Knowledge Discovery in the Blogosphere

Approaches and Challenges

Knowledge discovery in blogs is different from knowledge discovery in areas such as databases or Web documents due to blogs' unique characteristics, which introduce additional mining challenges. Although researchers have investigated several techniques to address different aspects of blog discovery, no comparisons among key knowledge discovery techniques for blogs exist. This article examines three prominent techniques that are frequently applied to discovery in blogs — clustering, matrix decomposition, and ranking. The authors compare them in terms of effectiveness in combating present challenges and their ability to accomplish challenging tasks required for effective blog mining.

Geetika T. Lakshmanan
IBM T.J. Watson Research Center

Martin A. Oberhofer
IBM Software Group, Germany

Internet users employ Web blogs, or *blogs*, for various reasons, including to disseminate information, discuss ideas, and ask questions. The *blogosphere* is the collection of all blogs and their interconnections, which can serve as a social network as participating bloggers form an online community. Blog content is unique in that it often contains both structured and unstructured information; bloggers often write entries in a rambling, unstructured narrative style, sometimes with spelling and grammatical errors. Bloggers might also use new words and grammar as a means to uniquely express an opinion.

Blogs act as rich sources of knowledge that can serve a variety of purposes.[1] Because of the increasing number of blogs and their unique characteristics, developing techniques for searching and mining them has become important. Individuals can use mined information from blogs to determine topics that are popular at a particular point in time, such as movies or interesting tourist destinations. In addition, we can apply knowledge discovery algorithms to determine why such topics are popular and categorize them according to blogger profiles and communities.[2] Blog recommendation engines, such as the one built in to Google Reader (www.google.com/reader), use mined information from diverse sources, including blogs, to

1089-7801/10/$26.00 © 2010 IEEE

make personalized, relevant recommendations to different individuals. Companies can use knowledge discovered in blogs to profile consumer preferences and obtain direct feedback about products through blog-style product reviews voluntarily written by enthusiastic consumers (see http://online.wsj.com/article/SB124045072480346239.html). Aggregating numerous blogs that offer diverse opinions on the same topic provides valuable collective wisdom and can, for instance, help individuals make a collective judgment about a particular product that they're considering.[3] Analytical tools applied to mine blogs for commercially available products can be helpful in indicating sales volumes and predicting market trends.[4] Figure 1 shows applications for knowledge discovery in blogs.

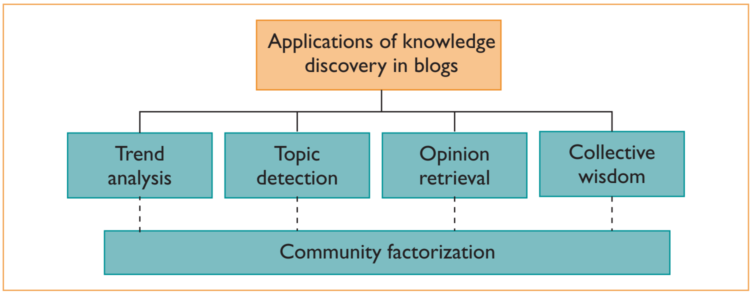

Figure 1. Knowledge discovery in blogs. Knowledge discovery is usually conducted to answer specific questions. We can capture this in terms of these four specific application areas. Some applications might consider the community of bloggers and dynamics between individuals or groups of bloggers.

Several commercial Web sites, such as Blogpulse (www.blogpulse.com), Technorati (www.technorati.com), and Google Blog Search (www.blogsearch.google.com), are available for mining and analyzing blog content. They provide services that include searching blogs based on query keywords, ranking blogs according to popularity, and identifying trends in keywords seen in the blogosphere. However, because blogs are very dynamic, we can't easily apply traditional Web mining techniques to them. Researchers have been tremendously active in inventing and prototyping knowledge discovery algorithms for blogs that address various inherent challenges. To provide a general understanding about the process of mining knowledge from blogs, we outline a framework to characterize the steps involved. We also classify techniques into three knowledge discovery methods: clustering, matrix factorization, and ranking.

Blog Mining Framework

Figure 2 shows our framework for knowledge discovery in blogs, which includes a spider, a parser, a preprocessor for preparing the data for input to the discovery algorithms, discovery algorithms, and a viewer for viewing discovered knowledge.

Blog spiders are similar to standard Web page spiders except that they need to download blog entries much more frequently, possibly even every minute, depending on the frequency with which bloggers update their entries. An alternative to storing and monitoring numerous blogs is to connect to popular blog search engines such as Technorati and Google Blog Search, perform a search on their entries, and combine the results.

