

Monitoring the evolution of interests in the blogosphere

Iraklis Varlamis^{#1}, Vasilis Vassalos^{#2}, Antonis Palaios^{#3}

[#] *Computer Sciences Department,*

Athens University of Economics and Business, Greece

¹varlamis@aueb.gr

²vassalos@aueb.gr

³p3020005@dias.aueb.gr

Abstract— We describe **blogTrust**, an innovative modular and extensible prototype application for monitoring changes in the interests of blogosphere participants. We also propose a new approach for the analysis of weblog contents that is supported by **blogTrust** and can yield new insights on the analysis of the blogosphere by monitoring the convergence or dispersion of blogosphere interests. The proposed process classifies weblog posts in predefined categories, generates a feature vector for each weblog from the post classification results, clusters together blogs with similar topics/interests, and, via visualization techniques, enables the detection of interest convergence or divergence among bloggers over different time periods. **BlogTrust** uses established, robust data mining techniques to support every step of the process.

The motivation for the work is a hypothesized strong connection between important (global or "local") events and the rapid reduction in the divergence of (global or "local") weblog topic coverage. We use preliminary experiments on a real data set as our running example. These preliminary experimental results provide support for our hypothesis, indicate the most critical points in the proposed process, and point to interesting directions for further research.

I. INTRODUCTION

Weblogs (or blogs) are a popular way for expressing personal opinions and interests on the Web. Blogs' freshness and directness has changed the sequence of reporting events. In several cases, blogs are often alerters of real-world events that have not been covered by the press, or, for local-scale events, that never reach the newspapers.

Popular search engines and new specialized engines (Google Blog Search, BlogPulse, Feedster, BlogScope [4]), offer trend analysis services that monitor the popularity of terms and topics over time, track conversations etc. It should be noted that capturing bloggers' interests with simple topic tags or term frequencies, as is the case with all existing tools, is quite limiting: similar interests can be expressed with similar or contrasting tags, related or unconnected terms [2] and similar terms can be used to express a variety of interests or topics. Traditional trend analysis methods fail to detect this similarity and consequently emerging interests or events may not be traced. As an example, checking the results of BlogScope and BlogPulse for "greece turkey football euro qualifying", we are expecting a popularity peak at October 16, when an important football match took place between the two countries. Both resulting trend graphs are based on a few dozen posts and

have many peaks, not including one on October 16th. A second query, for "greece turkey", matches many more posts, and included many more peaks, reflecting the fact that there are many issues and events that concern the two neighbouring countries. Focusing our analysis on similar clusters of blogs would help us easily notice such events of interest.

The paper introduces a new, promising approach for defining and monitoring blog interests. The blogosphere contents are examined per blog and not per post and the analysis is performed on the coincidence between blog contents and not on the increased appearance of certain topics or terms. Similarly to newsforums, we are interested in the cases where collections of blogs discuss similar topics without necessarily using the same buzzwords. Clusters of blogs that discuss similar topics are the "communities" of our blogosphere. These communities gather and disband when interests converge or diverge respectively. Taking interests' convergence one step further, we hypothesize that there is a connection between real-world events and the sudden convergence of bloggers' interest (all bloggers publish on the same topic) and provide initial evidence of this connection.

The suggested blog analysis process follows the typical steps of data analysis [7]: a) selection of the blogs to be processed and definition of the topics of interest, b) blog pre-processing, c) reduction of the problem dimensions, d) hidden patterns discovery and monitoring of the patterns' evolution over time, e) visualization and interpretation of the produced knowledge.

The second contribution of this paper is **blogTrust**, an extensible and modular tool developed to support this and similar blog analysis processes. It uses popular data processing and mining algorithms, and can be extended easily with more sophisticated algorithms and techniques. **BlogTrust** allows us to perform initial experiments to test our hypothesis regarding the relationship between convergence of interests and real-world events.

