Toward Personalized Query Expansion

Marin Bertier
INSA de Rennes, France

Rachid Guerraoui
EPFL, Switzerland

Anne-Marie Kermarrec
INRIA Rennes, France

Vincent Leroy
INSA de Rennes, France

ABSTRACT

Social networking and tagging have taken off at an unexpected scale and speed, opening huge opportunities to enhance the user search experience. We present GOSSPLE [1], a new, user-centric, approach to improve the exploration of the Internet. Underlying GOSSPLE lies the intuition that while social networks provides news from your old buddies, you can learn a lot more from people you don't know, but with whom you share many (tagging) interests. More specifically, considering a collaborative tagging system with active taggers annotating content, GOSSPLE expands the search query, of any user u, with tags that are considered "close" enough with respect to users that are "close" to u.

GOSSPLE users create their own network of social acquaintances in a gossip-based manner, by dynamically computing the estimation of a distance between taggers, based on cosine similarity between tags and items. These connections are used to feed a $TagMap$: our central abstraction that captures the personalised relationships between tags. The $TagMap$ is then used by GOSSPLE to meaningfully expand queries leveraging the personalised network. This is achieved through the $TagRank$ algorithm, an adaptation of the celebrated pagerank algorithm, which automatically determines which tags best expand a list of tags in a given query.

GOSSPLE has no central authority: every user stores its own items and its tagging behaviour is stored only by its neighbours. The resulting networks are live, dynamic and do not require any underlying structure. We report on our evaluation of GOSSPLE with CiteUlike traces, involving 33,834 users. In short, we show that, with little information stored at every peer, GOSSPLE enables to retrieve items that cannot be retrieved with state of the art search systems (completeness).

[1]This work is supported by the ERC Starting Grant 204742

SNS'09 , March 31, 2009 Nuremberg, Germany

1. MOTIVATION

The Web revolution.

The Web has turned from a read-only infrastructure with passive participants into a read-write platform with active players. The content of the Web is no longer generated only by experts but pretty much by everyone (YouTube, Flickr, Last.fm, Delicious, etc). Like any popular revolution, this goes through democratising the language: instead of subject indexing with a controlled vocabulary, freely chosen keywords are used to *tag* billions of items, e.g. URL (Delicious). The user-generated taxonomy is called *folksonomy* (folk + taxonomy) and is used to label and share user-generated content (*e.g* photographs), or to collaboratively label existing content (*e.g* Web sites, books, or blog entries). Part of the appeal of a folksonomy is its inherent subversiveness: folksonomies can be seen as a rejection of the traditional search engine status quo in favour of tools that are created by the community. In theory, precisely because folksonomies develop Internet-mediated personalised environments, one could dynamically discover the tag sets of another user who tends to interpret and tag content in a similar manner. The result could be a rewarding gain in the user's capacity to find related content, a practice known as "pivot browsing".

Personalisation goes with decentralisation.

While intriguing, this Web revolution is still in a preliminary stage, and this is at least for two main reasons. First, most collaborative tagging networks are controlled by centralised systems. So as much as users are first class citizens of the system and are free to introduce new items and tag them in their proper language, they are not free to choose where these items are stored and, more importantly, cannot usually freely decide to remove items and tags. In the long run, this might dissuade users from generating new content and expressing their tagging behaviour in an explicit manner. Furthermore, and no matter how powerful servers

can be, centralised solutions do not promote the maintenance of personalised relations between users, which might reveal crucial in the search as we will discuss below. These relations grow exponentially with the size of the system and the success of social tagging might simply kill the underlying centralised infrastructure.

Second, while the success of collaborative networks is clearly related to the freedom left to the users, this is also a drawback. The facts that such systems are not governed by specific structures (as opposed to ontologies for instances), and that tags are informally defined, and continually changing, mean there is no insurance that the tagging behaviour of a user on some content makes any sense for another one, nor does it prevent junk tagging and synonyms, which introduce significant noise in the process. The reactivity offered by fully decentralised solutions may solve this issue.

Beyond friends: discovering similar users.

We believe that the salvation can only come from pushing the revolution further. Basically, we argue for a fully user centric approach where every participant stores and controls not only her own items and tagging behaviour, but also her perspective on what portion of the network is relevant to her own search. Every user query is then expanded with tags that are considered appropriate with respect to the personalised network of that user.

