Influence of Social Communication on Content-Based Recommendation

Bernadetta Maleszka
Faculty of Computer Science
and Management
Wrocaw University of Science and Technology
St. Wyspianskiego 27, 50-370 Wroclaw
Email: bernadetta.maleszka@pwr.edu.pl

Marcin Maleszka
Faculty of Computer Science
and Management
Wrocaw University of Science and Technology
St. Wyspianskiego 27, 50-370 Wroclaw
Email: marcin.maleszka@pwr.edu.pl

***Abstract*—One of basic divisions of information retrieval systems is content-based and collaborative filtering. Some hybrid methods exist combining both of them, but certain aspects still remain unexplored. In this paper we explore one: the influence of users communicating via social media on content-based recommendation systems. While in the system users do not know each other, outside they may make their own preferences known (e.g. tweeting recommendations), thus influencing the preferences of other users. Here we simulate several different types of such communication and its influence on content-based recommendation system. We intend to use this results for improving the quality of such systems.**

***Keywords*—information retrieval; knowledge diffusion; user preference evolution; user profile**

I. Introduction

As the amount of information in the Internet is constantly increasing, information retrieval and user modeling need to deal with new challenges and even today remain valid research problems. In particular, user personalization in information retrieval systems is needed for recommending the user documents that are relevant to his information needs and limiting the overall number of documents displayed. Studies show that users will rarely browse more than the first few pages of presented results [12]. The basis of search engines – user query – often does not reflect the true need of the user, as they may not be able to properly formulate it. Additionally, the same query may have different meaning for different users, or even for the same user in different contexts.

In our previous research we were dealing with this problem by creating complex user models and adapting them to better reflect the user [11], [12], based on the population of other users in the system. One issue not taken into account in that research, is the communication of users outside the system, possibly by means of social media. A user, after viewing a document, may recommend it to his friends using some social platform. This information may also be seen by random people on the Internet. In both cases their own opinion of the document will improve and there will be a larger chance that they are interested in it, even if it is not among their previous interests.

In this paper we try to model such situation, using our previous user models created with hidden assumption of isolated users [12] and our research into agent models of social users communication [9]. We test different dynamics of social interaction of modeled users and their influence on models created in the information retrieval system. Such experiment is impossible on a real world basis, as users cannot be entirely isolated from their social communication channels. This paper presents only preliminary results of our experiments. We intend to use them for creating better user models in information retrieval systems.

The rest of this paper is organized as follows. Section II contains a short overview of relevant research. In Section III we provide description of the information retrieval system, assumptions made in its creation, and the model of user preference evolution. Section IV contains description of the experiment conducted on this model. The paper is concluded in Section V describing general results and future areas of research.

II. Related Works

Bouadjenek et al. [2] define Information Retrieval as the science that deals with the representation, storage, organization of, and access to information items in order to satisfy the user requirements concerning this information. It covers a wide range of problems connected with user satisfaction. One of the most important is ambiguity. Given a document, each user has his own understanding of its content. Therefore, each user uses different terms and words to describe or retrieve some information. To satisfy user needs, personalization is an appropriate solution to provide an adaptive model of the user and to improve the information systems usability.

In literature one can find two types of classical approaches: content-based information retrieval or collaborative filtering [11]. According to Hadjouni et al. [5] personalization methodologies are generally divided into two complementary processes, which are (1) the user information collection, used to describe the user interests and (2) the inference of the

978-1-5090-5795-5/17/$31.00 © 2017 IEEE

gathered data to predict the closest content to the user expectation. The first approach can be connected with content-based information retrieval and concentrates on modifications of users' queries, mainly expanding them by adding some additional terms. The aim of this process is to disambiguate the meaning of user request. Another possibility is to use collaborative filtering or social-based filtering where systems use information about connections between users in some groups. The second part of the process extracts contextual information from previous user activities.

One of the disadvantages of search engines available today is the fact that they are designed for a single user who searches alone. Thus, users cannot benefit from the experience of each other for a given search task. Morris [14] developed *SearchTogether*, a collaborative search interface, where several users who share an information need collaborate and work together with others to fulfill that need.

