Semantics-Aware Content-Based Recommender Systems: Design and Architecture Guidelines☆

Ludovico Boratto, Salvatore Carta, Gianni Fenu, and Roberto Saia[1]

Dipartimento di Matematica e Informatica, Università di Cagliari
Via Ospedale 72 - 09124, Cagliari, Italy

Abstract

Recommender systems suggest items by exploiting the interactions of the users with a system (e.g., the movies to recommend to a user are selected by considering all the movies she/he already evaluated). In particular, *content-based* systems suggest items whose content is similar to that of the items evaluated by a user. An emerging application domain in content-based recommender systems is represented by the consideration of the semantics behind the item description, in order to have a disambiguation of the words in a description and improve the recommendation accuracy. However, different phenomena, such as a changes in user taste over time or the use of her/his account by third parties, might affect the accuracy by considering items that do not reflect the real user preferences. Starting from analysis of the literature and of an architecture proposed in a recent survey, in this paper we first highlight the current limits in this research area, then we propose design guidelines and an improved architecture to build semantics-aware content-based recommendations.

Keywords: Semantics-aware Recommender Systems, Design, Architecture

---

☆This work is partially funded by Regione Sardegna under project SocialGlue, through PIA - Pacchetti Integrati di Agevolazione "Industria Artigianato e Servizi" (annualità 2010), and by MIUR PRIN 2010-11 under project "Security Horizons".

Email address: `{ludovico.boratto, salvatore, fenu, roberto.saia}@unica.it` (Ludovico Boratto, Salvatore Carta, Gianni Fenu, and Roberto Saia)

1. Introduction

A recommender system is designed to provide suggestions for items that are expected to interest a user [1]. One of the most employed approaches in the literature and in real-world applications (e.g., e-commerce websites) are the so-called *content-based recommender systems* [2]. These systems analyze the content of the items a user has previously evaluated (e.g., their textual description), in order to detect items that she/he has not considered yet and are similar to those she/he likes. Emerging application domains in this area are represented by those systems and services that involve the use of ontologies and semantic analysis tools in content-based recommender systems, in order to perform a disambiguation of the item descriptions and improve its accuracy [3, 4]. This leads to the generation of a class of systems known in the literature as *semantics-aware content-based recommender systems* [5, 2], which have recently emerged.

In their very recent survey, de Gemmis et al. proposed a high-level architecture of a semantics-aware content-based recommender system [2]. However, over the last few years several novel problems that involve the architecture and the engineering of a recommender system have arisen. These current open issues are now presented in detail.

Presence of incoherent items in a user profile. Most of the solutions regarding the *user-profiling* task of a recommender system involve a filtering of the whole set of items previously evaluated by a user, in order to measure their similarity with those that she/he did not consider yet, and recommend the most similar items [2]. Indeed, the recommendation process is usually based on the principle that users' preferences remain unchanged over time and this can be true in many cases, but it is not the norm due to the existence of temporal dynamics in their preferences [6, 7, 8]. Therefore, a static approach to user profiling can lead toward wrong results due to various factors, such as a simple change of tastes over time or the temporary use of their own account by other people.

Magic barrier problem. Some studies [9, 10] have showed that a subset of the user ratings might be considered as outliers, due to the fact that the same user may rate the same item with different ratings, at different moments in time. This is a well-known problem, which in literature is defined as *magic barrier* [11, 12, 13], a term used to identify the point at which, due to the noise in the data, the performance and accuracy of an algorithm cannot be further improved. After the magic barrier has been reached any improvement in terms of accuracy might mean an overfitting instead of a performance enhancement. Therefore, the magic barrier problem is very relevant in the recommendation research, but no approach has ever studied it from a content-based point-of-view how to filter out items whose content represents an outlier.

Our contributions. In this paper, we first analyze the state-of-the-art architecture of a content-based recommender system, then we will explore in detail the possible problems that might occur by employing it. Some design guidelines on how to enrich that architecture will be proposed, and a novel architecture, which allows the system to tackle the highlighted problems and improve the effectiveness of the recommendation process, will be presented. Even though we will focus on the emerging application domain we previously mentioned (i.e., the semantics-aware systems), we will also show the usefulness of our proposal on classic content-based approach. Being able to properly exploit the semantics of the content of the items is essential in order to build effective recommendations. Therefore, this study is meant to provide both architectural and practical tools for any researcher or developer involved in the development of real-world semantics-aware content-based recommender systems. The scientific contributions coming from the paper are now summarized:

- we will analyze the state-of-the-art architecture of a semantics-aware content-based recommender system to study, for the first time in the literature, what might happen in the recommendation process if incoherent items are filtered by the system;
- this is the first study in which the magic problem is studied in a content-

based recommender system and from the architectural point of view;

- we present design guidelines and a novel architecture, in order to improve the existing one and overcome the aforementioned issues;

- we will analyze the impact of the components we will introduce in the proposed architecture from a computational cost point-of-view.