A blog parser extracts information from blogs, including keywords or phrases, names of people, products, and organizations, and other patterns, such as dates, times, numerical expressions, monetary amounts, email addresses, and URLs. Before we can apply discovery algorithms to the blog parser's output, some preprocessing must occur that involves converting the parsed information into the required input format for a particular algorithm. For instance, a graph-based ranking technique for blog discovery might require that a set of n blogs be represented as a directed graph with the adjacency matrix G, where $G_{ij} = 1$ if blog i links to blog j, and $G_{ij} = 0$ otherwise.[5] Researchers apply discovery algorithms to the preprocessed data that produce output in the form of numerical or text-based data representing the discovered knowledge. Interpreting these results quickly and intuitively is imperative, and knowledge discovery methods frequently employ visualization techniques such as bar charts and node graphs to accomplish this. Effectively visualizing discovered knowledge from blogs can let users easily identify important trends, comprehend time-sensitive blog topics at a glance, identify interesting communities, and gain a quick understanding of bloggers' collective opinions on significant topics.

Knowledge Discovery Challenges

On the basis of a review we conducted on state-of-the-art work on knowledge discovery in blogs, we compiled a list of the major challenges that govern this problem (see Figure 3a). Some challenges are unique to blogs, whereas others

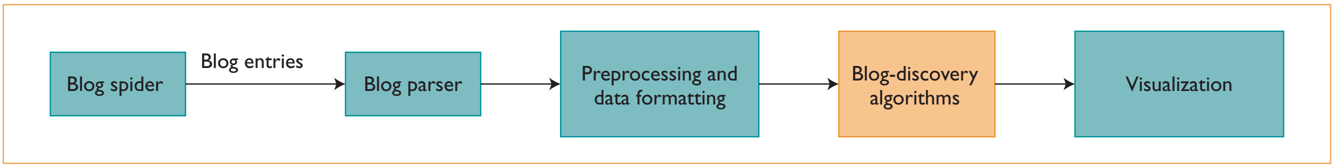

Figure 2. Knowledge discovery process for the blogosphere. Before researchers can execute algorithms for knowledge discovery, systems must first crawl the Web, aggregate different sources of blogs, parse their contents, and provide this data as input to discovery algorithms.

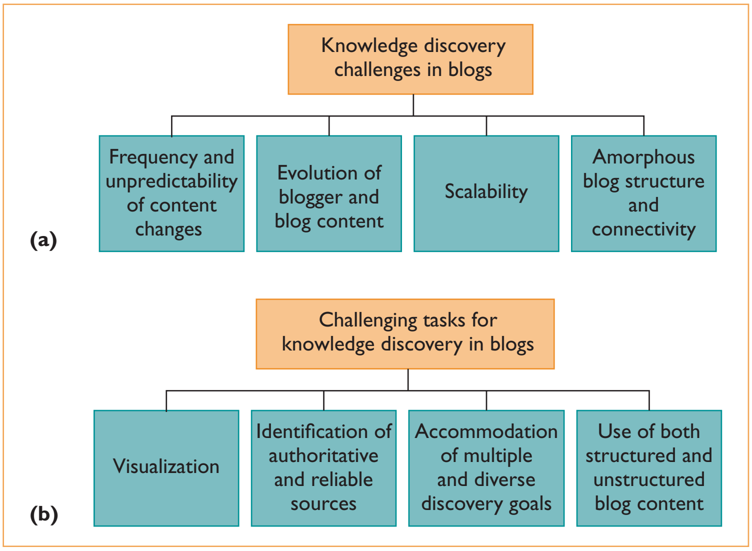
Knowledge discovery challenges in blogs
Frequency and unpredictability of content changes
Evolution of blogger and blog content
Scalability
Amorphous blog structure and connectivity
(a)
Challenging tasks for knowledge discovery in blogs
Visualization
Identification of authoritative and reliable sources
Accommodation of multiple and diverse discovery goals
Use of both structured and unstructured blog content
(b)

Figure 3. Knowledge discovery (a) challenges and (b) tasks in blogs. Some of these challenges are unique to blogs, whereas others are broadly applicable to discovery in many domains.

are more broadly applicable to discovery in any problem domain.