To illustrate the motivation behind our approach, consider the following (real) example. After living for several years in the UK, Anne is back to Rennes in France and, to maintain her kids' skills in English, is looking for an English speaking student who would be willing to trade baby-sitting hours against accommodation. Given the high number of students in Rennes, there is no doubt that such an offer would be of interest for many English speaking students. Anne's Google request "English baby-sitter Rennes" does not give anything interesting for baby-sitter is immediately associated with child minders or local (French) baby-sitting companies. Her Facebook buddies in Rennes or in the UK cannot really help either for none has ever looked for an English speaking baby sitter in Rennes. Consider now Alice leaving in Bordeaux, after several years in the US, and who is looking for a similar deal with her kids. Alice is however lucky to discover that teaching assistants in primary school are a very good match as they have the same working hours as kids, they do have a salary but would enjoy leaving within a family. Now if Alice associates "english-speaking baby-sitter" with "teaching assistant" in her search request, she does indeed find very good candidates. Clearly, if Anne could reuse Alice's discovery, she would also find good candidates in Rennes. Nevertheless, Alice and Anne do not know each other nor do they live in the same area, nor even have similar jobs. Yet, their past history made clear their links through the fact that they both lived in English speaking countries and both have kids around the same age and do need baby-sitters. Should a system be able to make the connection between Alice and Anne, the association between tags "teaching assistant" and "baby-sitter" could be helpful. Therefore a mechanism that would expand Anne's query "english-speaking baby-sitter" to "assistant etranger" or teaching assistant" would render her request solvable by any search engine.

Contributions.

The observation we drew from this example which, as we pointed out, is inspired by a real scenario, is that, in contrast to old buddies that do not bring much to the search, unknown people who share similar interests can do the job. Expanding a user's query by identifying her connection with personalised acquaintances is not immediate for this requires, within a huge dynamic system, maintaining implicit connections and deriving complementary tags on the fly for every query. This is the challenge addressed by GOSSPLE. In short, GOSSPLE *automatically* infers *personalised* connections between users and provides them with *semantically* related tags as companions to their queries.

At the heart of GOSSPLE lies the *TagMap* abstraction through which we capture the personalised relationship between tags. Every peer locally stores its TagMap which is fed by a (discovered) personal network. This network is dynamically created in a gossip-based manner computing the estimation of a distance between taggers, based on tagging behaviour. Cosine similarities on tags and items are used to create each user's TagMap. A key feature of GOSSPLE is to expand queries in a meaningful way, leveraging the TagMap. To this end, we use an algorithm called *TagRank* for it is inspired from the pagerank algorithm, to extract the most relevant tags from the *TagMap* for a given request in order to expand the query.

We report on our evaluation of GOSSPLE with CiteULike traces, involving 33,834 users. In short, we show that, with little information stored at every peer, GOSSPLE enables to retrieve items that cannot be retrieved with state of the art search systems (completeness) without hampering accuracy (increasing the number of false positives).

2. GOSSPLE IN A NUTSHELL

System model.

We consider a system composed of a set of users U. Users may tag a set of items I with tags from the set of tags T, The information space (IS) is defined as a set of triplets $(u, i, t) \in U \times I \times T$ representing the relationships

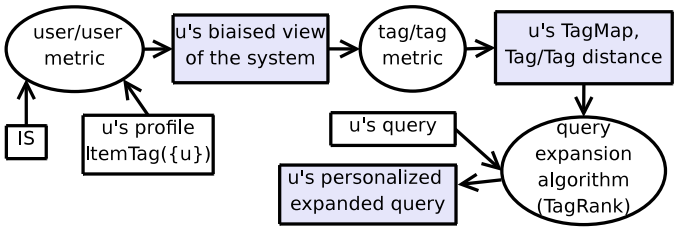

Figure 1: gossple overview

defined by users between items and tags.

The information space can be accessed by a set of functions defined as follow: $FunctionName(parameters)$ returns a set of $FunctionName$ for the fixed values of the parameters. For instance $Item(\{u_1\},\{t_1,t_2\})$ returns the sets of items tagged by u_1 with t_1 or t_2. Similarly $ItemTag(\{u_1\})$ returns the set of (i,t) such as $(u_1,i,t) \in IS$ (u_1 tagged i with t). This represents the profile of a user.