The rest of the paper is organized as follows: Section 2 presents related work on content-based recommender systems and on the emerging problems and application domains that have arisen over the last few years and that affect the classic architecture of a system; in Section 3 we will explore the state-of-the-art architecture of a semantics-aware content-based recommender system; Section 4 will highlight the limits that the current architecture presents and introduce design guidelines to improve it; Section 5 will propose an improved architecture, by following the design guidelines; Section 6 presents conclusions and future work.

2. Related Work

Content-based recommender systems suggest to users items that are similar to those they previously evaluated [2, 14]. The early systems used relatively simple retrieval models, such as the Vector Space Model, with the basic TF-IDF weighting. The Vector Space Model is a spatial representation of text documents, where each document is represented by a vector in a n-dimensional space (known as *bag of words*, and each dimension is related to a term from the overall vocabulary of a specific document collection. Examples of systems that employ this type of content filtering are [15, 16, 17, 18]. Due to the fact that the approach based on a simple bag of words is not able to perform a semantic disambiguation of the words in an item description, content-based recommender systems evolved and started employing external sources of knowledge (e.g., ontologies) and semantic analysis tools, to improve their accuracy [3, 4, 5].

Regarding the user profile considered by a recommender system, there is a common problem that may affect the effectiveness of the obtained results, i.e., the capability of the information stored in the user profile to lead toward reliable recommendations. In order to face the problem of dealing with unreliable information in a user profile, the state of art proposes different strategies. Several approaches, such as [7], take advantage from the Bayesian analysis of the user provided relevance feedback, in order to detect non-stationary user interests. Also exploiting the feedback information provided by the users, other approaches such as [8] make use of a tree-descriptor model to detect shifts in user interests. Another technique exploits the knowledge captured in an ontology [19] to obtain the same result, but in this case it is necessary that the users express their preferences about items through an explicit rating. In [20, 21, 22], the problem of modeling semantically correlated items was tackled, but the authors consider a temporal correlation and not the one between the items and a user profile.

Considering the item incoherence problem, it should be noted that there is another common issue that afflicts the recommendation approaches. This is a problem that in the literature is identified as *magic barrier* [11], a term used to define the theoretical boundary for the level of optimization that can be achieved by a recommendation algorithm on transactional data [23]. The evaluation models assume as a ground truth that the transactions made in the past by the users, and stored in their profiles, are free of noise. This is a concept that has been studied in [24, 9], where a study aimed to capture the noise in a service that operates in a synthetic environment was performed.

No approach in the content-based recommendation literature ever studied how the architecture and the flow of computation might be affected by the item incoherence and magic barrier issues.

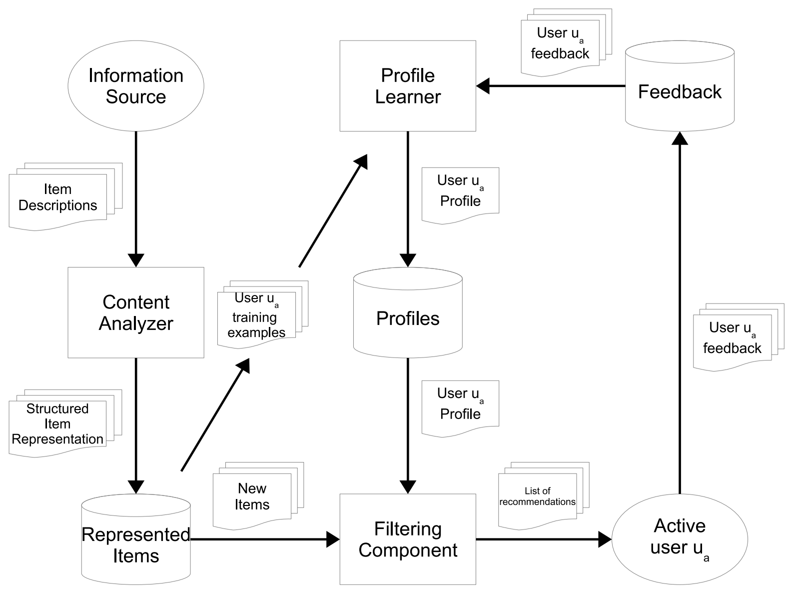

Figure 1: Architecture of a semantics-aware content-based recommender system.

