspaper. All these contributions are signed by their authors, who are identifiable (authenticated users). This extraction was performed by a wrapper specifically developed for USA Today, including HTML and JSON parsers. All contributions are stored as standard XML documents.

Early versions of our semantic graphs have been produced in GML format for viewing. Our graph visualizations have been produced with Tulip [6]. We plan to implement a SQL storage, to take into account the transitivity problems of a system operating in real time.

To develop the use of the system by the web analyst, we plan to define a social query language, which performs logical operations (union, intersection, complement, etc.) on the semantic graph.

VI. Related work

Since the Web birth until now, the community concept has evolued. Many works propose different approaches, depending on whether we consider a community as a set of Web pages, or as a group of people sharing a topic of interest.

Discoverning Web Communities

Since the early work on discoverning Web communities [7], hyperlink is used as a discovery basis. a major contribution in this regard is the Kleinberg HITS algorithm which defines the notions of authorities and hubs, structuring a community [8].

Imafuji et al. [9] define a page as member of a community if this page is more referenced from inside the community than outside. They use a maximum flow algorithm to isolate the nodes belonging to a community, based on the algorithm proposed by Flake et al. [10].

Dourisboure et al. [11] then identify, within a Web graph, communities as many dense bipartite sub-graphs in this graph. The bipartite graph represents for one side the interests of the community (according to the authorities HITS) and for the other side those who cite the community (the hubs). This method identify possible sharing of similar interests in different user communities, or rather the sharing of the same user group in different topic communities.

These approaches provide an advanced link analysis between pages, making topic communities, but however do not to bring users to their interests or activities: the hyperlink sharing is no longer necessarily the basis of the exchanges of the collaborative Web (content evaluation by the user, tags, etc.).

Semantic Distance

Cattuto et al. [12] propose another statistical approach for evaluating semantic distances. They validated it on data from the `del.icio.us` [13] website. This website has community structure, and the authors use the annotation data to construct a weighted network of resources. In this context, the similarity between resources is proportional to the overlap of their set of tag, representing a topic. To take into account the tag representativeness, the TF-IDF method is used. The authors propose to detect communities of users by the similarities of their tags. They use the Pearson correlation coefficient as similarity measure, and then apply methods of partitioning. As they do not reduce the number of tags handled, the tag set may be extremely large.

 ISBN: 978-1-61208-104-5

Recommendation Systems

The topic combination is also used in the recommendation systems. By defining the system *Socialranking*, Zanardi et al. [14] do an enrichment query based on tag similarity, based themselves on their common appearances on different resources. Another approach is proposed by Hotho et al. [15] under the name *FolkRank* and again using the graph theory. This approach use *PageRank* to model the relationships between resources, users and tags. This approach, which more exploits the sparse relations, is also explored by Bertier et al. [16] under *Gossple*. The authors use the probability of moving from one tag to another as an indicator of their similarity. Dziczkowski et al. [17] propose a recommendation system based both on the automatic analysis of uses (activity) and profiles written by users. Their method emphasizes the importance of linguistic classifier in understanding the user. This is one reason why we chose the mixed solution of Das et al. [4].

VII. Conclusion

In this paper, we have presented a complete system based on the analysis of user publications. We extract communities that depend on common interests of those users, based on their activities. The communities generated are depending of Web analyst query, validating the fact that there are no absolutes communities, but communities on application.

In order to provide an experimentation, this work will be extended so that social interactions between users are extracted, based on, for, example, forums threads. We plan to developp a complet tool that will allow the Web analyst to fully discover and exploit his communities, as explained on this paper.

References

[1] F. Abbattista, M. Degemmis, N. Fanizzi, O. Licchelli, P. Lopes, G. Semeraro, and F. Zambetta, "Learning user profiles for content-based filtering in e-commerce," in *AI*IA 2002: Proceedings of the 8th Congress of the Italian Association for Artificial Intelligence*, 2002.

[2] R. Carreira, J. M. Crato, D. Gonçalves, and J. A. Jorge, "Evaluating adaptive user profiles for news classification," in *IUI '04: Proceedings of the 9th international conference on Intelligent user interfaces*. New York, NY, USA: ACM, 2004, pp. 206–212.

[3] R. Cilibrasi and P. M. Vitanyi, "Automatic meaning discovery using google," in *Kolmogorov Complexity and Applications*, ser. Dagstuhl Seminar Proceedings, M. Hutter, W. Merkle, and P. M. Vitanyi, Eds., no. 06051. Dagstuhl, Germany: Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2006.

[4] S. R. Das and M. Y. Chen, "Yahoo! for amazon: Sentiment extraction from small talk on the web," *Management Science*, vol. 53, no. 9, pp. 1375–1388, 2007.

[5] "Usa today news," http://www.usatoday.com/news/, (visited 01/06/2010).

[6] D. Auber, "Tulip : A huge graph visualisation framework," in *Graph Drawing Softwares*, ser. Mathematics and Visualization, P. Mutzel and M. Jünger, Eds. Springer-Verlag, 2003, pp. 105–126.

[7] D. Gibson, J. Kleinberg, and P. Raghavan, "Inferring Web communities from link topology," in *HYPERTEXT'98: Proceedings of the ninth ACM conference on Hypertext and hypermedia : links, objects, time and space—structure in hypermedia systems*. New York, NY, USA: ACM, 1998, pp. 225–234.

[8] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," in *SODA'98: Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1998, pp. 668–677.

[9] N. Imafuji and M. Kitsuregawa, "Effects of maximum flow algorithm on identifying Web community," in *WIDM'02: Proceedings of the 4th international workshop on Web information and data management*. New York, NY, USA: ACM, 2002, pp. 43–48.

[10] G. W. Flake, S. Lawrence, and C. L. Giles, "Efficient identification of Web communities," in *KDD'00: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM, 2000, pp. 150–160.

[11] Y. Dourisboure, F. Geraci, and M. Pellegrini, "Extraction and classification of dense communities in the Web," in *WWW'07: Proceedings of the 16th international conference on World Wide Web*. New York, NY, USA: ACM, 2007, pp. 461–470.

[12] C. Cattuto, A. Baldassarri, V. D. P. Servedio, and V. Loreto, "Emergent community structure in social tagging systems," *Advances in Complex Systems (ACS)*, vol. 11, no. 04, pp. 597–608, 2008.

[13] "del.icio.us," http://delicious.com, (visited 01/06/2010).

[14] V. Zanardi and L. Capra, "Social ranking: uncovering relevant content using tag-based recommender systems," in *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*. New York, NY, USA: ACM, 2008, pp. 51–58.

[15] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme, "Folkrank : A ranking algorithm for folksonomies," in *LWA 2006: Lernen - Wissensentdeckung - Adaptivität, Hildesheim, October 9th-11th 2006*, 2006, pp. 111–114.

[16] M. Bertier, R. Guerraoui, V. Leroy, and A.-M. Kermarrec, "Toward personalized query expansion," in *SNS '09: Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*. New York, NY, USA: ACM, 2009, pp. 7–12.

[17] G. Dziczkowski, L. Bougueroua, and K. Wegrzyn-Wolska, "Social network - an autonomous system designed for radio recommendation," *Computational Aspects of Social Networks, International Conference on*, vol. 0, pp. 57–64, 2009.

 ISBN: 978-1-61208-104-5