The paper primarily focuses on describing the proposed analysis process and **blogTrust** and on providing the foundation for future research on relating blog epidemics and real-world events.

In the following section we present an overview of research work in blog content analysis with emphasis on the temporal aspect of blogging and the relation of blog contents with real world events. Section 3 presents the phases and steps of the

suggested blog analysis method. Section 4 presents the prototype application developed to support the method and section 5 summarizes its features. Finally, section 6 presents an example of the use of blogTrust in real blog data, and section 7 the conclusions of our work and our next steps.

II. RELATED WORK

The blogosphere is been examined under three different scopes in the literature: contents, graph structure and usage. Blogs carry additional information on author, publishing date, backward links, affiliated blogs etc, which may lead to interesting conclusions if exploited properly.

In blog link analysis, additional information is exploited in order to discover important blogs [8], [15] and to analyse blog epidemics [1].

Semantic analysis of blog contents aim to discover topic trends [3], [5], [9] and variations in bloggers' mood [17]. Clustering of contents and links [14] reveals tags that represent similar topics [13] and groups of cross-referenced bloggers. The evolution of the blogosphere is consequently represented as change in the clusters of tags and movement of bloggers from topic to topic over time [13].

In the aforementioned works blog samples are acquired from different sources, i.e. blog directories, open crawling of the web, aggregators or large blog applications (Technorati, Blogger, Livejournal) etc. To examine the time dimension, datasets are selected to cover a few weeks or months. Publicly available tools, such as Blogpulse Trends, MoodViews, and BlogScope [4], analyse blog contents and offer graph representations of the evolution of the blogosphere in different time scales [17], [20], [21].

The drawback of existing approaches in capturing the blogosphere sentiment is that they examine blogs in a micro-scale, under the prism of terms and semantics used in each post. They represent evolution over time for a single concept or sentiment but miss the community aspect which is present in the blogosphere.

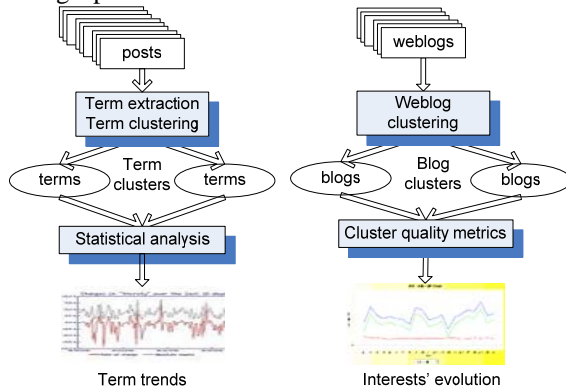


Fig. 1 Trend analysis and interests' evolution

In contrast to existing work, instead of looking for important terms, or clustering terms into topics or moods (Fig. 1, left), we suggest clustering blogs and then visualise the evolution of clusters over time (Fig. 1, right). We cluster blogs based on content similarity (more features can be exploited in order to discover blog communities [16]) and use a set of clustering quality metrics to portray the compactness and

discreteness of clusters. The result is a high level clustering of blogs (implicit interest-based communities).

Similarly to [14], we use content clustering to group blogs and a time-series analysis to monitor changes. However, we don't assign a single tag to each blogger, nor do we monitor the flow of bloggers from one topic to another. We rather look for those cases where blog contents focus around a few topics and form well defined and strong groups. The rationale behind our approach is to process content under the wider prism of bloggers' groups and their interests and produce knowledge concerning the convergence or diffusion of groups' interests and not the dominance of certain interests/topics against others. As a result, the post contents are processed only in order to group blogs together and not in order to provide us with frequent terms. The time aspect is examined over the blog groupings and not over the term clusters (representing buzz topics, moods, sentiments etc).

In the following we summarize the blog analysis process and demonstrate the blogTrust application and the solutions available so far for each module.