3. A State-of-the-Art Architecture for Semantics-Aware Content-based Recommender Systems

This section will present the high-level architecture of a semantics-aware content-based recommender system proposed in [2] and presented in Figure 1. In order to highlight the limits of this architecture and present our proposal, we will explore it by presenting the flow of the computation of a system that employs it.

The description of the items usually has no structure (e.g., text), so it is necessary to perform some pre-processing steps to extract some information from it. Given an *Information source*, represented by the *Item Descriptions* (e.g., product descriptions, Web pages, news, etc.) that will be processed during the filtering, the first component employed by a system is a Content Analyzer. The component coverts each item description into a format processable by the following steps (i.e., keywords, n-grams, concepts, etc.) thanks to the employment of feature extraction tools and techniques. The output generated by this component is a *Structured Item Representation*, stored in a *Represented Items* repository.

Out of all the represented items, the system considers the ones evaluated by each active user u_a to whom recommendations have to provided (*User* u_a *training examples*), in order to build a profile that contains the preferences of the user. This task is accomplished by a PROFILE LEARNER component, which employs Machine Learning algorithms to combine the *structured item representations* in a unique model. The output produced by the component is a *user profile*, stored in a *Profiles* repository.

The recommendation task is performed by a FILTERING COMPONENT, which compares the output of the two previous components (i.e., the profile of the active user and a set of items she/he has not evaluated yet). Given a new item representation, the component predicts wether or not the item is suitable for the active user u_a, usually with a value that indicates its relevance with respect to the user profile. The filtered items are ranked by relevance and the top-n items in the ranking represent the output produced by the component, i.e., a *List of recommendations.*

The *List of recommendation* is proposed to the *active user* u_a, which either accepts or rejects the recommended items (e.g., by watching a recommended movie, or by buying a recommended item), by providing a feedback on them (*User* u_a *feedback*), stored in a *Feedback* repository.

The feedback provided by the active user is then used by the system to update her/his user profile.

4. Limits at the State of the Art and Design Guidelines

In the previous section, we presented the state-of-the-art architecture of a semantics-aware content-based recommender system. We will now present the possible problems that might occur by employing it and provide design guidelines on how to improve it.

The possible problems that might occur will be presented through possible use cases/scenarios that might occur.

Scenario 1. The account of the active user is used by another person, who evaluates items that the user would have never evaluated (e.g., she/he buys items that the active user would have never bought). This would lead to the presence of noise in a user profile, since the *Structured Item Representation* of these incoherent items with respect to the user profile would be considered by the PROFILE LEARNER component. The component would make them part of the *user u_a profile*, stored as it is in the *Profiles* repository, and employed in the recommendation process by the FILTERING COMPONENT. This would generate bad recommendations and the accuracy of the system would strongly be affected.

Scenario 2. The preferences of the active user change over time, but the oldest items that do not reflect the current preferences of the user, but positively evaluated by her/him, are still part of the user profile. A form of *aging* of the items in a user profile would allow the system to ignore such items after some time, but until that moment those items would represent noise. That noise might affect the system for a lot of time, since the aging process is usually gradual and the items age slowly. Again, this would affect the recommendation accuracy.

Scenario 3. If a mix of the two previous scenarios occurs and these type of problems are iterated over time, the system would reach the so-called *magic barrier*, i.e., a point where the noise affects the system so much that it is impossible to improve the accuracy any further. As highlighted in Section 2, the problem has been widely studied in the Collaborative Filtering literature, in order to identify and remove the noisy items based on their ratings. No paper in the literature studied the magic barrier from a content-based point of view, so the state-of-the-art architecture previously presented is limited also from that perspective.

The three previously presented scenarios put in evidence that the architecture of a semantics-aware content-based system should be able to deal with the presence of incoherent items in the user profiling process, in order to avoid the previously aforementioned problems. Therefore, we will now present design guidelines on how to improve the state-of-the-art-art architecture of a system.