Content Change Frequency and Unpredictability

Predicting when and how frequently new blog content will change can be difficult. Content might include new entries from the blogger or comments responding to a post. Blog entries sometimes appear in bursts, depending on the author's enthusiasm for the topic and availability to write. On *The New York Times'* Bits blog, Saul Hansell posted an entry entitled "The Problem with Cable Is Television" on 1 May 2009 at 12:40 p.m. (see http://bits.blogs.nytimes.com/2009/05/01/the-problem-with-cable-is-television/?apage=3#comments). The blog received its first comment at 1:30 p.m., 20 more by 2:30 p.m., and seven more by 3:30 p.m.; it had 46 comments by the day's end. The blog had its last and 85th comment on 14 May at 4:17 p.m. (last checked on 14 May). Thus, within approximately 12 hours of posting, the blog received half the total comments it received over two weeks, illustrating that the interest in a single post can vanish quickly over time. Knowledge discovery algorithms must be flexible enough to accommodate blogs that change every minute compared to those that change only once a week. For blogs that change frequently, discovery algorithms must be dynamic so that they can update their existing content with new results without diminishing the data's meaning or importance.

Blogger and Blog Content Evolution

Bloggers' personal tastes, beliefs, and interests can change over time, influencing the type of content they post and whether the blog remains active, inactive (or dormant), or dies. For instance, although Opher Etzion's blog on "Event Processing Thinking" (www.epthinking.blogspot.com) mostly contains posts on event processing, Etzion also occasionally blogs about diverse topics that shape his day-to-day experience, such as "Friday the 13th," and insights from a vacation in Finland. Data acquisition techniques for discovery algorithms that target specific blogs on specific content must be prepared for such changes, and discovery algorithms need to be sensitive to the blog data's temporal characteristics.[6]

Scalability

BlogScope, a system for analyzing temporally ordered streaming text online, currently tracks more than 36.88 million blogs with 837.39 million posts in the blogosphere. On average, the crawler for this system fetches 14,000 new documents every hour.[7] So, knowledge discovery techniques must be computationally efficient to process blog data at this magnitude.

Amorphous Blog Structure and Connectivity

The connectivity between blogs that arises due to cross-postings and cross-linking from different bloggers in a community is dynamic and unpredictable. By addressing the amorphousness of blog structure and connectivity, knowledge discovery algorithms can achieve community detection. Algorithm designers can then combine these issues with different discovery goals, such as trend analysis and hot-topic discovery, to provide vivid and interesting results. Blogs' community structure and connectedness (in addition to number) also changes rapidly.[7,8] Since its inception, the blog "Sramana Mitra on Strategy" (www.sramanamitra.com), for instance, has been syndicated by Seeking Alpha, Yahoo! Finance, ReadWriteWeb, Cadwire, Emergic, GigaOm, TheStreet.com, and many other Web sites, leading to a rapid increase in the connectivity and community structure of Mitra's blog. Data acquisition techniques for discovery algorithms can leverage such connectivity to discover other relevant blogs to mine, and discovery algorithms can incorporate community structure with knowledge discovery to find interconnecting relationships and patterns.

Knowledge Discovery Tasks

In addition to addressing the challenges inherent to blogs, relevant literature indicates that knowledge discovery techniques must accomplish particularly challenging *tasks* to accurately and effectively discover knowledge. Let's look at some key tasks (see Figure 3b).