In this paper we propose a methodology to describe a model of user communication. The basis for it is the research done on agent communication with some influence from social network analysis.

Multiple multi-agent systems used as decision support systems use centralized approach to communication, i.e. there is some single supervisor agent that gathers information from all observation/ functional agents. The other agents may not even know about their neighbors. Some examples include traffic control systems [7], where most agents control small parts of the system (e.g. a single crossroads), but a single supervisor agent is tasked with optimizing overall traffic flow in the whole area. This may be extended to more levels in a larger decision making or learning hierarchy [1]. On the other hand there are more decentralized systems. Examples include a decentralized network of agents working for clustering some data [3], with additional verification agents controlling the system and if necessary – correcting the behavior of other agents; multi-agent systems used for different games, like Go [8]; or surveillance systems [15]. There are also hybrid centralized-decentralized systems [6], which sometimes use asynchronous modes of decentralized communication.

Our approach to communication, described in detail in [9], is based in large part on these ideas for multi-agent systems. We use asynchronous communication between identical agents in a decentralized system, but we also add some additional structure of more preferred communication channels. This is similar to the *friend* relation and other ideas from the study of social group communication and social networks. In this research areas, different modes of communication were also studied. For example the centralized systems may represent hierarchies in employee structure in some companies (studied to improve productivity) and decentralized systems are the very idea of social networks. Our approach is similar to proposed in [13], where authors considered a group of students working on some research problems, showing that the underlying social network is not important to the final result of integration. Only if strong ties may be determined between participants, the improvement of the group results is possible. Building a model for knowledge dissemination in a social group communication was also proposed in [10], in order to improve the teaching process.

III. Model Description

A. User Preference

Document Model

In our system we assume the following definition of library (set of documents).

$$D = \{d_i : i = 1, 2, \ldots, n_d\} \tag{1}$$

where n_d is a number of documents and each document d_i is described in the following way:

$$d_i = \{(t^i_j, w^i_j) : t^i_j \in T \wedge w^i_j \in [0.5, 1), j = 1, 2, \ldots, n^i_d\} \tag{2}$$

where t^i_j is index term coming from assumed set of terms T, w^i_j is appropriate weight and n^i_d is a number of index terms that describe document d_i.

Determining User Preference

In the system we have a user that retrieves some documents in the library, using some keywords – index terms. We assume that the user is represented by his preference in the following way:

$$T_U = \{(t_j, v_j) : t_j \in T \wedge v_j \in [0.5, 1), j = 1, 2, \ldots, n_u\} \tag{3}$$

where t_j is the index term, v_j is the appropriate weight of user interest in particular term t_j and n_u is the number of user preferences at the moment.

User preference is a set of weighted terms which are taken from the set of terms T. We need to model user preference because we would like to perform the experimental evaluations without real users (instead of this we prepare a method to simulate user activities in document retrieval system).

The overall idea of creating user preference is as follows. To obtain user preference we define a set of relevant documents which are taken from a library. Next, we calculate an average value of weights for each term. If the calculated weight is larger than assumed threshold w_{min} and occurs in the documents many times (greater than d_{min}) we assign this weight for the term. This procedure is presented in details in Algorithm 1.

To simulate user activities in document retrieval system we need to determine a way to create user queries and a method for preference modification.

Clarke et al. [4] have shown that more than 85% user queries consist of a maximum of three terms. Based on this result, we assume that the user puts two or three terms in single query. The terms are taken from his preference.

In our previous works [12] we dealt with the problem of user preference modification. We assumed there that preference is changing with time. It was modeled in the following way: in the subsequent series of user queries in the system, some of relevant documents should be changed by a new one. In a result of this procedure, weights of some terms are modified.

Algorithm 1: Determining user preferences.

```
Input: $D_r$ – set of relevant documents
Output: $Pref$ – user preference (set of weighted terms)
foreach term $t$ in documents descriptions from set $D_r$ do
    Calculate how many times this term occurred – $l(t)$ ;
    Sum the weights $w(t)$ ;
    if $l(t) > d_{min}$ and $w(t) > w_{min}$ then
        Calculate the average value of each term $t$ – $w_{avg}(t)$;
        Add new term $t$ with weight $w_{avg}(t)$ to user preferences;
```

In this paper we consider another way of user preference modification – we hope that proposed method is closer to real (taking into account social connections between users).