The first scenario highlighted the need for a system to detect how coherent is an item with the rest of the items that have been evaluated by a user, in order to detect the presence of noise. This could be done by comparing the content of the item (i.e., the *structured item representation*) with that of the other items evaluated by the user (*user u_a training examples*).

Scenario 2 confirms the need for a system to evaluate the temporal correlation of an item with the rest of the items in the user profile. Indeed, if an item is too old and, as previously said, too different with respect the other items, it should be removed from a user profile.

Both the second and the third scenarios highlighted that the presence of noisy/incoherent items on a user profile should be reduced to a very limited amount of time. In particular, thanks to scenario 3 we know that these items should not be discarded gradually, but the system should be able to do a one-off removal. This would allow the filtering component to consider only items that are coherent with each other and with the preferences of the users.

The next section will adopt these design guidelines to present an architecture that overcomes these issues.

5. An Improved Architecture to Build Semantics-aware Content-based Recommender Systems

In this section we will propose our architecture. The updated high-level architecture of the system is first proposed (Section 5.1), and in Section 5.2 we will present the details of the novel component that faces the problems highlighted in the previous section. We will close our presentation with a brief analysis that shows how our proposal fits with the development of a real-world system in the big data era (Section 5.3).

5.1. High-Level Architecture

Figure 2 presents an updated version of the state-of-the-art architecture illustrated in Section 3. The proposed architecture integrates a novel component, which we named PROFILE CLEANER, with the aim to analyze a profile and remove the incoherent items, before storing it in the *Profiles* repository. In order to solve the previous problems, the component should be able to remove an item if it meets the following two conditions:

1. the coherence/content-based similarity of the item with the rest of the profile is under a *Minimum Coherence* threshold value;
2. it is located in the first part of the user iteration history. Based on this requirement, an item is considered far from the user's preferences only when it goes up in the first part of the iterations (i.e., when the distance with the last evaluated item is higher than a *Maximum Temporal Distance* threshold).

By removing the incoherent old items, the FILTERING COMPONENT would consider only the real preferences of the users and the previously mentioned problems are solved. Indeed, by checking that both conditions are met, the system avoids removing from a profile the items that are diverse from those she/he previously considered, but that might be associated to a recent change in the preferences of the user.

Regarding scenario 1, if among a *user* u_a *training examples* there is an incoherent item evaluated by a third party, it would be detected by the component, since it receives it as an input. Regarding scenarios 2 and 3, by checking the temporal correlation of an item with the others in the user profile, the component would be able to remove an item as soon as it becomes old and incoherent, avoiding the problems related to the *aging* strategies (which might still be employed by the PROFILE LEARNER, but are not enough) and to the presence of too many incoherent items that would lead to the *magic barrier* problem.

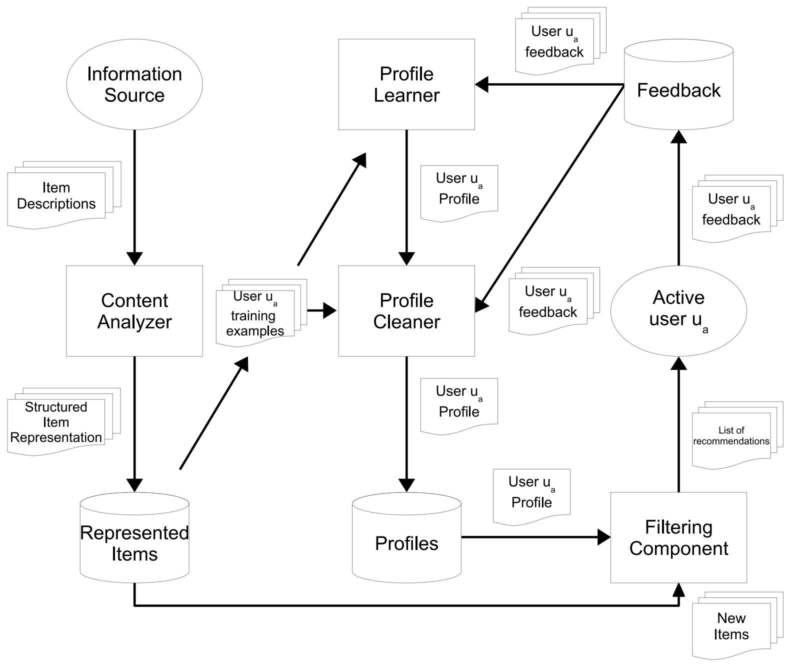

Figure 2: Architecture of a semantics-aware content-based recommender system.