III. BLOG ANALYSIS PROCESS

The general aim of the data analysis process [7] is to exploit accumulated data in order to find business, operational, or scientific knowledge. The typical process involves the following steps: a) selection of a target dataset that is of special interest, b) preprocessing of this set (data cleaning, outliers and missing data handling, data representation and storage), c) transformation and reduction of data, d) mining of data and extraction of patterns, e) interpretation and evaluation of the extracted patterns.

Similarly, our blog analysis process aims at finding new knowledge by monitoring blogs and involves (see Fig. 1 Fig. 2: actions below arrows, output below rectangles): a) selection of blogs and collection of their posts for a period of time, b) keyword extraction and representation of each post in the keyword space, c) mapping of the keyword representations of blogs to topic representations using a classifier, d) clustering of blogs based on topic similarities, e) visualization of the clustering scheme features.

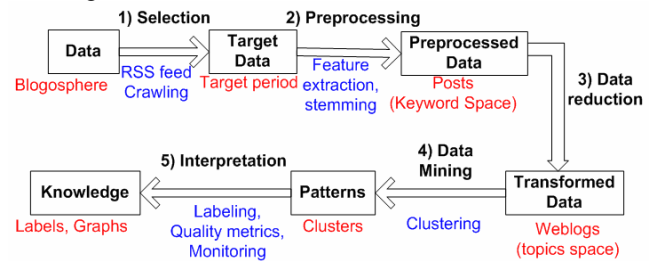


Fig. 2 The data analysis process and our approach

The last step comprises cluster labelling [18], and/or clustering schema metrics (i.e. compactness, separateness, homogeneity etc.) presented in charts [10].

Our aim is to group blogs together based on the topics they discuss and consequently monitor how this grouping evolves over time. Processing is repeated for each time slot (i.e. day, hour, etc) in the period of interest and the evolution of the clustering metrics is monitored. This evolution is the added

knowledge emanating from the suggested process. The interpretation of this knowledge, e.g., the association between clustering schemas and real world events is a subject of investigation, for which we present some preliminary results.

IV. THE *BLOGTRUST* APPLICATION

To support the proposed process as well as other blogosphere analysis tasks we built *blogTrust* (Fig. 3), an extensible, modular tool that processes blog contents and visualizes how blog clusters evolve. In this section we describe *blogTrust* by detailing its modules and how they support the proposed blogosphere analysis process. We focus on the techniques used in each step (one or more of which have already been implemented in the prototype). We also describe the extensibility mechanism that allows more techniques to easily be incorporated in *blogTrust*.

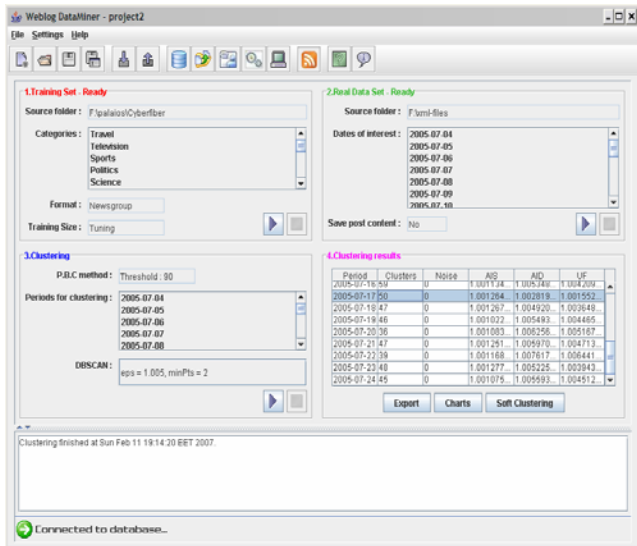


Fig. 3 The *blogTrust* application main screen

Due to lack of space we only briefly describe the "Corpus selection" and "Document pre-processing" phases (see [19]). As depicted in Fig. 4, the application supports Atom and RSS to read blog feeds and allows batch processing of blog posts from XML files. Document pre-processing involves content processing (keyword extraction, stopword removal, stemming, TF*IDF weighting etc.), small posts removal and posts representation to the topics vector space.