B. Evolution of User Preference

We model communication of users outside the system by an agent communication model we described in [9], called asynchronous decentralized communication. The model operates based on the following description:

- Between sessions there is some time where users may communicate in irregular intervals. Every discrete time moment each of them has a chance P_c that it will start communication. The probability that they will communicate in the first moment is $P_1 = P_c$ and in each following moment it is $P_n = P_c \cdot (1 - P_c)^{n-1}$. The chance of communication P_n is a limited sequence, but outgoing communication is not guaranteed (its probability increases for $P_c \to 1$). The user (user agent) selects one of the terms that are interesting to him with probability:
$$P_t = \frac{v_t}{\sum_i v_i}. \tag{4}$$
The communicate that they will send is a pair $< t, v_t >$.
- Communication is outgoing only. The user randomly selects one other user, that they will send their term information. Response is not expected and verification of properly receiving the information is not conducted.
- Users have some preferred receivers (”friends”). When the user (user agent) selects the target of their communication there is a chance that they will conduct the selection from the overall population or only from its subgroups (only ”friends”).
- The receiving user (user agent) first determines if the message was send along a preferred communication channel (”to a friend”) or broadcast to general population. Depending on this, they will use different parameter α for integrating this information. The integration will be done by changing the weight for this term in the following way:
$$v_t^{n+1} = (1 - \alpha)v_t^n + \alpha v_t', \tag{5}$$
where v_t' is the received weight.

Based on our previous research [9], each user will be informed of at least one term from each other user at longest in the following timeframe:

$$T_D = max_{i,j \in \{1,\ldots,n\}} \Big(min\{T_{i,j}, T_{i,x_1} + T_{x_1,j}, \ldots, T_{i,x_1} + \textstyle\sum_{z=1}^{n-1} T_{x_z,x_{z+1}} + T_{x_n,j}\} \Big) \tag{6}$$

where $T_{i,j}$ is the estimated time between communicates being exchanged between user i and user j. Each user will be informed about every interest of every other user in a multiple of this value, estimated as:

$$T = \frac{T_D}{min_t P_t}. \tag{7}$$

C. Modeling User Profile

The aim of document retrieval system is to present relevant documents to the user. To obtain the objective, the system should gather some information about user interests. In our approach the system does not know user preference (neither terms nor theirs weights). It builds user profile which should converge to user preference. Better user profile (closer to user preference) should guarantee better documents recommendation.

In a static case, when user preference is stable (does not change), the more information about user interests system has, the better profile should be created. In current approach, the preference is changing and procedure of profile modification should allow to adapt the user profile to real user interests.

In our previous works [11] we have presented a method for building and tuning user profiles. Here we present the overall idea. We divided user activities into sessions (a few queries). In each session user obtains some set of documents connected with submitted queries:

$$D(s) = \{(q_i^{(s)}, d_{i_j}^{(s)}) : i_j = 1, 2, \ldots, i_J\} \tag{8}$$

where s is session’s number, i is query’s number and i_J is a number of documents relevant to query $q_i^{(s)}$.

Based on the set of documents the system can calculate the mean value of each term’s weight:

$$w_d^{(s)}(t_j) = \frac{1}{n_s} \sum_{i=1}^{n_s} w_{d_i}^{(s)}(t_j) \tag{9}$$

where $w_d^{(s)}(t_j)$ is the mean value of weights in document set $D(s)$ and $w_{d_i}^{(s)}(t_j)$ is weight of concerned term t_j in single document d_i in current session s.

When user is interacting with the system, his profile should be modified according to his current interests. We propose to calculate the difference between weights in two last sessions using two methods: absolute and relative.