5.2. Low-level Representation of the Profile Cleaner

In Figure 3 we inspect furthermore on the component introduced in our architecture, to present a low-level analysis and the subcomponents it should employ to accomplish its task.

As Figure 2 showed, the profile cleaner takes as input both an *item* i a user has evaluated (i.e., one of the *training examples* or of the *feedbacks* provided by a user) and a *user profile*.

The Items Coherence Analyzer subcomponent compares the structured representation of an item i with the rest of the user profile, in order to detect the coherence/similarity of the item with the rest of the profile. If the *Structured Item Representation* involves semantic structures (e.g., Wordnet synsets), as the modern content-based systems do, several metrics can be employed to evaluate the semantic similarity between two structured representations that involve synsets. The state-of-the-art ones are the following five: i.e., *Leacock*

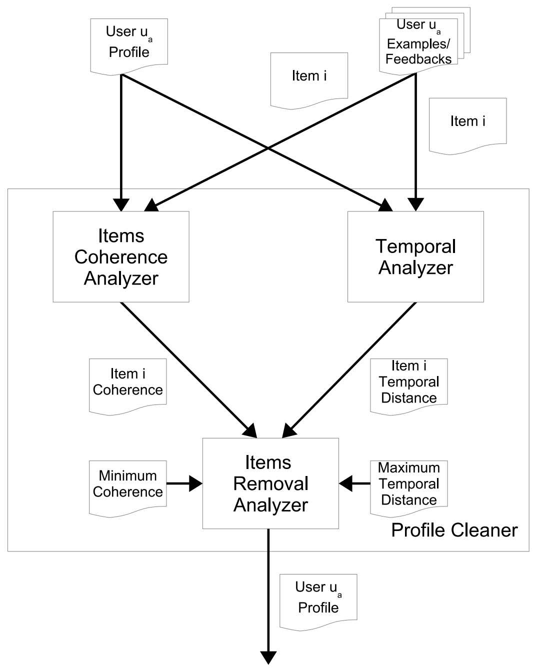

Figure 3: Architectural organization of the profile cleaner task.

and Chodorow [25], *Jiang and Conrath* [26], *Resnik* [27], *Lin* [28], and *Wu and Palmer* [29]. However, any type of similarity/coherence might be employed, even if no semantic information is available the item representation (e.g., TF-IDF). The output produced by the subcomponent is an *Item i Coherence* value, which will be later employed by the Items Removal Analyzer subcomponent to decide if the item should be removed or not.

In parallel, the *Temporal Analyzer* subcomponent will consider how far was the evaluation of the considered item with respect to that of the other items in the user profile (and especially the last evaluated one). The distance threshold might be defined as a fixed value, or by defining *regions* based on the chronology with which the items have been evaluated (e.g., to remove an item if it was evaluated in the first two quarters that contain the oldest items). The output is an *Item i Temporal Distance*, which will also be employed by Items Removal Analyzer subcomponent.

The output of the two previously subcomponents is then handled by the Items Removal Analyzer which also receive as input the *Minimum Coherence* and *Maximum Temporal Distance* thresholds, and decides if the considered item i should be removed from a user profile or not. The output produced by the subcomponent (and by the Profile Cleaner main component) is a cleaned *user u_a profile*, which does not contain the incoherent and oldest items.

5.3. Developing a System that Employs this Architecture

It becomes natural to think that the introduction of a Profile Cleaner component, even if useful, might lead to heavy tasks to be computed by the system. Indeed, the component has to deal with a comparison between each item and the rest of the user profile, and this similarity might involve semantic elements and measures, which are usually very heavy to compute. Given the widely-known *big data* problem that characterizes and affects the systems nowadays, here we will try to inspect on how to develop this component in real-world scenarios.

Indeed, the computation of the coherence of each of the new items with the rest of the user profile might distributed over different computers, by employing large scale distributed computing models like MapReduce. Moreover, this process can be handled in background by the system, since when a user evaluates a new item, it would hardly make any instant difference on the computed recommendations. Therefore, if it gets removed in a reasonable time and with a distributed approach, the employment of Profile Cleaner component would be both effective and efficient at the same time.

Moreover, we studied the structure of the Profile Cleaner component to let it run two subcomponents in parallel, so that even under this perspective the process can be parallelized and efficient.