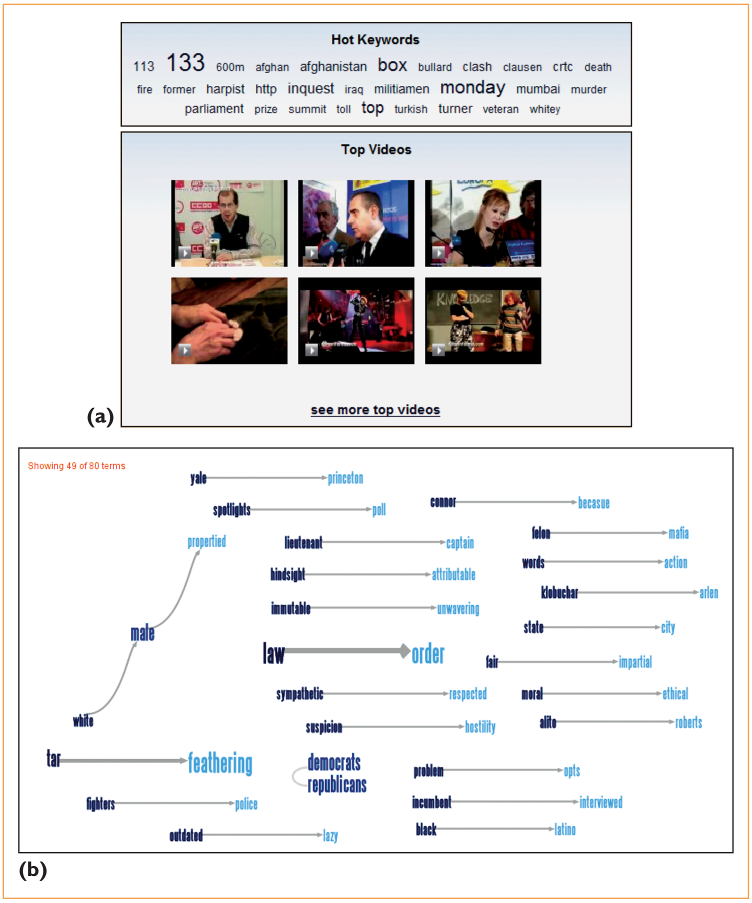

Figure 4. Visualization techniques for information mined from blogs. We can see (a) BlogScope's hot keyword cloud and top videos for 9 December 2009 as well as (b) a ManyEyes phrase net for a blog entry about US Supreme Court nominee Sonia Sotomayor.

Visualization

One general knowledge discovery challenge is quickly summarizing discovered results in a concise, easy-to-understand, intuitive format. Knowledge discovery algorithms' output doesn't necessarily meet these criteria and could be very large and convoluted. So, researchers developed visualization techniques to better convey these algorithms' results. BlogScope (www.blogscope.net) assists users in discovering interesting information from millions of blogs via visualization techniques such as popularity curves, information burst identification, related terms, and geographical search. Engineers can then implement them on top of knowledge discovery algorithms to mine blog data. Figure 4a shows a screenshot of BlogScope's hot keyword cloud and top videos for 9 December 2009. ManyEyes (manyeyes.alphaworks.ibm.com) is another site that enables data visualization, including blog data. Figure 4b shows a *phrase net* from ManyEyes, which displays a network of related words and phrases, about a blog entry on US Supreme Court nominee Sonia Sotomayor.

Identifying Authoritative and Reliable Sources

Measuring blog participants' authority is becoming increasingly important, particularly when professional application areas are involved.[9] Authoritative bloggers might be experts on a particular subject or persuasive forces that influence others. Weighing blog posts with respect to the author's authority is a significant challenge for knowledge discovery algorithms. Another aspect to this problem is verifying the authenticity of data shared on blogs. Information frequently flows from blog to blog, making it difficult to track the information's origin, provenance, or credibility.[10] Discovery algorithms that incorporate techniques such as pattern mining to discover information

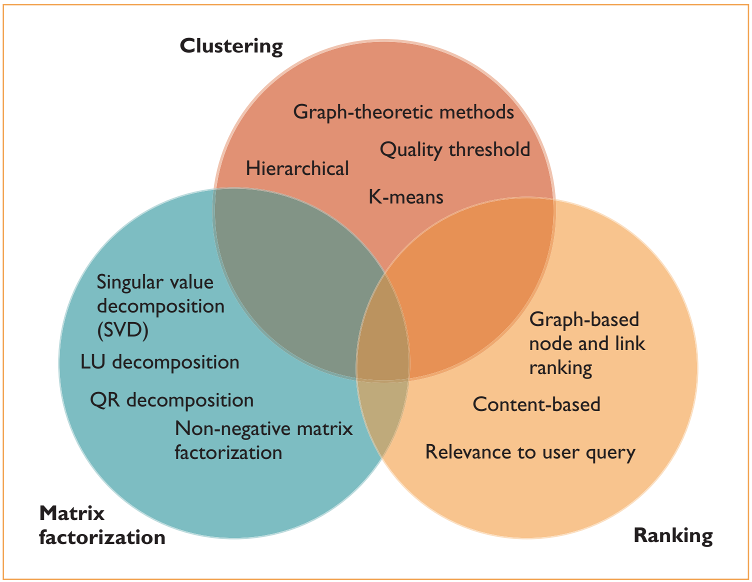

Figure 5. Strategies for knowledge discovery in blogs. We can classify these strategies in to three main areas, and many are "hybrid," or combinations of techniques in each of these three domains.

flows between blogs can use such patterns to give users more reliable data.

