A Demo Blog Recommendation System for SmartScribo

Diana Zaiceva, Dmitry Korzun
Department of Computer Science
Petrozavodsk State University
Petrozavodsk, Russia
{zaiceva, dkorzun}@cs.karelia.ru

Abstract

The SmartScribo system is an application for mobile multi-blogging implemented on the Smart-M3 information sharing platform. We consider the problem of semantic blogging in such a massive and heterogeneous media-space as the blogosphere is recently. In our earlier work, we introduced a simple tag ranking model for retrieval a small set of relevant blogs from the blogosphere and for their provision to the user. This demo implements the model for proactive and personalized blog recommendations; it shows experimentally the validity of our approach for semantic blogging.

Index Terms: Smart-M3, Recommendation system, Mobile semantic multi-blogging.

SmartScribo is a Smart-M3 application for advanced access to the blogosphere [1]. The SmartScribo architecture consists of three types of distributed agents: clients, blog processors, and mediators. They collaborate by sharing knowledge in the blogosphere smart space. Clients operate with blogs locally and synchronize their content with the space. Since a client typically runs on a mobile user device, the former should not implement computationally expensive operations, delegating most of processing to blog processors and mediators. Blog processors track notifications (in the blogosphere smart space) from clients, relaying particular blog-services. Introduction of blog mediators into the architecture aims at additional features for blogging, including such scenarios of semantic blogging as blog recommendation service or blog rankings [2], [3].

We focus on the following scenario (Fig. 1), which implements a personalized and proactive blog recommendation service for SmartScribo users. A user operates with her/his SmartScribo client as previously. SmartScribo collects context of user activity and blog popularity. Based on the context, the client shows a dialog message with recommendations for reading new blogs. Then the user can retrieve the posts of the recommended blogs and start their reading or she/he can discard the recommendation.

Our design of the recommendation system employs a context subspace in the blogosphere smart space to share "statuses" of blog data. A status reflects such user activity as sending, reading, or deleting blog messages. Other context information is related to blogs, e.g., reflecting their instant popularity. In the recommendation scenario, $n_p \geq 1$ is the sum number of tags in post p and T_{pu} is the latest access time of user u to post p. Clients and blog processors update this information during their regular activity.

The most part of processing is delegated to a dedicated blog mediator. In our recommendation scenario, the mediator accesses the context subspace, computes personalized ranks, finds appropriate blogs using available blog search engines and then publishes the recommendation into the blogosphere smart space.

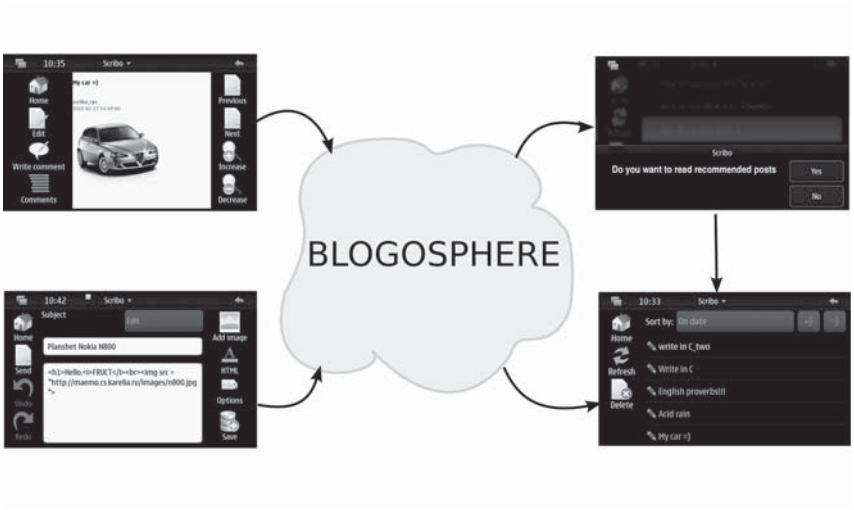

Fig. 1. Blog recommendation scenario for SmartScribo users

Our demo implements the tag ranking model that we developed in our previous work [3]. The blog mediator tracks the personal context space of a given user u. The tag-based blog activity can be characterized with the time elapsed from the latest access to posts with tag t,

$$\delta_t = \delta_t(u) = T_0 - \max_{t \in q, q \neq p} T_{qu}, \tag{1}$$

where T_0 is the current time.

The personalized tag ranks are (re)computed whenever the blog mediator detects in the smart space some activity with t. If a post p with a new tag t appears then the initial rank $R_{tu} = 1/n_p$ is formed. If u reads or writes a post p with the known tag t then the rank is updated:

$$R_{tu} = (1 - \alpha) R_{tu}^{\text{old}} + \frac{\alpha}{\delta_t n_p}, \tag{2}$$

where $0 < \alpha \leq 1$ is the weight of new information. That is, if u has read posts with t very recently then u likely accepts other blogs with t. The dependence on n_p states that blogs with narrower thematic are more influential.

The blog mediator was implemented in Python. Appropriate modifications were done in the original SmartScribo client (Python, Maemo 5). The code is available for free download and use at http://sourceforge.net/projects/smartscribo/. Early experiment results show that proactive and personalized retrieval of blogs can be implemented on the Smart-M3 platform in computationally effective manner. Although the ranking model is simple, the approach allows discovering relevant blogs.

REFERENCES

[1] D. Zaiceva, I. Galov, and D. Korzun, "A blogging application for smart spaces," in *Proc. 9th Conf. of Open Innovations Framework Program FRUCT and 1st Regional MeeGo Summit Russia–Finland*, Apr. 2011, pp. 154–163.

[2] D. Zaiceva, I. Galov, A. Sannikov, and D. Korzun, "Mobile multi-blogging in Smart-M3: Architecture and scenarios," in *Proc. 8th Conf. of Open Innovations Framework Program FRUCT*, Nov. 2010, pp. 225–233.

[3] D. Zaiceva and D. Korzun, "A tag-based blog recommendation system for Smart-M3 multi-blogging," in *Proc. 10th Conf. of Open Innovations Association FRUCT and 2nd Finnish–Russian Mobile Linux Summit*, Nov. 2011, pp. 163–169.