In conclusion, we believe that even if we are introducing a possibly heavy computational process, the improvements in terms of accuracy and the structure of the component would overcome the complexity limits. Moreover, this complexity would also be efficiently dealt with the current technologies employed to face the big data problems (e.g., Hadoop's MapReduce).

6. Conclusions and Future Work

In this paper, we dealt with the problems that might occur with the current way in which content-based recommender systems are engineered and designed.

Given the high impact that emerging aspects are having in research and real-world recommender systems, such as the introduction of the semantics in the filtering process and the so-called *magic barrier* problem, we analyzed the current architecture employed by a content-based recommender system and highlighted current limits. Indeed, we showed that a form of cleaning of the user profiles is necessary in order to overcome these limitations.

We then proposed an updated architecture, which was analyzed both from a high-level point of view and by inspecting on the component that allows a system to clean a profile. Moreover, we studied the application of our proposal in real-world scenarios, which would probably be characterized by the big data problem.

Future work will move from the software engineering perspective of our study, to develop real-world efficient implementations of this architecture (e.g., on a grid), in order to study its efficiency and effectives in scenarios characterized by the big data (e.g., the recommendations performed by an e-commerce website).

References

[1] F. Ricci, L. Rokach, B. Shapira, Introduction to recommender systems handbook, in: F. Ricci, L. Rokach, B. Shapira, P. B. Kantor (Eds.), Recommender Systems Handbook, Springer, 2011, pp. 1–35.

[2] M. de Gemmis, P. Lops, C. Musto, F. Narducci, G. Semeraro, Semantics-aware content-based recommender systems, in: F. Ricci, L. Rokach, B. Shapira (Eds.), Recommender Systems Handbook, Springer US, 2015, pp. 119–159. `doi:10.1007/978-1-4899-7637-6_4`.
URL `http://dx.doi.org/10.1007/978-1-4899-7637-6_4`

[3] M. Capelle, F. Frasincar, M. Moerland, F. Hogenboom, Semantics-based news recommendation, in: Proceedings of the 2Nd International Conference on Web Intelligence, Mining and Semantics, WIMS '12, ACM, New York, NY, USA, 2012, pp. 27:1–27:9.

[4] M. Capelle, F. Hogenboom, A. Hogenboom, F. Frasincar, Semantic news recommendation using wordnet and bing similarities, in: Proceedings of the 28th Annual ACM Symposium on Applied Computing, SAC '13, ACM, New York, NY, USA, 2013, pp. 296–302.

[5] P. Basile, C. Musto, M. de Gemmis, P. Lops, F. Narducci, G. Semeraro, Content-based recommender systems + dbpedia knowledge = semantics-aware recommender systems, in: V. Presutti, M. Stankovic, E. Cambria, I. Cantador, A. D. Iorio, T. D. Noia, C. Lange, D. R. Recupero, A. Tordai (Eds.), Semantic Web Evaluation Challenge - SemWebEval 2014 at ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers, Vol. 475 of Communications in Computer and Information Science, Springer, 2014, pp. 163–169.

[6] L. Li, Z. Yang, B. Wang, M. Kitsuregawa, Dynamic adaptation strategies for long-term and short-term user profile to personalize search, in: G. Dong, X. Lin, W. Wang, Y. Yang, J. X. Yu (Eds.), Advances in Data and Web Management, Joint 9th Asia-Pacific Web Conference, APWeb 2007, and 8th International Conference, on Web-Age Information Management, WAIM 2007, Huang Shan, China, June 16-18, 2007, Proceedings, Vol. 4505 of Lecture Notes in Computer Science, Springer, 2007, pp. 228–240.

[7] W. Lam, S. Mukhopadhyay, J. Mostafa, M. J. Palakal, Detection of shifts in user interests for personalized information filtering, in: SIGIR, 1996, pp. 317–325.

[8] D. H. Widyantoro, T. R. Ioerger, J. Yen, Learning user interest dynamics with a three-descriptor representation, JASIST 52 (3) (2001) 212–225.

[9] X. Amatriain, J. M. Pujol, N. Oliver, I like it... I like it not: Evaluating user ratings noise in recommender systems, in: G. Houben, G. I. McCalla, F. Pianesi, M. Zancanaro (Eds.), User Modeling, Adaptation, and Personalization, 17th International Conference, UMAP 2009, formerly UM and AH, Trento, Italy, June 22-26, 2009. Proceedings, Vol. 5535 of Lecture Notes in Computer Science, Springer, 2009, pp. 247–258.