We consider that the users are connected through a connected network, typically through the use of random peer sampling service [1].

Overview of GOSSPLE.

Query expansion definition: The query expansion is a process that transforms a user query in order to improve the performance of the search engine. It involves transforming the query terms (correcting spelling and stemming words), adding new terms and weighting them. We will not consider correcting spelling and stemming since they usually rely on known local algorithms and dictionaries, they do not require the personal network knowledge. Query expansions adds terms to the query, increasing the number of results given to the user. The query expansion has to be precise enough to be able to the add relevant documents to the result set while keeping the number of irrelevant documents low. This is a trade off between recall and precision.

The goal of GOSSPLE is to discover users sharing similar interests (personalised network), and to gather their information in order to improve query expansion. Figure 1 presents an overview of GOSSPLE. The first step is to identify the relevant users to form the personal network. This is achieved by relying on a distance metric between user, using the cosine similarity between sets of items. The personal network, much smaller than the whole network, typically 20 neighbours are enough in a 33,834 user system as we show in the experiments, is used to build a each user's *TagMap*. The TagMap represents a personalised view of the relation between tags, as a distance between tags. The query expansion algorithm, called TagRank, exploits the user's TagMap to expand a given query.

In the sequel we describe how each user creates its *TagMap*, a matrix TM_u, capturing the relationships between tags where $TM_u[t_i,t_j]$ contains a value reflecting the relationship between tags t_i and t_j as seen by the user u. This is computed from the information of each user's personal network. We then present the *TagRank* algorithm, exploiting the TagMap to expand the query on a per query basis. Finally, we present the way each user discover the neighbours to form its personal network in a fully decentralised way through gossip protocols.

3. THE TAGMAP: A PERSONALISED VIEW OF THE RELATIONS BETWEEN TAGS

In this section, we first present the metric used to compute the distance between users based on their tagging behaviour, namely cosine similarity between items [2].

3.1 Rating the users: items cosine similarity

The TagMap capture the distance between tags, this is extracted automatically from the tagging behaviour. Detecting users sharing interest requires to be able to compute a distance between users. The most natural metric to consider is the overlap between the items they tag. However, this simple metric suffers from several drawbacks. Users that have a very high tagging activity, will exhibit a high overlap with any other user, while this does not reflect any specific interest. In addition, the proximity between users with a relevant metric not only should increase when interests are similar, but it should also decrease if many other interests are not shared.

Instead we use a well-known metric, used in data mining, namely the cosine similarity between items. This can be seen as a normalised overlap. Items are represented as vectors in a multidimensional space, the number of dimension being $|I|$.

More formally, the cosine between two vectors of items is defined as follows:

$\cos(\vec{v_1},\vec{v_2}) = \frac{\vec{v_1}\cdot\vec{v_2}}{\|\vec{v_1}\|\times\|\vec{v_2}\|}$

The item cosine is defined as follows:

$ItemCos(u_1,u_2) = \frac{|Item(\{u_1\})\bigcap Item(\{u_2\})|}{\sqrt{|Item(\{u_1\})|\times|Item(\{u_2\})|}}$

The score between two users increases when interests are shared and decreases when they are not.

3.2 Creating the TagMap

The distance between users is used by users to create their personal network, so that the information about tags, collected from users u_i are incorporated in the TagMap of user u_j depending on its distance to u_i. We consider that each user has a personalised network, we will come back on the discovery of such a personalised network in the next section. $Neighbours(u)$ is defined as the set of users in the personalised network of u.

[2]Note that there are many metrics that could be used, we chose this one for the purpose of comparison with related works.

Again there are many ways to use the information provided by the neighbour's profile to fill the TagMap depending whether the query expansion should rely on a dictionary of synonyms or a hierarchical relationship [2]. For space reasons, we focus on synonyms in this paper.