The absolute change is calculated using the following formula:

$$\Delta_1 w_{t_l}(s) = \begin{cases} w_{d_i}^{(s)}(t_l) - w_{d_i}^{(s-1)}(t_l), & \text{if } s > 1 \\ w_{d_i}^{(s)}(t_l), & \text{otherwise} \end{cases} \tag{10}$$

where $w_{d_i}^{(s)}(t_l)$ is the average weight of term t_l after current session s and $w_{d_i}^{(s-1)}(t_l)$ is a weight of term t_l after previous session $s-1$.

The relative change is calculated in the following way:

$$\Delta_2 w_{t_l}(s) = \begin{cases} \dfrac{w_{d_i}^{(s)}(t_l) - w_{d_i}^{(s-1)}(t_l)}{w_{d_i}^{(s-1)}(t_l)}, & \text{if } w_{d_i}^{(s-1)}(t_l) > 0 \\ w_{d_i}^{(s)}(t_l), & \text{otherwise} \end{cases} \tag{11}$$

where $w_{d_i}^{(s)}(t_l)$ is the average weight of term t_l after current session and $w_{d_i}^{(s-1)}(t_l)$ is a weight of term t_l after previous session.

The relative change takes into account the proportion of absolute difference to the value.

Based on the difference we prepare two methods to update the set of terms and theirs weights. The methods were described in details in [12] – here we present it only in short. The first proposition is based on popular assumption that never terms are more important than older ones (12).

$$w_p^{(s+1)}(t_l) = \begin{cases} w_d^{(s)}(t_l), & \text{if } t_l \text{ is new term} \\ w_p^{(s)}(t_l) + \gamma \cdot \Delta_1 w_d^{(s)}(t_l), & \text{otherwise} \end{cases} \tag{12}$$

where $w_{t_l}(s)$ is weight of term t_l in user profile in session s, $w_d^{(s)}(t_l)$ is average weight of term t_l in description of relevant documents in current session, $\gamma \in [0,1]$ is a factor that reflects how important is new knowledge and $\Delta_1 w_d^{(s)}(t_l)$ is absolute change of user interests' degree in considered term t_l calculated using equation (10).

In the second approach we use relative change of interests in particular terms (13).

$$w_p^{(s+1)}(t_l) = \beta \cdot w_p^{(s)}(t_l) + (1-\beta) \cdot \frac{A}{1+\exp(-B \cdot \Delta_2 w_d^{(s)}(t_l) + C)} \tag{13}$$

where $w_p^{(s+1)}(t_l)$ is weight of term t_l in user profile in session $s+1$; $\Delta_2 w_d^{(s)}(t_l)$ is relative change of user interests' degree in considered term t_l calculated using equation (11) and A, B, C and β are parameters that should be attuned in experimental evaluation.

Both methods of profile updating are based on incremental improvement of terms' weights.

IV. Experimental Evaluation

In this initial phase, the main aim of the experiments is evaluating the effectiveness of user profile tuning methods, created for randomly changing preferences, in situation where the change of preferences depends on the social communication of users. In general a favorable result in such experiments would be decreasing distance between user profile and user preference, as well as low initial distance. To explore the effectiveness of our model in the proposed social communication schema, we conducted simulations in a prototype information retrieval system developed on Java platform.

A. Plan of Experiments

We operate on a randomly created set of documents, with parameters based on the real world library of Wroclaw University of Science and Technology. To generate it, we first create a set T of 100 terms: $T = \{t_1, t_2, \ldots, t_{100}\}$, then using it we create set D of documents, each described by 2 to 5 index terms with weight between 0.5 and 1:

$$\begin{gathered} D = \Big\{ d_i = \{(t_i, w_i)\} : t_i \in T \wedge \\ \wedge w_i \in [0.5, 1), i \in \{1, 2, \ldots, 10000\} \Big\} \end{gathered} \tag{14}$$

We create 100 users, with initial preferences being a set of 6 to 10 random terms selected from T and weights calculated as described in Algorithm 1. For each user we add up to 5 preferred other agents, that have similar preferences (up to 50% of terms and weights may be identical).

We simulate the activity of one observed user in the system – we assume that he conducts queries in blocks consisting of 150 queries with 2 to 4 terms (this follows observations noted in [12]).