Let us note that the selection of topics and training documents is application dependant, and the training set must provide appropriate coverage of the community of interest (whether it is all of the blogosphere or geographical or other subsets) and must be frequently updated in order to follow emerging blogosphere interests. Also note that it is preferable to process individual posts in the first two steps instead of merging all blog contents in a single file: we are then able to perform day by day clustering of blogs (by aggregating posts per day) or to perform bloggers' clustering (by aggregating posts per author) in a later phase.

A. Classification and dimension reduction

Classification is used to categorize posts into predefined categories and consequently reduce the dimensionality of the clustering phase. It involves the following steps:

1) Classifier training

The classifier (i.e. decision trees, decision rules, k-nearest neighbors, Bayesian approaches, neural networks, regression-based methods, and vector-based methods [2]) is trained using a set of pre-categorized documents in order to create classification patterns. The application currently uses the **centroid algorithm** [12] to classify posts, but allows the addition of additional algorithms in a plug-and-play fashion. During training, a centroid vector \vec{C} is created for each topic based on the set of training documents S falling into this topic.

$$\vec{C} = \frac{1}{|S|} \cdot \sum_{d \in S} \vec{d}$$

Cross validation is used to evaluate the classifier and the cosine function is used to measure similarity between training documents or posts and centroid vectors. The computational complexity of training, when using the centroid algorithm, is linear to the number of documents and to the number of terms in the training set. The classification complexity for a new training document is linear to the number of classes. The size of the training set can be optimized (using cross validation) to be minimal without affecting classification accuracy.

2) Classification and post tagging

A post d_i can be classified to: a) the most similar topic, b) the k most similar topics or c) the most similar topic and a few more that are almost as similar. The result is always a set (singleton or not) of topics for d_i and respective weights computed using the cosine similarity.

$$d_i := \langle Cat_1: Sim_1, \dots, Cat_m: Sim_m \rangle$$

The time required to classify d_i is at most $O(kn)$, where n is the number of terms present in d_i . Thus, the overall computational complexity of this algorithm is very low, and is identical to fast document classifiers such as Naive Bayesian and C4.5 [12]. Any two-step supervised document classification algorithm (i.e. naive Bayes classifier, support vector machines, neural networks, k-Nearest Neighbours, decision trees etc) will work similarly. We are currently working on adding more algorithms to *blogTrust*.

3) Creation of the blog's feature vector

After classification, each post has a representation in the vector space of topics. In order to further reduce the amount of data, we group post information per blog and per time slot (i.e. day). This is reasonable, since a blog is itself a small community (or a single "unit") and we are interested in the common interests of the community in a period of time and not in the contents of a single post. Moreover, it is expected that processing posts in this aggregated manner will result in a stronger, collective indication of interests.