The information needed to fill the tag map is for each tag, the number of occurrences of the use of that tags per items, namely:

For all $t \in Tag(Neighbours(u)$, a vector V_t of dimension $|I|$ is maintained such that if $V_t[item_i] = x, x = |User(Neighbours_u, \{i\}, \{t\})|$, namely the number of times the item $item_i$ has been tagged with t by the neighbours of u. The TagMap is then filled as follows:

$TM[t_i, t_j] = cos(\vec{V_{ti}}, \vec{V_{tj}})$

4. THE TAGRANK ALGORITHM: PERSONALISED QUERY EXPANSION

The TagMap represents the personalised relationships between pairs of tags to be used to expand queries. A straightforward solution, used in [3], to exploit the TagMap directly, is to consider only tags close to the tags of the query. This is an issue for the items suffer from a high sparsity: as there is a very large number of items, relationships between tags are sometimes hidden and can be hardly discovered. Consider for example a query on t_1, the TagMap provides a link between t_1 and t_2 (based on a set of items). Consider now that t_2 and t_3 are also close in the same TagMap (based on a different set of items), this straightforward solution will never discover a link between t_1 and t_3.

By iterating on the set of added tags, more relevant tags could be added to the query. To this end, we designed an algorithm called TagRank, inspired from PageRank[4]. The TagMap is represented as a graph in which all the tags in the TagMap are vertices. They are connected by weighted edges so that $weight(t_i, t_j) = TagMap(t_i, t_j)$ and $weight(t_i, t_i) = 1$ [3]. In PageRank, a random surfer walks in a graph of Web pages. The importance of each page is the probability of the surfer to be on that page at any time. At each step of the walk, the surfer either follows a link on the page or moves to a page chosen uniformly at random on the whole graph. In TagRank, the transition probability from one tag to another depends on the edge weight:

$TransitionProbability(t_1, t_2) = \frac{TagMap[t_1, t_2]}{\sum_{t \in T} TagMap[t_1, t]}$

The original PageRank algorithm computes a score for each vertex, but that score only depends on the structure of the graph, not on a user query. Like in personalised versions of PageRank, we modify the set of vertices the surfer can move to at random and limit it to the tags of the query. Therefore, the score computed is biased by the query, the query tags being the ones that spread importance into the graph. Calculating exact TagRank scores in a big graph can be a long process. Since this process is repeated at each query, this might be an issue in the long run. Therefore, we use an algorithm from [5] in order to provide a more efficient approach. For each query, the computation is split in order to compute partial scores for each tag in the query. At the end, all the partial scores are added to get the TagRank score of each tag. This saves a lot of processing time. The partial scores are approximated through random walks.

$TagRank(query, TagMap)$ outputs the list of all the tags in the TagMap associated with a weight. Since each weight is a probability, they sum up to one. The expanded query consists in the original query, plus additional terms chosen by descending weight. The system can either use the top-k extra tags, or add enough tags to "capture" a given amount of the weight.

5. CREATING THE PERSONAL NETWORK

Our algorithm is based on profile proximity between users. As presented in the subsection 3.2, the TagMap of a user is created from the profile of users which belong to her personalised network. The aim of the personal network is so to connect a user with their k closest users according to the metric presented in the subsection 3.1. k represents the trade-off between the amount of available information and the personalisation degree of this information (in other words, its quality).

We assume that the users are connected through an unstructured overlay implemented by a peer sampling service [1]. Basically, each user is provided with a (changing) random sample of the network (a view of say 20 random users). This protocol ensures that the network is connected and that new relevant users may be discovered when maintaining the personal network.

The creation of the personal work is achieved through a clustering gossip protocol. To this end each user maintains a view of k neighbours forming its personalised network. Starting from a random sample (typically provided by the underlying peer sampling service), this network is refined as follow. Periodically, a user contacts another user from her neighbours to exchange neighbours. When a user receive new neighbours upon a gossip interaction, it keeps from its own neighbours and the discovered one the k closest according to the metric defined in Section 3.1. This process is iterated and converges in a few cycles [6]. The TagMap of each user is then built from the profile of those k users, forming the personal network,

In order to reduce the message size, users exchange a Bloom filter representing a hash of their items vectors instead of the whole profile. The Bloom filter provides a reasonably good approximation of the user profile that

[3]This is directly infered from the metric based on cosine of vectors of items.

can be used to compute the cosine similarity with a small error margin. If the value of the cosine between the user's vector and the one infered from the Bloom filter, the users are considered closes enough and the entire profile is then exchanged. Otherwise, there is no further exchange. This avoid the transfers of useless entire profiles.

6. EVALUATION

In this section, we present preliminary experimental results. We run experiments using the CiteULike dataset of the 2008-10-09. $|U| = 33,834$, $|I| = 1,134,167$, $|T| = 237,450$, $|IS| = 4,064,310$. We build a profile for each user $u \in U$.

