Algorithms for continuous top-k processing in social networks

Abdulhafiz Alkhouli, Dan Vodislav, Boris Borzic

Algorithms for continuous top-k processing in social networks

Abdulhafiz ALKHOULI, Dan VODISLAV, Boris BORZIC
ETIS, ENSEA / University of Cergy-Pontoise / CNRS
abdulhafiz.alkhouli@ensea.fr, dan.vodislav@u-cergy.fr, boris.borzic@ensea.fr

Abstract

Information streams are mainly produced today by social networks, but current methods for continuous top-k processing of such streams are still limited to content-based similarity. We present the SANTA algorithm, able to handle also social network criteria and events, and report a preliminary comparison with an extension of a state-of-the-art algorithm.

I. Introduction and motivation

Publishing and consuming content through *information streams* is nowadays at the heart of the new Web. Information streams consist of flows of items, usually short semi-structured text messages, possibly containing links to some Web resources (images, videos, pages, etc.), and continuously published through specific diffusion channels, e.g. RSS feeds, blogs, discussion forums, social networks, etc. In this context, users may be both content producers and consumers; they may subscribe to several information channels of interest and continuously receive on it, in real-time, new published content.

Relationships between producers and consumers in this pub/sub framework introduce a *social network* dimension, that varies from no relation at all in the case of RSS feeds, to possible interaction with published messages on blogs and discussion forums, and to explicit user relationships on social networks. To measure the potential interest of information items for users, the social network dimension provides additional criteria, beyond the traditional content-based ones.

A major challenge in this context comes from the huge amount of information available, even when restricted to channels of interest for a user. To target only useful items, filtering and ranking models embedded into easy to use subscription languages and tools are necessary. We focus here on *top-k continuous queries* over information streams, based on a ranking model including social network criteria.

In our view, the ranking model must include at least the following factors: (i) *content based*, measuring the adequacy of the message content with the subscription query; (ii) *user based*, measuring the importance of the message publisher and of its relationship with the subscriber in the social graph; (iii) *interaction based*, measuring the importance of messages by the reaction they provoked through actions of other users on that message (likes, comments, forwards, etc.); (iv) *time based*, measuring the decrease of importance for a message in time, e.g. through sliding time windows [3] or time decay functions [5][4].

Another major challenge in this pub/sub approach for information streams is the design of *efficient processing algorithms* at a very large scale in the number of users / streams. In our top-k query context, the main difficulty comes from the need of continuously (re-)computing the score of every message relative to every subscription query to maintain the result lists. The complexity of this task depends not only on the number of messages and queries, but also on the form of the scoring function.

Two main categories of processing models have been proposed to date. *The static approach* is based on periodic snapshot queries over the set of published messages to get the top-k list for each user. *The continuous approach* handles subscriptions as continuous queries reacting to new messages and to other events, in order to incrementally maintain the top-k results. If the continuous approach is more efficient, it also has more difficulties to handle complex scoring functions. To the best of our knowledge, the continuous methods proposed so far only explored simple scoring functions, most of the time based on the textual content, eventually combined with time factors. More complex scoring, including social network factors has been proposed, but only handled through a static approach.

Our purpose is to go beyond the state of the art methods for continuous processing of top-k queries over information streams, by considering a social network environment with complex scoring functions that include the ranking factors mentioned above. In previous work [1] we proposed the following general form for the scoring function in the case of asymmetric social networks where users may emit messages and interact with other messages:

$$score(m,u) = \alpha CS(m,u) + \beta_1 UI(u_m) + \beta_2 f(u,u_m) + \gamma_1 AI(m) + \gamma_2 AR(m,u) \quad (1)$$

Here the importance of message m for user u is a linear combination of several factors: (i) the content similarity $CS(m,u)$ between m and u's profile, (ii) the global importance $UI(u_m)$ of u_m (the creator of m) in the social network, (iii) the importance $f(u,u_m)$ of u_m for u in the social network, (iv) the global importance $AI(m)$ of m given by the interactions with it, and (v) the importance $AR(m,u)$ of the interaction with m of users that are important for u. For simplicity, we excluded here time factors that we handle through order-preserving decay functions, such as in [5] and [4], since they have no impact on the design of the top-k algorithms. Note that a user profile is here assimilated to a content-based query; in the common case of text messages

and keyword queries, a user profile may be described by a set of keywords with associated weights.

Almost every continuous top-k algorithm for information streams proposed so far is limited to content similarity and time factors. Content similarity is based on homogeneous and monotonic functions, such as cosine similarity combined with tf-idf weights. The closest approach to our work is [3], which proposes the COL-Filter algorithm that indexes subscription queries through ordered lists for each subscription term and uses a threshold algorithm inspired from TA [2] to limit the index traversal. Threshold algorithms on a different index structure are also used by [4] to adapt two IR top-k retrieval strategies to information streams. Only [5] goes beyond content-based factors by adding a global importance of the message, which may be assimilated to the UI and AI terms in (1). They use a two-dimensional inverted query indexing scheme which drastic pruning of the search space. Compared to these approaches, we aim at designing methods able to handle all the terms in (1) and to react to other events than message creation.