Accommodating Multiple, Diverse Goals

Given the variety of uses for knowledge discovered from blogs, a particular discovery technique should be capable of discovering knowledge in multiple facets. For instance, if we can use the same technique for topic detection and trend analysis, this technique would be more widely applicable to a large user base. Clustering-based knowledge discovery techniques, for example, are effective at detecting hot topics and extracting stories,[8] extracting trends by creating persistent, time-resistant keyword clusters,[7] discovering blog communities by clustering blogs by topic,[2] and clustering blogs with collective wisdom.[11]

Using Structured and Unstructured Blog Content

Tagging is a popular phenomenon in blogs. Users tend to employ descriptive tags to annotate the blog content they're interested in. In addition to mining unstructured text in blogs, discovery algorithms could leverage structured information such as tags to discover social interests, for instance.[12]

Knowledge Discovery Strategies

We examined three of the most popular strategies for discovering knowledge in blogs (see Figure 5): clustering, matrix factorization, and ranking.

Clustering

Clustering is one of the most common machine learning techniques researchers have applied to knowledge discovery in blogs.[7,8,11] Clustering assigns a set of observations to subsets, referred to as clusters, such that observations in the same cluster are similar according to prespecified criteria. Data clustering algorithms can be hierarchical or partitional. Hierarchical algorithms find successive clusters using previously established ones. Such algorithms either start with each element as a separate cluster and merge them into successively larger clusters or take the whole set and divide it into successively smaller clusters. Partitional algorithms typically determine all clusters at once but can also act as divisive algorithms in hierarchical clustering. K-means clustering and quality threshold (QT) clustering are both partitional clustering algorithms. Clustering algorithms might require users to specify the number of clusters the algorithm should produce in the input data set. An important step in clustering is to select a distance measure, which will determine how the algorithm calculates two elements' similarity. This influences cluster shape because some elements might be close to one another according to one distance and farther away according to another. Once the algorithm has established clusters, another important step in clustering is to determine the membership of newly arrived data into preexisting clusters.

In the blogosphere, researchers have used clustering to create keyword clusters in specific temporal intervals and investigate algorithms to identify keyword clusters that persist over time.[7] Each cluster identifies a discussion on a specific event or topic. Nitin Agarwal and his colleagues propose a method for clustering blogs that leverages bloggers' collective wisdom by creating a label relationship graph that assumes the bloggers themselves label their blogs; this method clusters labels in order to cluster similar blog sites.[11]

Arun Qamra and his colleagues' work demonstrates clustering applied to discovery in the blogosphere.[8] They consider blogs' content and community structure and apply a probabilistic two-stage clustering model. First, they divide blog entries into community-topic clusters, such that the entries in one cluster are from a group of bloggers who are likely to have a similar interest in the given topics and who might discuss these issues online. This method determines the number of clusters based on the data — users don't need to specify them. In the second stage, the algorithm further divides clusters into stories using time stamps in addition to entry content, employing a probabilistic model to perform clustering.

Recently, coclustering algorithms have emerged that cluster along both rows and columns. For example, a user might be interested in finding similar documents and their interplay with word clusters. When dealing with sparse and high-dimensional data, coclustering is scalable to large matrices.[13]

Matrix Factorization

Matrix factorization is another common tool researchers use for knowledge discovery in blogs. This technique involves decomposing a matrix into some canonical form. Many different matrix decompositions exist, such as LU decomposition (which writes a matrix as the product of a *lower* triangular matrix and an *upper* triangular matrix), singular value decomposition (SVD), Cholesky decomposition, and QR decomposition, and each is useful for particular problems. For instance, SVD is applicable to an m by n matrix A:

$$A = UDV^T,$$

where D is a nonnegative diagonal matrix, U and V are unitary matrices, and V^T denotes the conjugate transpose of V (or simply the transpose, if V contains only real numbers). D's diagonal elements are considered A's singular values. We can view SVD as a pattern-recognition technique that we can use to discover the answer to various objectives, such as whether the same person has written multiple documents or whether documents discuss the same topic. To apply SVD, users must first arrange documents in matrix form, with terms along the columns and documents along the rows. The next step involves expressing the term-document matrix in the form UDV^T.

