BlogScope: A System for Online Analysis of High Volume Text Streams

Nilesh Bansal
University of Toronto
nilesh@cs.toronto.edu

Nick Koudas
University of Toronto
koudas@cs.toronto.edu

ABSTRACT

We present BlogScope (www.blogscope.net), a system for online analysis of temporally ordered streaming text, currently applied to the analysis of the Blogosphere[1]. The system currently tracks over ten million blogs and handles hundreds of thousands of updates daily. BlogScope is an information discovery and text analysis system that offers a set of unique features. Such features include, spatio-temporal analysis of blogs, flexible navigation of the Blogosphere through information bursts, keyword correlations and burst synopsis, as well as enhanced ranking functions for improved query answer relevance. We describe the system, its design and the features of the current version of BlogScope.

1. INTRODUCTION

Blogs have been proliferating over the last couple of years. It is estimated [9] that the size of the Blogosphere in August 2006 was two orders of magnitude larger than three years ago. According to the same sources, the total number of blogs is doubling every two hundred days. Technorati, a weblog tracking company, has been tracking fifty million blogs in August 2006. Blogging activity includes personal diaries, traveling experiences, opinions (about products, events, people, music groups, or businesses), howto guides, and politics. Collecting, monitoring and analyzing information on blogs can provide key insights on 'public opinion' on a variety of topics, such as products, political views, or entertainment. It can also be a source of competitive intelligence information [4] and market trends [6]. As a result, techniques that aid the collection, analysis, mining and efficient querying of blogs are important and this trend is expected to persist, given the growing popularity of blogs.

BlogScope is a system for online analysis and information discovery of text streams. A text stream in this case is defined as a temporally ordered collection of text documents. The Blogosphere is a natural example of such a text stream. Other examples include news feeds, mailing lists, forums and newsgroups. BlogScope is currently applied to the analysis of the Blogosphere. We confine our discussion to blogs for simplicity of argument, but much of our discussion is pertinent to all temporally ordered streaming text sources.

At the time of writing, BlogScope was tracking around ten million blogs, indexing over 75 million documents in its database. On an average, 14000 new documents are fetched by the crawler every hour. The system records visitors from around 11000 IP addresses everyday, hence serving thousands of requests every hour. It is extremely important, given the analysis the system conducts, for the techniques employed to be computationally efficient in order to scale at this level. We have therefore developed effective and efficient algorithms for burst identification, discovering correlated terms, mining hot keywords, and burst synopsis generation. We describe the analysis paradigm and features of the current version of the system in the next section. A brief overview of the system design and the architecture is provided in the Section 3. Section 4 presents the demonstration plan and concludes the paper.

2. ANALYZING THE BLOGOSPHERE

The information in blogs and its dynamics differ from the traditional web content. Significant differences include: (1) blog posts have a time of creation adding a temporal dimension, (2) blog posts may trigger additional posts by the same or other bloggers leading to a discussion in the Blogosphere, and (3) blog posts can be easily associated with a geographical location which is the same as the location of the author[2]. We introduce BlogScope, a system with enhanced analysis capabilities (well beyond keyword search) for blogs.

The analysis paradigm that BlogScope facilitates is segmented in four steps. BlogScope identifies *what* is 'interesting', *when* it was 'interesting', *why* it is 'interesting', and *where* it is 'interesting'. On its front page, BlogScope displays a list of hot keywords. Such keywords are computed daily from the actual content of blog posts. Based on this list, a user can formulate a query to seek relevant blog posts. The traditional text query interface is also supported to identify posts relevant to a query, in case one is seeking specific information. Once the keywords of interest are identified, a query is formed and relevant blog posts are retrieved. The next question BlogScope aids to answer is when it was

---

[1]Blogosphere is the collective term encompassing all weblogs as a community.