II. The SANTA algorithm

We report here preliminary results of our Social and Action Network Threshold Algorithm (SANTA) for continuous top-k processing on information streams, able to handle complex scoring functions such as (1) for both message creation and action events. It uses an index structure based on sorted lists (Figure 1) composed of a text index, a social index and the list μ of the current k-th score for each user. The text index is composed of lists for each term τ_i, containing the users u that have τ_i in the profile, sorted in descending order of the term's weight w_{iu}. The social index is composed of lists for each user u_m, containing the other users u sorted by decreasing importance of u_m for them. The μ list is sorted in descending order of -μ_u (i.e. increasing μ_u).

For simplicity, here we leave out the last term of the scoring function (1) and consider a cosine-like text similarity function, i.e. $score(m,u) = \alpha \sum_{\tau_i \in m} w_{im} w_{iu} + \beta G(m) + \gamma f(u,u_m)$, where w_{im} is the weight of τ_i in m and $G(m)$ gives the global importance of m in the network by grouping the UI and AI terms of (1). Then a message m enters into the top-k of u if $F(m,u) = score(m,u) - \mu_u > 0$. SANTA applies a threshold strategy such as TA [2] to traverse the index lists and get candidates u for top-k change. It maintains a decreasing threshold $T = \overline{F}(m,u)$ computed with the current value in each list. The monotony of F and of the index lists enables the algorithm to stop when $T \leq 0$.

We compare SANTA with an extension of the state-of-the-art algorithm COL-Filter [3]. COL-Filter was designed for text similarity only, based on a text index similar to ours. The difference is that COL-Filter combines the text and μ criteria

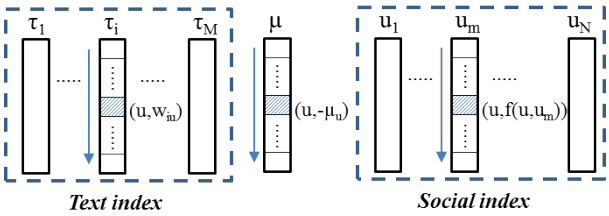

Figure 1: *The SANTA index structure*

by dividing the w_{iu} entries by μ_u, so does not need a μ list. A message m enters the top-k of u if $GF(m,u) = score(m,u)/\mu_u > 1$. If this strategy accelerates the search, it cannot be applied to scoring functions such as (1) and needs updates of the index entries when μ changes in time. We propose here an extension CF+ of the COL-Filter strategy to scoring function (1) as follows. We divide the entry values in the text and social index by μ_u and sort the μ list by $1/\mu_u$. Here the COL-Filter strategy can be applied with $GF(m,u) = score(m,u)/\mu_u = \alpha \sum_{\tau_i \in m} w_{im} w_{iu}/\mu_u + \beta G(m)/\mu_u + \gamma f(u,u_m)/\mu_u$.

We compared SANTA and CF+ on a social network of 103,000 users extracted from Twitter. Algorithms are initialized with 300,000 messages then processing time is measured on 22,020 messages and 30,000 actions. Note that an action increases the score of a message through the AI component of $G(m)$. We consider also a variant, SANTA$_{CF}$, which handles index updates exactly like CF+, after top-k updates, while SANTA makes updates during search. The table below reports the average processing time per event (in ms), with search + update time for CF+ and SANTA$_{CF}$.

Event	CF+	SANTA$_{CF}$	SANTA
New message	0.14 + 8.37	0.41 + 1.01	0.37
New action	1.95 + 970.45	5.74 + 2.12	8.41

Results show that SANTA behaves much better than CF+, which is heavily penalized by the update time. SANTA is better than SANTA$_{CF}$ for messages and similar for actions. Future work will address improvements for action processing and extensions of the scoring components.

References

[1] A. Alkhouli, D. Vodislav, and B. Borzic. Continuous top-k processing of social network information streams: a vision. In *ISIP 2014 post proceedings*. Springer, 2015.

[2] R. Fagin. Combining fuzzy information: An overview. *SIGMOD Rec.*, 31(2):109–118, June 2002.

[3] P. Haghani, S. Michel, and K. Aberer. The gist of everything new: Personalized top-k processing over web 2.0 streams. In *CIKM '10*, pages 489–498, 2010.

[4] A. Shraer, M. Gurevich, M. Fontoura, and V. Josifovski. Top-k publish-subscribe for social annotation of news. *Proc. VLDB Endow.*, 6(6):385–396, Apr. 2013.

[5] N. Vouzoukidou, B. Amann, and V. Christophides. Processing continuous text queries featuring non-homogeneous scoring functions. In *CIKM '12*, pages 1065–1074, 2012.