Yun Chi and his colleagues apply SVD to analyze the eigen trend — that is, the temporal trend in a group of blogs with common interests[14] — and extract multiple eigen trends that reflect the blogosphere's structural changes over time. Yun Chen and his colleagues use SVD to distinguish the semantic similarity between blogs and improve business blog classification by topic areas.[15] Finally, Ka Cheung Sia and his colleagues use nonnegative matrix factorization methods to achieve personalized aggregations of independent opinions expressed in blogs.[16]

Let's look more closely at the matrix factorization method from Chi and his colleagues. They assume that the blogosphere consists of m blogs and that a keyword k's popularity score among those blogs within a time window j is given as a popularity vector $\vec{x} = \left(x_{1j}, \ldots, x_{mj}\right)^T$. The authors view this vector through n consecutive time windows and stack it into an m by n matrix $X = \left(\vec{x}_1, \ldots, \vec{x}_n\right)$; thus, x_{ij} is the number of entries in blog i that contain keyword k at time j. The authors' goal is to find a trend vector $\vec{t} = \left(t_1, \ldots, x_n\right)^T$ that represents the temporal aspect of the popularity score X, where t_j represents the overall popularity score at time j. As part of their solution, they represent the observed data X with a pair of vectors: a trend vector $\vec{t}$ that represents the overall trend over time and an authority vector $\vec{a}$ that represents the bloggers' contributions to individual trends. They show that by using SVD, they can obtain a $\vec{t}$ and $\vec{a}$ that best approximate the observed data X.

Finding an exact SVD isn't efficient with regard to time. The time complexity can be on the order of $O(\min(n^2m, nm^2))$ for an m by n matrix A. Furthermore, the technique could be costly with regard to space because U is often dense even if A is sparse. Finding the appropriate interpretation for the singular vectors isn't always easy in practice. Finally, updating SVD results can be difficult if the graph evolves over time.[14] Researchers have developed techniques, such as the Colibri methods, to address SVD's challenges.[17] However, their effectiveness for knowledge discovery in blogs is still undetermined.

Further Reading

Several blog data sets exist for individuals wishing to conduct research on blog discovery:

- Spam-blog data set: http://ebiquity.umbc.edu/resource/html/id/212/Splog-Blog-Dataset
- BLOGS06 data set created and distributed by the University of Glasgow: http://ir.dcs.gla.ac.uk/test_collections/blog06info.html
- Weblog Data Collection provided by BuzzMetrics: www.icwsm.org/data.html

Table 1. How discovery algorithms address challenges in the blogosphere.

	Content change frequency and unpredictability	Blogger and blog content evolution	Amorphous blog structure and connectivity
Clustering	Creates time-stable clusters (Bansal[7])	Uses community and time-based story extraction (Qamra[8])	Leverages tagging in blogs (Li[12])
	Uses community and time-based story extraction (Qamra[8])	Uses time- and content-sensitive dynamic clustering (Agarwal[11])	Leverages blog labels (Agarwal[11])
	Uses time- and content-sensitive dynamic clustering (Agarwal[11])		
Matrix factorization	Uses eigen trend to capture the temporal trend in a group of blogs (Chi[14])	Allows personalized aggregations so user receives only relevant opinions (Sia[16])	Takes into account dynamically changing graph structure derived from linkage between blogs (Chi[14])
Ranking	Addresses blogs represented as dynamically changing graphs (Kumar[20])	Addresses dynamic changes in blog content over time (Kumar[20])	Takes into account links between blog entries (Song,[5] Kumar[20])

Ranking

A user might choose to drive knowledge discovery in blogs by specifying criteria for ranking retrieved blog entries. Ranking blogs is quite similar to ranking Web pages. PageRank and HyperText Induced Topic Selection (HITS) are two popular techniques for Web page ranking that exploit the link structure between such pages. These algorithms focus on a directed graph setting that describes resources via nodes and hyperlinks. Link-popularity-based algorithms, however, might not work well for blog mining because blog pages aren't well linked and bloggers might try to exploit such a system to boost their rank. This observation has led to several new blog ranking techniques that exploit both links and content for ranking.[9,18] A user might want to retrieve blog entries ranked according to opinions on a certain topic expressed in those entries.[19] Similarly, the influenceRank algorithm enables blog ranking according to how influential each one is compared to others as well as the originality of the opinions the blogs specify.[5] The system from Xiaodan Song and her colleagues summarizes opinions in the blogosphere by applying the influenceRank algorithm and selects more influential blogs with novel information compared to PageRank.[5] In the context of blogs, researchers have also explored link-popularity-based graph algorithms that use techniques such as random walks and random sampling combined with ranking for both static and dynamically changing graphs.[20]

As an example of ranking applied toward blog discovery, let's look at the method Ahmed Hassan and his colleagues employ.[18] They define the importance score $S(p)$ of a blog p recursively in terms of its neighbors' scores as follows:

$$S(p) = \sum_{q \in adj(p)} \frac{S(q).sim(p,q)}{\deg(q)},$$

where $\deg(q)$ is the degree of node q, $adj(p)$ is the set of all nodes adjacent to p in the network, and $sim(p, q)$ is the similarity between blogs p and q. The authors estimate this similarity using the cosine similarity between the posts' term frequency vector representation and inverse document frequency vector representation. We can compute cosine similarity as the cosine of the angle between the term frequency and inverse document frequency vectors.