After each 5 blocks of user activity, we conduct a communication phase, where other users may influence the preference of this user (and in turn, he may influence their preferences). This follows the procedure described in Section III-B, but users send information about documents, not single terms. This phase is split into 100 time moments. In each, every user has a chance P_c to initiate communication. They may select a random one of their preferred users (with chance P_f) or a random user from the overall group.

As the user works with the system, we tune his profile closer to his preference. If his preference changes, the profile will start adapting to the new one.

B. Results

The tested models of user profile adaptation were designed to work with assumption of no communication between users, so a long parameter tuning process was required to achieve good quality of results. After tuning the models gave results comparable to their initial use.

Assuming different parameters of the social communication of users, we observed the adaptation of the user profile by both models over one hundred iterations, each consisting of 5 blocks of activity and communication phase. We measure the euclidean distance between the user profile and user preference as quality of the model. Some sample runs are shown in figures 1 – 4. In particular, figures 1 and 2 show the influence of communication chance on models, and figures 3 and 4 show the influence of chance of communication to friends. In all runs, the first model (denoted on graphs as *mode1*) performed worse than the second one (denoted *mode2*), but while the former is mostly constant, the latter at first adapts quickly but then slowly its quality decreases.

Analysis of entire results show two especially important observations. First, as the chance of social communication increases and the user preference changes more often, both models of user profile adaptation become more similar. The second is that for high chance of friend communication, the first model does not adapt quickly enough to overcome this ”*peer pressure*”.

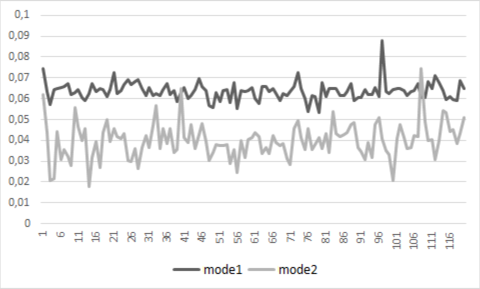

Fig. 1. Sample simulation result for low overall communication chance.

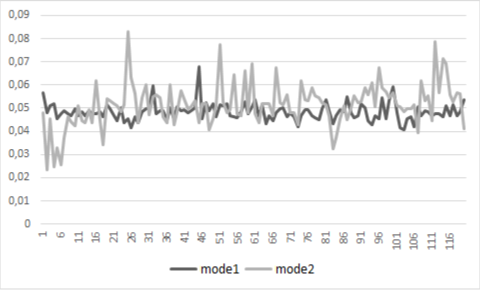

Fig. 2. Sample simulation result for high overall communication chance.

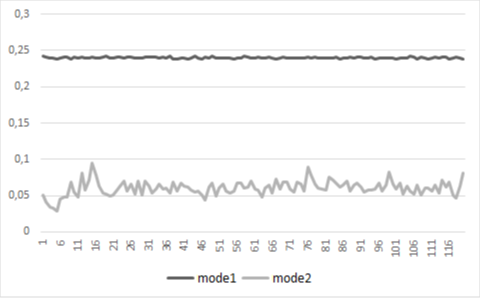

Fig. 3. Sample simulation result for average overall communication chance and high chance of communication to friends.

V. Conclusion

In this paper we applied our previous research into social agent communication to modeling user evolution in a content-based collaborative system. These systems rarely focus on more than one user, as their interaction is not modeled. We show that models omitting it may still work, but dedicated ones should still be created. In today world of common social media interaction, isolated users are hard to find and taking into account these interactions should be obligatory. We speculate that with time, purely content-based systems may become obsolete with hybrid content-collaborative systems taking their place. Still, the experimental simulation shows that after fine tuning models assuming a single user still work, but the adaptability in them is significantly lacking. But even those models show, that different social situations may lead to increased errors in those models, like the shown case of ”*peer pressure*”.