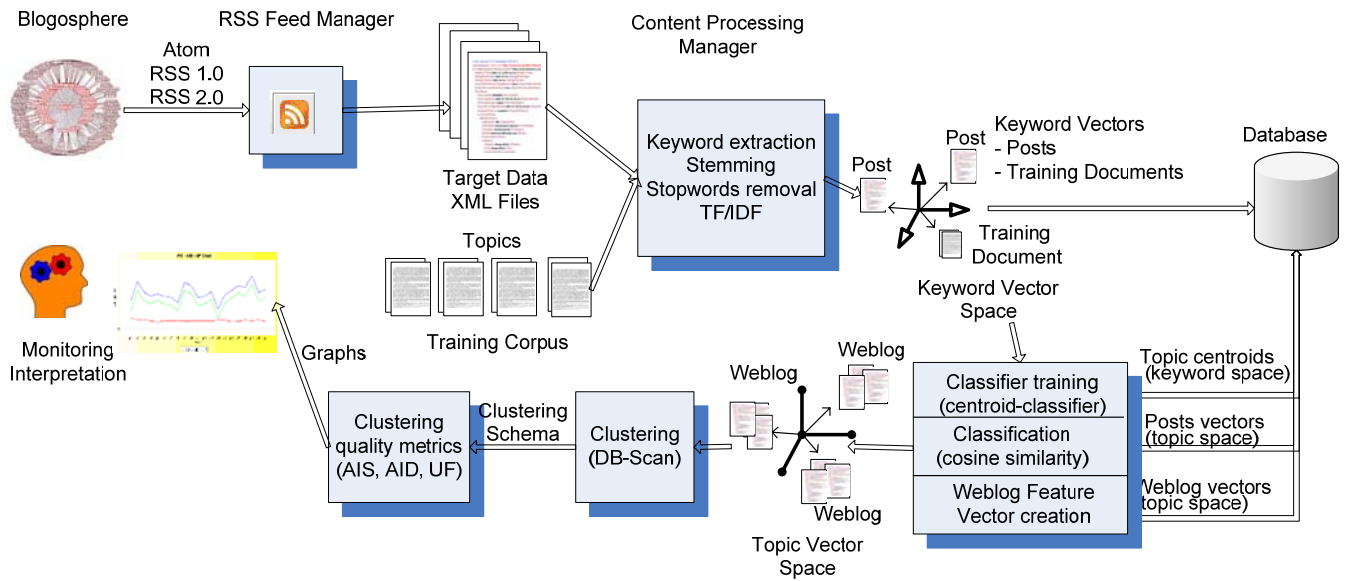


Fig. 4 The application architecture and information flow

All topics or the top-k topics or the most prominent topics can be selected. In the latter option, the ratio $(Sim_j/MaxSim)$ determines whether a topic j is “prominent” or not. When the ratio is zero, all topics are used, and when it is 1 the most similar topic is assigned to a post. An auto-tuning option is available in order to automatically determine this ratio value that less affects the number of “prominent” topics.

The supervised step of classification as described above is performed in order to reduce the dimensions of the clustering step that follows. We can instead perform blog clustering in a different way, for instance, by using Latent Semantic Analysis to locate the most important features for blogs, using hyperlink information instead of contents, or using other blog features to perform clustering.

B. Clustering

Clustering algorithms group together blogs that share similar characteristics. BlogTrust implements the density based clustering algorithm DBSCAN [6], and supports any other document clustering algorithm that implements a simple clustering algorithm interface (an apply method and a set of input parameters). For more information on DBSCAN and its parameters, see [6]. The result of the clustering phase is a clustering schema, containing clusters of topic related blogs and some unclustered blogs (noise).

C. Clustering quality metrics

Blogs are grouped into clusters based on the similarity of their contents in a certain period of time. Clusters are not known a priori, and depending on the clustering method and input parameters, the final partitioning of data requires evaluation. A mechanism for automatically tuning the algorithms' parameters and finding the best clustering scheme is provided in the prototype application. Clustering quality metrics (compactness, separation [11] and a utility function that combines the two) help in finding the optimal clustering scheme for a period. Clustering quality metrics give an

indication on how clustering evolves over time. The utility function $UF = \frac{separation}{Size\ of\ noise \cdot compactness}$ takes into account

the number of unclustered blogs (noise), the compactness and the separation of the clusters, and favours the clustering schemes which contain little noise and together present a good separation. The best clustering scheme is the scheme with the maximum UF value.

Clustering quality metrics provide a quantifiable aspect of the blogosphere evolution. This aspect can be enriched with cluster labeling, or other qualitative information. An advantage of the proposed method is that the convergence or divergence of blogosphere interests corresponds to the peaks and valleys of the graph, whereas it is difficult to be noticed by, e.g., examining the cluster labels. The following section gives a summary of the modularity and extensibility features of the prototype version of blogTrust.