Workload.

To evaluate our algorithm, we generate queries to expand. After the query expansion, we launch the query to build a result set. The result set contain the items which match the query's tags. We generate a query for each item $i \in Item(\{u\})$ such as $User(\{i\}) > 1$ (an item has to be tagged by at least 2 users). For an item i, we choose a user u and we use the tags used by u on the item i to fill the query. As u will launch the query, we delete from the Information Space, the information used by the query generation ($IS - (u, i, t)$, $t \in Tag(\{u\}, \{i\})$).

The query succeeds when i is in the result set. The query goes through the query expansion process and we modify the number of tags added to the query in order to evaluate the impact on the recall, which in that case is the proportion of items found using the tags that were assigned to them by a given user.

Settings.

To evaluate our approach we run the following experiments on the same trace.

Global TagMap, simple query expansion: a global TagMap is built based on the same metric, namely cosine of item vectors. The distance between tags is therefore not personalised as it takes into account the information of all users. The query expansion is the simple one considered before, used in [3] considering only the tags related to the query tags. This is typically representative of a centralised approach, where personalised TagMap are too space intensive to maintain.

Personalised TagMap, simple query expansion: the TagMap is personalised, based on the profile of the ($k = 20$) closest neighbours. The simple query expansion mechanism is used here. The goal is to evaluate the impact of the TagMap personalisation.

Personalised TagMap, TagRank based query expansion: the TagMap is personalised, based on the profile of the ($k = 20$) closest neighbours. TagRank is used to expand the queries. This enables to evaluate the benefit of TagRank over a simple query expansion mechanism.

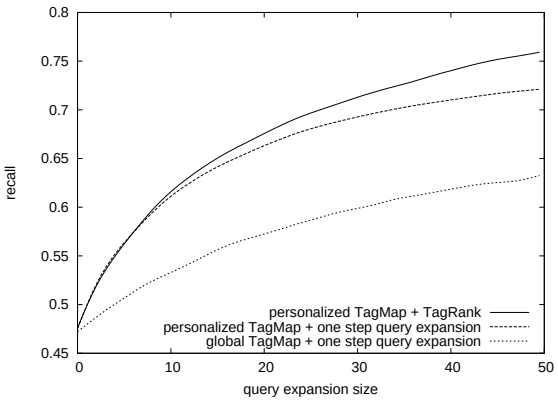

Figure 2: recall performance evaluation

Figure 2 shows the results of our simulations. In all cases, a query expansion size of 0 gives a recall of 47%. That means that in 47% of cases, when the item has been tagged by more than 2 users, at least one other user has used one tag in common. In all the other cases, the system has to rely on the query expansion process to add relevant tags to the query and improve the recall rate.

We observe that the personalised TagMap performs a lot better than the global TagMap, with on average 8% more recall. This shows first that the personal network is effective to personalise in a meaningful way the TagMap and generate a substantially more accurate query expansion. Second, it shows that only a small portion of the network is required to personalise in an effective manner the TagMap. The TagMap contains much less information, but since this information is centred on the user, the tags added through the query expansion are more relevant.

Finally, we observe that TagRank also contributes to improving the quality of the results, especially when it comes to producing a long query expansion. The recall is improved by up to 4% with a query expansion size of 50. This experiment demonstrates the limits of the one step distance when using the TagMap. The sparsity of the information in folksonomies limits the number of related tags that can be found. Since TagRank distributes weight in the whole graph, it can find tags that seem not related to the query but are still relevant.

7. RELATED WORK & CONCLUDING REMARKS

Collaborative social tagging schemes have received a growing attention, they provide a huge potential for discovering new information through implicit connections. In this paper, we presented the query expansion feature of GOSSPLE, a user centric system to discover and maintain such acquaintances. The GOSSPLE query expansion mechanism improves the completeness of the

search queries over state of the art alternatives without hampering the search accuracy. This is achieved with little information maintained at each peer, in the form of the TagMap. Interestingly, each peer discovers its personal network, locally stores the relevant, to itself, relationships between tags, and its tagging behaviour is only recorded by its personal neighbours. The TagMap is exploited by an original TagRank algorithm to expand in an effective way queries.