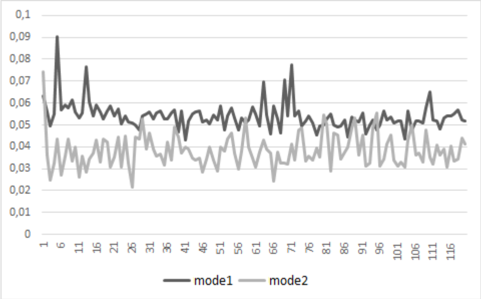

Fig. 4. Sample simulation result for average overall communication chance and low chance of communication to friends.

We intend to use all these results in our future research, working to create a hybrid information retrieval system better suited to users communicating via social media.

Acknowledgment

This research was co-financed by Polish Ministry of Science and Higher Education grant.

References

[1] Abed-alguni B.H., Chalup S.K., Henskens F.A., Paul D.J.: A multi-agent cooperative reinforcement learning model using a hierarchy of consultants, tutors and workers. In: Vietnam Journal of Computer Science Vol 2, Issue 4, Springer 2015, 213-226.

[2] Bouadjenek M. R., Hacid H., Bouzeghoub M.: Social Networks and Information Retrieval, How Are They Converging? A Survey, a Taxonomy and an Analysis of Social Information Retrieval Approaches and Platforms. Information Systems, Elsevier, 2016, 56, pp. 1–18.

[3] Chaimontree S., Atkinson K., Coenen F.: A multi-agent based approach to clustering: Harnessing the power of agents. Agents and Data Mining Interaction, Springer Berlin Heidelberg, 2012, pp. 16–29.

[4] Clarke C. L. A., Cormack G., Tudhope E. A., Relevance ranking for one to three term queries. Information Processing & Management 36 (2000) 291–311.

[5] Hadjouni M., Haddad M. R., Baazaoui H., Aufaure M. A., Ghezala H. B.: Personalized Information Retrieval Approach. IN: Proceedings of WISM 2009.

[6] Hale M. T., Nedic A., Egerstedt M.: Cloud-Based Centralized/Decentralized Multi-Agent Optimization with Communication Delays. arXiv preprint arXiv:1508.06230, 2015.

[7] Iscaro G., Nakamiti G.: A supervisor agent for urban traffic monitoring. IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), IEEE 2013, pp. 167–170.

[8] Jiang A., Marcolino L. S., Procaccia A. D., Sandholm T., Shah N., Tambe M.: Diverse randomized agents vote to win. Advances in Neural Information Processing Systems, 2014, pp. 2573–2581.

[9] Maleszka M.: Select this result for bulk action Observing collective knowledge state during integration. In: Journal of Intelligent & Fuzzy Systems, vol. 32, no. 2, 2017, pp. 1241–1252.

[10] Maleszka M., Nguyen N.T., Urbanek A., Wawrzak-Chodaczek M.: Building Educational and Marketing Models of Diffusion in Knowledge and Opinion Transmission. Computational Collective Intelligence, Technologies and Applications, Lecture Notes in Artificial Intelligence, Springer International Publishing, 2014, pp. 164–174

[11] Mianowska B., Nguyen N. T.: A Method for Tuning User Profiles Based on Analysis of User Preference Dynamics in Document Retrieval Systems. W: Proceedings of ICAISC 2012. LNCS 7267, s. 673–681 (2012).

[12] Mianowska B., Nguyen N. T.: Tuning User Profiles Based on Analyzing Dynamic Preference in Document Retrieval Systems. Multimedia Tools and Applications, DOI 10.1007/s11042-012-1145-6 (2012).

[13] De Montjoye Y.-A., Stopczynski A., Shmueli E., Pentland A., Lehmann S.:The Strength of the Strongest Ties in Collaborative Problem Solving. Scientific reports 4, Nature Publishing Group 2014.

[14] Morris M. R., Horvitz E., Searchtogether: An interface for collaborative web search. In: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, UIST 2007, 3–12.

[15] Peterson C. K., Newman A. J., Spall J. C.: Simulation-based examination of the limits of performance for decentralized multi-agent surveillance and tracking of undersea targets. SPIE Defense+ Security, International Society for Optics and Photonics, 2014, pp. 90910F–90910F.