V. BLOGTRUST FEATURES

The blogTrust prototype employs a modular architecture and supports different algorithms, inputs and metrics for every step of the analysis process. Specifically:

- Posts can be imported through XML files, or collected via RSS feed. A SAX parser retrieves post contents and the post and blog feature vectors are stored in a database. The database schema is automatically generated.
- The training set documents (newsgroup messages or plain texts) are grouped into topics using the Topics Management Panel. The training set size can be defined manually or automatically.
- English and Greek content can be processed. TF*IDF and log-likelihood weighting schemes can be chosen.
- The centroid-based classifier has been incorporated in the application and the kNN is under development.

- The Clustering Configuration Panel allows choosing different methods for finding "prominent" categories, grouping dates of interests into longer periods of interest and finally picking a clustering algorithm (currently DBSCAN is the one available) and defining its parameters (manually or with automatic tuning).
- The application visualizes the evolution of three different clustering validity metrics (inter cluster similarity, intra cluster dissimilarity, and the utility function).

The application structure allows new algorithms to be incorporated easily. BlogTrust supports extensibility by exploiting object-oriented programming features (i.e. design patterns that minimize coupling between modules and increase cohesiveness). The adoption of generic interfaces for all data analysis steps allows one or more implementations to be developed and plugged in easily. BlogTrust supports auto-tuning of parameters for the appropriate process steps (e.g., clustering), achieving best results with minimum user interaction (see [19]).

The following section presents an example of the use of BlogTrust over a collection of blogs.

VI. EXPERIMENTAL USE OF BLOGTRUST

The following example provides evidence on the efficiency of the algorithms and the effectiveness of the proposed process in monitoring changes in the blogosphere's interests. Cyberfiber is the source of our training documents and categories. A sample blog dataset provided by Nielsen BuzzMetrics, Inc. is our blog dataset. The dataset spans a period before and after an important event: the London bombings (4/7/2005 – 24/7/2005).

The experiments were performed on a PC (Pentium 4, 3.2 GHz, 2Gb memory, 200 Gb hard drive running Windows XP Server). The same PC hosted the database and the post contents.

A. Choosing the training set

The training set from Cyberfiber comprises 15 top level categories and 11000 newsgroup posts. The selection of a wide and deep training set is important for the quality of the classification process. The use of a set with limited topic coverage results in the misclassification of posts, and consequently affects clustering results, as we discuss at the end of this section.

Feature vectors were computed in less than 9 minutes for all 11000 newsgroup posts. Using auto tuning, in less than a minute BlogTrust determined the optimum training set size to 17 documents per category (255 documents in total).

B. Creating the blog posts feature vector

The parsing of the XML files located 175,301 posts from 2,366 blogs that have postings in every single day (in our period of interest), and created their feature vectors. We selected only blogs that have posts for all the days, so that we can monitor how blog clusters evolve over the given period by day. We use the day granularity for our experiments, and monitor how blog clustering changes per day.

Auto-tuning was used for selecting the most prominent topics for each post. The tuning process used a varying

threshold, ranging from 0.20 to 1 with a 0.05 step (16 iterations) and took approximately 20 minutes to complete. The process is sped up considerably (~1 minute) by selecting the top-k topics or a default threshold value (i.e. 95% gives k=3 topics in average).

C. Blog clustering and metrics

Clustering, using DBSCAN, was performed for each of the 21 groups of blogs (one for each day) with the same parameter values. The clustering quality measures evolution is depicted in Fig. 5. The horizontal axis shows time. The vertical axis corresponds to the three clustering quality metrics.