---

[2]Traditional websites, e.g., en.wikipedia.org or www.yahoo.com, do not have a well defined geographical location.
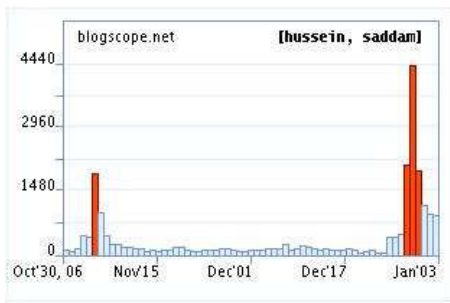
Figure 1: Popularity curve for the keywords *saddam hussein*. Saddam was convicted on November 5 2006, and was executed on December 30 2006. Regions marked in red indicate bursts.
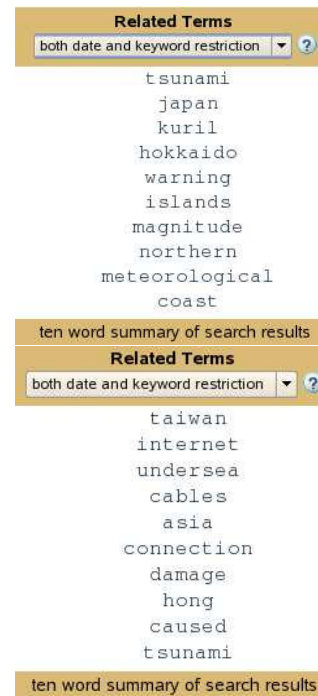

Figure 2: Correlations for keyword *earthquake* for: (1) Top, November 15 2006, soon after the Kuril Islands earthquake happened. The Japanese Meteorological Agency initially estimated tsunami waves to be as tall as 2 metres when it hit the Japanese northern cost, but it turned out to be 40 centimeters when it reached Hokkaido. (2) Bottom, December 27 2006, soon after an undersea earthquake in Taiwan that resulted in disruption of internet services.

interesting. To answer this question, BlogScope plots the popularity of the query keywords in blog posts, as a function of time, and identifies and marks interesting temporal regions as bursts in the keyword popularity. The third step of the analysis is to investigate why it is interesting. Correlated keywords (intuitively defined as keywords closely related to the keyword query at the specified temporal interval) are automatically displayed by BlogScope. Such keywords aim to provide explanations or provide insights as to why the keyword experiences a surge in its popularity. Based on these keywords, one can refine the search and drill down in the temporal dimension towards a more focused subset of blog posts. The final step is to identify where it is interesting. BlogScope associates with each blog its geographical coordinates. This information is used to annotate the world map with regions where bloggers are writing about the searched query.

It must be noted that all the analysis is performed on the actual textual content of blog posts, and not on *tags* because: (1) tagging requires manual effort, (2) most blogs posts are not tagged, and (3) a few tags can not accurately capture complete information present in a post.

2.1 Popularity and Bursts

The popularity curve for a keyword (or set of keywords) displays how often the specified keywords are mentioned in the Blogosphere as a function to time. Such a curve and its fluctuation can provide insight regarding the keyword popularity evolution over time. Figure 1 provides an example of the popularity curve for the query *saddam hussein*.

Although blogging activity is uncoordinated, whenever something of interest to a fraction of bloggers takes place (e.g., a natural phenomenon like an earthquake), bloggers write about it. As a result, the popularity of certain keywords increases. This allows BlogScope to identify and mark such interesting events on a popularity curve. We refer to these events as *bursts*. BlogScope employs a variety of techniques to identify and quantify unexpected popularity. Bursts play a central role in analysis and blog navigation using BlogScope, as they identify temporal ranges to focus and drill down, refining the search.