Recently, several centralised systems have addressed the personalisation of search in the context of folksonomies. These approaches mostly focus on top-k processing. In [7], the investigate network-aware top-k processing. They show that full personalisation which would result in maintaining data structures (typically inverted lists) on a per user basis are too space intensive. Instead the proposed algorithms rely on maintaining such data structure per cluster of socially related users and adapt the traditional centralised top-k algorithms to that setting.

In [3], a centralised system proposing both query expansion and top k processing also relies on tags association. Yet, the tag association is not personalised, nor the system is decentralised. The personalisation is addressed only in the top-k processing. One of the main reasons is that a personalised association between tags is too space intensive in a centralised system. Our experiments showed the benefit of the GOSSPLE personalised query expansion over this approach.

In [8], several types of social decentralised routing strategies are considered. Although they do not deal with query expansion, they confirm our intuition that social explicit connexions ala Facebook are useless for many requests. Instead they show that semantic routing, contacting neighours for given request are dependent or the content of the request, or spiritual routing, contacting neighbours having behavioural affinity provide the best results.

In [9], the authors explore different ways of providing personalised query expansion. They show that adding information extracted from the user's profile can help increasing the rank of a relevant document in the result list. Their approach is based on using the user's profile only, while our algorithms take advantage of the knowledge of the other users in the system. Therefore, our approach is more relevant for discovering new tags and increasing the recall of the requests.

Many centralised search engines provide non personalised query expansion. They add to the query synonyms and related concepts, found in a taxonomy in order to improve the quality of the results. They rely on hand-generated information like Yahoo! Directory [4], Wordnet [5] or the Open Directory Project [6]. The main difference with our system is that this data is neutral and objective, while our system aims at a subjective, user-related query expansion. Furthermore, our system is able to directly extract the knowledge from the information space while those information sources need to be maintained by users. Although we limited our approach to adding synonyms in this paper, GOSSPLE can determine different kind of relations between tags and use the same approach and is also able to build a biased taxonomy that reflects the interests of the user.

We believe that the way to personalise Internet search, in a world where users are free to express their opinion and interests goes with a fully decentralised system. To the best of our knowledge, we are the first to propose a personalised query expansion in a fully decentralised manner. We foresee many perspectives to that work, such as leveraging the TagMap for recommendation systems for instance and addressing dynamic networks.

[4]http://dir.yahoo.com/

[5]http://wordnet.princeton.edu/

[6]http://www.dmoz.org/

8. ACKNOWLEDGEMENT

We warmly thank Vivien Quéma for his help at early stages of the work.

9. REFERENCES

[1] M. Jelasity, R. Guerraoui, A.-M. Kermarrec, and M. van Steen. The peer sampling service: experimental evaluation of unstructured gossip-based implementations. In *Middleware '04: Proceedings of the 5th ACM/IFIP/USENIX international conference on Middleware*, pages 79–98, New York, NY, USA, 2004. Springer-Verlag New York, Inc.

[2] C. Cattuto, D. Benz, A. Hotho, and G. Stumme. Semantic analysis of tag similarity measures in collaborative tagging systems. In *Proceedings of the 3rd Workshop on Ontology Learning and Population (OLP3)*, pages 39–43, July 2008. ISBN 978-960-89282-6-8.

[3] V. Zanardi and L. Capra. Social ranking: uncovering relevant content using tag-based recommender systems. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 51–58, 2008.

[4] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30:107–117, 1998.

[5] D. Fogaras, B. Rácz, K. Csalogány, and T. Sarlós. Towards scaling fully personalized pagerank: Algorithms, lower bounds, and experiments. *Internet Mathematics*, pages 333–358, 2005.

[6] M. Jelasity and O. Babaoglu. T-Man: Gossip-Based Overlay Topology Management. *Engineering Self-Organising Systems*, 3910:1–15, 2006.

[7] S. Amer-Yahia, M. Benedikt, L. Lakshmanan, and J. Stoyanovich. Efficient network aware search in collaborative tagging sites. *Proc. VLDB Endow.*, pages 710–721, 2008.

[8] M. Bender, T. Crecelius, M. Kacimi, S. Miche, J. Xavier Parreira, and G. Weikum. Peer-to-peer information search: Semantic, social, or spiritual? *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 2007.

[9] M. Carman, M. Baillie, and F. Crestani. Tag data and personalized information retrieval. In *SSM '08: Proceeding of the 2008 ACM workshop on Search in social media*, pages 27–34, New York, NY, USA, 2008. ACM.