Knowledge Discovery Strategy Comparison

Table 1 compares specific clustering, matrix factorization, and ranking algorithms according to how they address the knowledge discovery challenges we identified previously. As the table shows, different authors have applied all three techniques (denoted by the first author's last name) to address the challenges inherent to blogs. We also identified several algorithms that combine one or more of these techniques with classification, a supervised machine learning procedure. We omit scalability from the list of challenges in this comparison because we didn't find appropriate information to warrant a fair comparison between the three techniques' scalability when applied to blogs.

Table 2. How discovery algorithms accomplish challenging tasks in the blogosphere.

	Accommodating multiple and diverse discovery goals	Identifying authoritative and reliable sources	Visualization	Use of structured and unstructured blog content
Clustering	Uses hot-topic detection (Bansal,[7] Qamra[8])	Uses blog labels to distinguish between relevant sources for clustering (Agarwal[11])	Shows simple cluster graphs of keywords (Bansal,[7] Agarwal[11])	Leverages tagging in blogs (Li[12])
	Uses trend analysis (Bansal[7])	Factors community and time to identify relevant sources for clustering (Qamra[8])		Leverages blog labels (Agarwal[11])
	Uses collective wisdom (Agarwal[11])			
Matrix factorization	Uses collective wisdom (Sia[16])			
	Uses trend analysis (Chi[14])	Lets user personalize the choice of information sources (Sia[16])	Easily shows top authorities for the first scalar eigen trend (Chi[14])	Takes into account dynamically changing graph structure derived from linkage between blogs (Chi[14])
Ranking	Uses topic detection (Song[8])	Identifies influential bloggers (Agarwal[9])	Permits graph visualizations of influential bloggers in their networks (Song[5])	Takes into account the links between blog entries (Song,[5] Kumar[20])
	Uses opinion retrieval (Zhang,[19] Song[5])	Identifies influential blogs (Song,[5] Kumar[20])	Shows effective summaries of influential bloggers' blogging behavior (Agarwal)[9]	
	Uses influential blog retrieval (Song[5])	Retrieves blog entries ranked according to opinions expressed in them (Zhang,[19] Song[5])		

Table 2 compares algorithms based on the three knowledge discovery strategies according to how they accomplish the key tasks we identified previously. Table 2 indicates that all these knowledge discovery techniques have gone toward accomplishing different blog discovery goals, identifying authoritative and reliable blogs or blog posts, visualizing mined blog data, and using both structured (such as tags) and unstructured information (such as random links and misspelled or non-English text) for blog mining.

Our research indicates that no common evaluation techniques, criteria, or benchmarks exist to quantitatively compare one knowledge discovery technique with another. Comparing different blog discovery algorithms' scalability and robustness is particularly difficult without standard benchmarking tools. Furthermore, humans evaluate mining algorithms' effectiveness and accuracy as regards their ability to accomplish discovery goals, such as hot-topic detection or popular-story extraction — although relying on humans for validation is certainly important, we need objective measures to verify these algorithms' accuracy. Such open problems make it difficult to determine which single algorithm (if any) is the best knowledge discovery algorithm for blogs.

Despite the extensive recent activity on knowledge discovery in blogs, more research needs to occur before we have algorithms that are scalable, accurate, and robust, and provide interpretable results. Blog data is no longer just numerical or discrete. Because bloggers now express themselves through videos, photos, and tweets in addition to text-based posts, algorithms must combine mining capabilities for different social media to effectively mine the blogosphere.