In addition to aiding navigation in BlogScope, bursts can also be used to produce intelligent alerts for users. Subscribing to specific keyword queries, BlogScope can generate an alert (in the form of email) only when a burst occurs in the

popularity of specific keywords. In this way an alert will be raised only when something potentially interesting has occurred. Currently available alert services (e.g., Google Alerts [5]) suffer from two main problems: (1) An alert is raised whenever any new document (e.g., blog post) containing the query is encountered by the crawler. Discovery of a new document may not necessarily imply occurrence of an interesting event. (2) The number of alerts is large to handle, if the number of documents containing the specified query is large. BlogScope's alert service is free from these two problems.

2.2 Keyword Correlations

Information in the Blogosphere is highly dynamic in nature. As topics evolve, keywords align together to form stories; and as topics recede, these keyword clusters dissolve. This formation and dissolution of clusters of keywords is captured by BlogScope in the form of correlations.

With every search, a list of keywords in blog posts most closely related to the search query keywords is displayed. Such keywords can be seen as representative tokens for chatter in the Blogosphere, and can be used to obtain insight regarding the posts relevant to a query. Roughly speaking, such keywords are those that most frequently co-occur with the searched query terms (weighted by their *idf*). Correlations are not static, as they may change according to the
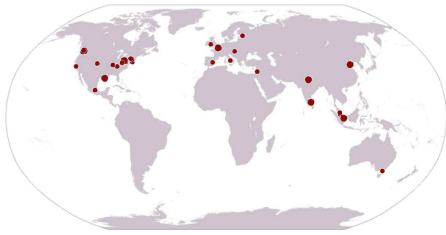
Figure 3: GeoSearch for the query *iphone*. Black dots on the map represent regions where bloggers are writing about the searched query.
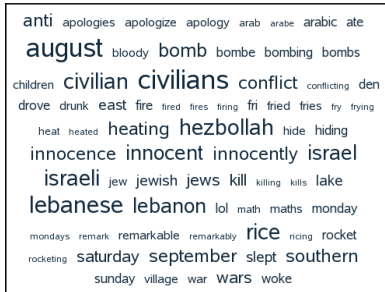

Figure 4: Hot keywords cloud for July 30 2006

temporal interval specified in the query. Users can specify a temporal range for which a list of keywords correlated to the query keywords is produced. Provided that users navigate, drilling down to posts related to a burst, such correlations can be used to reason why a burst occurred. Figure 2 shows a screenshot of correlations for the query *earthquake* for two different time periods. It can be seen that correlations are different across time intervals, and they reflect events occurring in them. Correlations are also employed by BlogScope to provide an exploratory navigation system. A user can easily refine the initial query using correlated keywords and thus focus on more specific posts.

2.3 Spatial Search

With every post that is indexed by BlogScope, a city, state and country is maintained and when possible exact geographical coordinates (in terms of latitude and longitude). As a result of a query, BlogScope has the option to display the blog post results in the selected temporal interval on a map, displaying the distribution of the posts in cities of each country per continent. Users can focus by selecting countries or cities on the map and drill down to the posts in each geographical region. Figure 3 shows an screenshot for the search query *iphone*.

2.4 Hot Keywords

On its front page BlogScope displays a list of *hot keywords* for that day in the form of a cloud tag. BlogScope uses a measure of 'interestingness' for keywords and ranks all keywords for a day according to this measure. Interesting does not necessarily refer to popular. For example, keywords that exhibit sudden change in their popularities are more interesting. In a few occasions, Blogscope tracked popular keywords that corresponded to events that have not made mainstream news media. For example the term *math* was highly popular on the week of August 7 2006 in the Blo-

gosphere as reported by BlogScope. The event corresponded to the news about the Poincare conjecture proof by Grigory Perelman. New York Times had an article on this on August 15 2006. Figure 4 displays an example screenshot taken on July 30 2006.

2.5 Authoritative Blog Ranking

Two features in BlogScope enhancing the spatio-temporal search are authoritative ranking and burst synopsis. The semantics associated with the burst synopsis set for an initial query q is that it is the maximal set of keywords associated with q that exhibits a bursty behavior in the associated popularity curve for the set. Synopsis sets may have an arbitrary size (number of keywords) provided that all included keywords contribute to the burst. Authoritative blogs are blogs read by a large number of readers, and are usually first to report on news. These blogs play an important role in dissemination of opinions in the Blogosphere.