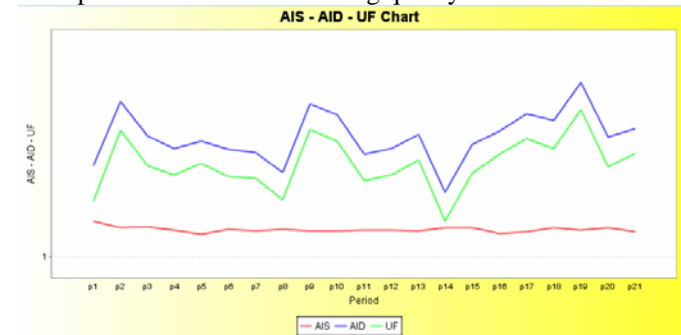


Fig. 5 Evolution of the clustering quality measures

D. Discussion on the results

Our early results (Fig. 5) give an indication of the relationship between real world events and blogosphere reaction: the day after the second group of bombing attempts in London (7/21/2005 – p18 on the graph) UF reaches its maximum value. On July 12 (p9) we have a local maximum. An examination of the timeline of the 2005 London bombings reminds us that on July 12 three suspects were identified. The first bombing event causes a similar effect although to a smaller degree (see p5 on the graph).

Training set topic coverage

In order to demonstrate the importance of topic coverage for the training set, we repeat all process steps, this time using only 12 of the categories of our training set (we omit news, regional and society categories). In Fig. 6 we display the utility function for the 21 days period for the two training sets (complete = 15, incomplete = 12 categories). The vertical dashed lines correspond to the first and second London Bombings. The peaks are obvious when the complete training set is used. The use of the incomplete training set results in a misclassification of posts, which affects clustering and consequently the position and intensity of peaks in the graph.

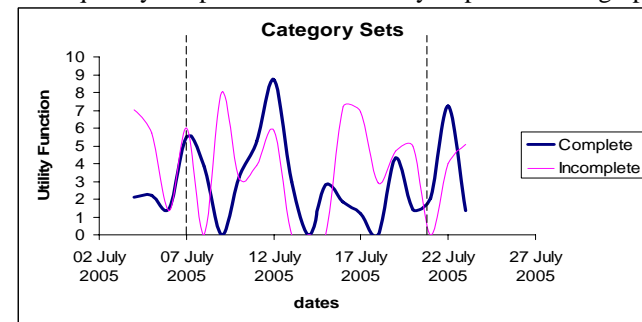


Fig. 6 Utility function evolution using different training sets

As discussed earlier, we can also avoid classification and perform dimensionality reduction using an unsupervised technique (i.e. latent semantic analysis) or cluster blogs using other features (i.e. hyperlinks and graph clustering algorithms). We are exploring this option in ongoing work.

The above examples are indications that the interests of bloggers converge in the presence of (locally or globally) significant events. These results were produced without specific knowledge of the specific topics or interests that attracted bloggers' attention. The exact relationship between the number of clusters, the clustering quality metrics and important events are topics of continuing research for which blogTrust is very valuable. Moreover, blogTrust can be used for the monitoring of specific parts of the blogosphere (e.g., language or domain-specific parts, a predefined set of blogs etc), assuming a proper training. The adopted methodology can also be applied to other parts of the web such as newsgroups or forums. Finally, it can easily be adapted to sentiment analysis of the blogosphere, simply by replacing topics by moods.

VII. CONCLUSIONS

We presented a novel blog analysis process that monitors, analyzes and visualizes interests and topics in the blogosphere over time by dividing the blogosphere into clusters and monitoring cluster compactness and separateness over time. We also presented blogTrust, an extensible application for supporting this and similar blog analysis processes. The application incorporates modules for collecting blog data, analyzing blog contents, classifying and clustering blogs and displaying the evolution of the blog community's interests. It can be easily extended with new data mining algorithms, metrics, and processing steps for monitoring the blogosphere 'state'. Use of blogTrust allowed us to test the interesting hypothesis that an abrupt change of results in terms of the variety of blog interests from one period to the next is indicative of a real world event (that can be global and general-interest, or local and narrow interest). Initial experimental results validate the hypothesis: the evolution of the clustering schema gives useful insights on the bloggers' interests and by consequence on the real world. Further investigation is necessary into the connection between changes in the metrics used and real world events. We plan on experimenting with more focused datasets (i.e. in the greek portion of the blogosphere) and for longer periods of time, and search for convergence of interests associated with smaller (i.e., national) scale events. We also plan to experiment with different parameters: different granularity in time (e.g., hour basis), different topic distribution (e.g., classify post in the more detailed subcategories of CyberFiber, or use another training set), different clustering algorithms and metrics (e.g., entropy). Finally, we are currently working on implementing an unsupervised learning process in order to avoid the challenges associated with training set selection and update.