References

1. N. Agarwal and H. Liu, “Blogosphere: Research Issues, Tools, and Applications,” *ACM Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD) Explorations*, vol. 10, no. 1, 2008, pp. 18–31.
2. Y-R. Lin et al., “Blog Community Discovery and Evolution Based on Mutual Awareness Expansion,” *Proc. Conf. Web Intelligence*, ACM Press, 2007, pp. 48–56.
3. K.C. Sia et al., “Efficient Computation of Personal Aggregate Queries on Blogs,” *Proc. Knowledge Discovery and Data Mining Conf.*, ACM Press, 2008, pp. 632–640.
4. D. Gruhl et al., “The Predictive Power of Online Chatter,” *Proc. Int’l Conf. Knowledge Discovery and Data Mining*, ACM Press, 2005, pp. 78–87.
5. X. Song et al., “Identifying Opinion Leaders in the Blogosphere,” *Proc. Conf. Information and Knowledge Management*, ACM Press, 2007, pp. 971–974.
6. Y. Chi et al., “Structural and Temporal Analysis of the Blogosphere through Community Factorization,” *Proc. Knowledge Discovery and Data Mining Conf.*, ACM Press, 2007, pp. 163–172.
7. N. Bansal et al., “Seeking Stable Clusters in the Blogosphere,” *Proc. Very Large Databases Conf.*, ACM Press, 2007, pp. 806–817.
8. A. Qamra, B. Tseng, and E.Y. Chang, “Mining Blog Stories Using Community-Based and Temporal Clustering,” *Proc. Conf. Information and Knowledge Management*, ACM Press, 2006, pp. 58–67.
9. N. Agarwal et al., “Identifying the Influential Bloggers in a Community,” *Proc. Web Search and Data Mining Conf.*, ACM Press, 2008, pp. 207–217.
10. A. Stewart et al., “Discovering Information Diffusion Paths from Blogosphere for Online Advertising,” *Proc. Workshop Data Mining and Audience Intelligence for Advertising in Conjunction with Knowledge Discovery and Data Mining*, ACM Press, 2007, pp. 46–54.
11. N. Agarwal et al., “Clustering Blogs with Collective Wisdom,” *Proc. Int’l Conf. Web Engineering*, IEEE CS Press, 2008, pp. 14–18.
12. X. Li, L. Guo, and Y. Zhao, “Tag-Based Social Interest Discovery,” *Proc. World Wide Web Conf.*, ACM Press, 2008, pp. 675–684.
13. I.S. Dhillon, S. Mallela, and D.S. Modha, “Information Theoretic Coclustering,” *Proc. Int’l Conf. Knowledge Discovery and Data Mining*, ACM Press, 2003, pp. 89–98.
14. Y. Chi, B.L. Tseng, and J. Tatemura, “Eigen-Trend: Trend Analysis in the Blogosphere Based on Singular Value Decompositions,” *Proc. Conf. Information and Knowledge Management*, ACM Press, 2006, pp. 68–77.
15. Y. Chen, F.S. Tsai, and K.L. Chan, “Blog Search and Mining in the Business Domain,” *Proc. Workshop on Domain Driven Data Mining in Conjunction with Knowledge Discovery and Data Mining*, ACM Press, 2007, pp. 55–60.
16. K.C. Sia et al., “Efficient Computation of Personal Aggregate Queries on Blogs,” *Proc. Knowledge Discovery and Data Mining*, ACM Press, 2008, pp. 632–640.
17. H. Tong et al., “Colibri: Fast Mining of Large Static and Dynamic Graphs,” *Proc. Conf. Knowledge Discovery and Data Mining*, ACM Press, 2008, pp. 686–694.
18. A. Hassan et al., “Content-Based Recommendation and Summarization in the Blogosphere,” *Proc. Conf. Weblogs and Social Media*, ACM Press, 2009; www.aaai.org/ocs/index.php/ICWSM/09/paper/view/203.
19. W. Zhang, C. Yu, and W. Meng, “Opinion Retrieval from Blogs,” *Proc. Conf. Information and Knowledge Management*, ACM Press, 2007, pp. 831–840.
20. R. Kumar et al., “Efficient Discovery of Authoritative Resources,” *Proc. Int’l Conf. Data Engineering*, ACM Press, 2008, pp. 1495–1497.

Geetika T. Lakshmanan is a research staff member at the IBM T.J. Watson Research Center. Her research interests include interdisciplinary approaches toward solving problems in data monitoring, management, mining, and provenance, and business processes. She’s interested in applying the results of this research on developing middleware for computer systems. Lakshmanan has a PhD in computer science from Harvard University. She’s a member of the ACM and the Sigma Xi Scientific Research Society. Contact her at gtlakshm@us.ibm.com.

Martin A. Oberhofer is a software engineer at the IBM Software Group in Germany. His areas of expertise include master data management, enterprise information integration, database technologies, Java development, and IT system integration. Oberhofer has an MS in mathematics from the University of Constance, Germany. He’s a member of the Gesellschaft for Informatik, an association for researchers and professionals in computer science in Germany. Contact him at martino@de.ibm.com.