Consider the query 'italy'; blog posts may mention the keyword 'italy' in connection to both soccer and political events. All such posts contribute to the burst in the popularity of the keyword 'italy'. The keywords 'soccer' and 'politics' are both correlated to keyword 'italy' in the associated temporal interval. However expanding the search and observing the popularity curves of 'italy, soccer' and 'italy, politics' turns out that only the curve for 'italy, soccer' has a burst in the temporal interval of the three summer months of 2006. BlogScope can automatically generate such *synopsis* keyword sets for a burst. In this case, only the set 'italy, soccer' will be identified and suggested by BlogScope as a synopsis set, associated with the initial keyword query 'italy'. Notice that the set 'italy, politics' will not be identified as a synopsis set, because 'italy, politics' does not have a burst in the corresponding popularity curve.

Based on such keyword sets, BlogScope automatically ranks blog posts related to the synopsis set based on *authority*. Authoritative blogs are the ones that gave rise to the burst on the synopsis keyword set. These are blogs that are relevant to the synopsis set, temporally close to the occurrence of the burst and most linked in the Blogosphere.

As an additional example, a search using query 'cars' on June 9th 2006 results in the synopsis set {cars, pixar, disney, movie} which disambiguate the burst resulted from the release of the movie Cars, from general discussion about automobiles in the Blogosphere. Such set is accompanied with authoritative blog posts that were the first to report the event and were most linked in the Blogosphere.

3. SYSTEM ARCHITECTURE

This section provides a brief overview of BlogScope's design and algorithms. Details are available in [1]. Figure 5 presents the overall system architecture. The main components of the system are as follows:

Crawler: Crawling the Blogosphere is different from crawling the web. RSS feed is available for most blogs, and the crawler can fetch and parse the RSS XML instead of HTML. There is no need to follow outlinks as services like `blo.gs` and `weblogs.com` maintain a list of most recently updated blogs. The BlogScope crawler receives a list of blogs updated in the last one hour from `weblogs.com` and schedules them to be fetched.

Spam Analyzer: Spam is a big problem in the Blogosphere. Our experience with `blogspot.com` data shows that
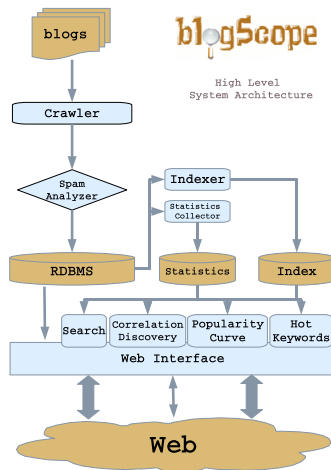
Figure 5: High level system architecture of BlogScope.

half of the blogs are spam[3]. These pages exist to drive traffic to some commercial sites. Spammers use intricate language modeling and javascript obfuscation techniques to generate spam blog pages. Effective spam detection is an active research area [10]. BlogScope's spam analyzer builds upon previous work, utilizing a Bayesian classifier in conjunction with many simple (but effective) heuristics.

Spatial Component: BlogScope associates a geographical location with every blog post. This is done by extracting location string from author profiles, and utilizing approximate match technology (e.g. [2]) against lists of known cities.

Searching and Indexing: The crawler stores all its data in a relational database. Every three hours, newly added data is indexed to generate inverted lists and other statistics like *idf* values over sliding window of length 365 days. These indexes and statistics are used for generating search results and supporting various analysis capabilities in BlogScope.

Popularity and Bursts: Popularity values are maintained precomputed for frequent single word tokens. Popularity of multiword queries is computed by merging the inverted lists for each of the tokens in the query with the list of posts for that day. For burst identification we consider the approach by Kleinberg [7], but discard it for its high computational cost. We instead use statistical tests that are computationally efficient and amenable to stream processing for finding outliers in the popularity distribution for a query.