REFERENCES

- [1] Adar, E., Adamic, L., "Tracking Information Epidemics in Blogspace". IEEE/WIC/ACM Conf. on Web intelligence 2005.
- [2] Avesani, P., Cova, M., Hayes, C., Massa, P., "Learning Contextualised Weblog Topics", 2nd Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, WWE 2005.
- [3] Bansal, N., Chiang, F., Koudas, N., Tompa, F., "Seeking Stable Clusters in the Blogosphere", VLDB 2007.
- [4] Bansal, N., Koudas, N., "BlogScope: spatio-temporal analysis of the blogosphere", WWW 2007.
- [5] Chi, Y., Tseng, B., Tatemura J., "Eigen-trend: trend analysis in the blogosphere based on singular value decompositions", CIKM 2006.
- [6] Ester M., Kriegel H.-P., Sander J., Xu X., "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", 2nd Int. Conf. on Knowledge Discovery and Data Mining, KDD 1996.
- [7] Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., "The KDD process for extracting useful knowledge from volumes of data", Communications of the ACM 39, 11. Nov. 1996.
- [8] Fujimura, K., Inoue, T., and Sugisaki, M., "The EigenRumor Algorithm for Ranking Blogs", 2nd Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, WWE 2005.
- [9] Gance, N. S. and Hurst, M. and Tomokiyo, T., "BlogPulse: Automated Trend Discovery for weblogs". Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, WWE 2004.
- [10] Golden, B., Raghavan, S., Wasil, E., "Assessing Cluster Quality Using Multiple Measures - A Decision Tree Based Approach in the Next Wave in Computing", Optimization, and Decision Technologies Vol. 29, Springer, 2005.
- [11] Halkidi, M., Vazirgiannis, M., Batistakis, I., "Quality scheme assessment in the clustering process." 4th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD),
- [12] Han, E., Karypis G., "Centroid-Based Document Classification Algorithms: Analysis & Experimental Results", Tech.Report Univ. Of Minnesota.
- [13] Hayes, C., Avesani, P., Veeramachaneni, S., "An Analysis of the Use of Tags in a Blog Recommender System", IJCAI 2007.
- [14] Hayes, C., Avesani, P., Veeramachaneni, S., "An Analysis of Bloggers and Topics for a Blog Recommender System". WebMine Workshop on ECML/PKDD, 2006.
- [15] Kritikopoulos, A., Sideri, M., and Varlamis, I., "BlogRank: ranking weblogs based on connectivity and similarity features", 2nd International Workshop on Advanced Architectures and Algorithms For internet Delivery and Applications, Pisa, Italy, 2006.
- [16] Lin, Y.R., Sundaram, H., Chi, Y., Tatemura, J., Tseng, B., "Discovery of Blog Communities Based on Mutual Awareness", 3rd Annual Workshop on the Weblogging Ecosystem, WWE 2006.
- [17] Mishne, G., de Rijke, M., "Capturing Global Mood Levels using Blog Posts", AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW 2006), 2006.
- [18] Treeratpituk, P., Callan, J., "An experimental study on automatically labeling hierarchical clusters using statistical features". SIGIR 2006
- [19] Varlamis, I., Vassalos, V., "Monitoring the evolution of interests in the blogosphere (ext. version)", Technical report. 2007. Available at: <http://wim.aueb.gr/research/blog-techreport07.pdf>
- [20] Blogpulse Trends' web site <http://www.blogpulse.com/trends.html>
- [21] Technorati's "Mentions by day", <http://technorati.com/tag/mentions>