Keyword Correlations: The notion of collocations is a well studied topic in natural language computing [8]. In information theory, mutual information [3] is commonly used to measure the mutual dependence of the two variables. Pointwise mutual information $c(a, b)$ between two tokens a and b is,

$$c(a, b) = \log \frac{P(a \in D | b \in D)}{P(a \in D)} = \log \frac{P(b \in D | a \in D)}{P(b \in D)}$$

$$= \log \frac{P(a \in D \text{ and } b \in D)}{P(a \in D) P(b \in D)}$$

---

[3]We have manually looked at a random sample of few hundred blogs from `blogspot.com` which is a Google owned blog hosting service.

where $P(t \in D)$ denotes the probability of token t appearing in some document D in the collection $\mathcal{D}$. In words, correlation between a and b is the amplification in probability of finding the token a in a document given that the document contains the token b. Calculation of correlations using such semantics requires checking each pair of tokens. With tokens in the order of millions, calculating $c(a, b)$ using the above formula for every possible pair across several temporal granularities would amount to a large computational effort. This is complicated by the fact that such correlations have to be incrementally maintained as new data arrive. Increasing the number of keywords one wishes to maintain correlations for, from two to a higher number, gives rise to a problem of prohibitive complexity. BlogScope uses co-occurrence information of keywords in a random sample of relevant documents along with the precomputed *idf* values to approximate the mutual information for two keywords.

Hot Keywords: Interestingness is naturally a subjective measure, as what is interesting varies according to the group of individuals it is intended for. Given the difficulty and the subjective nature of the task, BlogScope adopts a statistical approach to the identification of *hot keywords*. A mix of scoring functions are used to identify top keywords for a day. These scoring functions measure the *interestingness* of each keyword by accessing only precomputed statistics (and not data). In order to produce a final list, scores from all different scoring functions are aggregated.

4. DEMO PLAN AND CONCLUSIONS

We presented BlogScope, a text analysis system suitable for temporally ordered streaming text, currently applied to the analysis of the Blogosphere. For the VLDB demonstration session, we plan to let the participants interact with the system directly. Participants can use BlogScope to analyze recent events and their impact in the Blogosphere, generate popularity curves, burst synopsis sets, and keyword correlations, and perform geographical searches. Due to space limitations we were not able to present a detailed account of the techniques developed for BlogScope. Interested participants will be encouraged to discuss the techniques employed, scalability issues, and computational complexity of algorithms in detail. In the future, we plan to continue enhancing BlogScope with several features to improve navigation, information discovery and performance.

5. REFERENCES
[1] N. Bansal and N. Koudas. Searching the Blogosphere. In *WebDB*, 2007.
[2] A. Chandel, O. Hassanzadeh, N. Koudas, M. Sadoghi, and D. Srivastava. Benchmarking declarative approximate selection predicates. In *SIGMOD*, 2007.
[3] K. W. Church and P. Hanks. Word association norms, mutual information and lexicography. In *ACL*, 1989.
[4] Cymfony's influence 2.0: Blog analysis. *http://blog.cymfony.com/blog_analysis/index.html*.
[5] Google Alerts. *http://www.google.com/alerts*.
[6] D. Gruhl, R. V. Guha, R. Kumar, J. Novak, and A. Tomkins. The predictive power of online chatter. In *SIGKDD*, 2005.
[7] J. M. Kleinberg. Bursty and hierarchical structure in streams. *Data Mining Knowledge Discovery*, 2003.
[8] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.
[9] State of the Blogosphere - aug 2006. *http://www.sifry.com/alerts/archives/000436.html*.
[10] Y.-M. Wang, M. Ma, Y. Niu, and H. Chen. Spam double-funnel: connecting web spammers with advertisers. In *WWW*, 2007.