[10] W. C. Hill, L. Stead, M. Rosenstein, G. W. Furnas, Recommending and evaluating choices in a virtual community of use, in: I. R. Katz, R. L. Mack, L. Marks, M. B. Rosson, J. Nielsen (Eds.), Human Factors in Computing Systems, CHI '95 Conference Proceedings, Denver, Colorado, USA, May 7-11, 1995., ACM/Addison-Wesley, 1995, pp. 194–201.

[11] J. L. Herlocker, J. A. Konstan, L. G. Terveen, J. Riedl, Evaluating collaborative filtering recommender systems, ACM Trans. Inf. Syst. 22 (1) (2004) 5–53.

[12] A. Said, B. J. Jain, S. Narr, T. Plumbaum, Users and noise: The magic barrier of recommender systems, in: J. Masthoff, B. Mobasher, M. C. Desmarais, R. Nkambou (Eds.), User Modeling, Adaptation, and Personalization - 20th International Conference, UMAP 2012, Montreal, Canada, July 16-20, 2012. Proceedings, Vol. 7379 of Lecture Notes in Computer Science, Springer, 2012, pp. 237–248.

[13] A. Bellogín, A. Said, A. P. de Vries, The magic barrier of recommender systems - no magic, just ratings, in: V. Dimitrova, T. Kuflik, D. Chin, F. Ricci, P. Dolog, G. Houben (Eds.), User Modeling, Adaptation, and Personalization - 22nd International Conference, UMAP 2014, Aalborg, Denmark, July 7-11, 2014. Proceedings, Vol. 8538 of Lecture Notes in Computer Science, Springer, 2014, pp. 25–36.

[14] M. J. Pazzani, D. Billsus, Content-based recommendation systems, in: P. Brusilovsky, A. Kobsa, W. Nejdl (Eds.), The Adaptive Web, Springer-

Verlag, Berlin, Heidelberg, 2007, pp. 325–341.
URL http://dl.acm.org/citation.cfm?id=1768197.1768209

[15] M. Balabanović, Y. Shoham, Fab: Content-based, collaborative recommendation, Commun. ACM 40 (3) (1997) 66–72.

[16] D. Billsus, M. J. Pazzani, A hybrid user model for news story classification, in: Proceedings of the Seventh International Conference on User Modeling, UM '99, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1999, pp. 99–108.
URL http://dl.acm.org/citation.cfm?id=317328.317338

[17] H. Lieberman, Letizia: An agent that assists web browsing, in: Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI'95, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1995, pp. 924–929.
URL http://dl.acm.org/citation.cfm?id=1625855.1625975

[18] M. Pazzani, J. Muramatsu, D. Billsus, Syskill & webert: Identifying interesting web sites, in: Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 1, AAAI'96, AAAI Press, 1996, pp. 54–61.
URL http://dl.acm.org/citation.cfm?id=1892875.1892883

[19] V. Schickel-Zuber, B. Faltings, Inferring user's preferences using ontologies, in: Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA, AAAI Press, 2006, pp. 1413–1418.

[20] G. Stilo, P. Velardi, Time makes sense: Event discovery in twitter using temporal similarity, in: Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT) - Volume 02, WI-IAT '14, IEEE Computer Society, Washington, DC, USA, 2014, pp. 186–193.

[21] G. Stilo, P. Velardi, Temporal semantics: Time-varying hashtag sense clustering, in: Knowledge Engineering and Knowledge Management, Vol. 8876 of Lecture Notes in Computer Science, Springer International Publishing, 2014, pp. 563–578.

[22] G. Stilo, P. Velardi, Efficient temporal mining of micro-blog texts and its application to event discovery, Data Mining and Knowledge Discovery.

[23] A. Said, B. J. Jain, S. Narr, T. Plumbaum, S. Albayrak, C. Scheel, Estimating the magic barrier of recommender systems: a user study, in: W. R. Hersh, J. Callan, Y. Maarek, M. Sanderson (Eds.), The 35th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '12, Portland, OR, USA, August 12-16, 2012, ACM, 2012, pp. 1061–1062.

[24] X. Amatriain, J. M. Pujol, N. Tintarev, N. Oliver, Rate it again: increasing recommendation accuracy by user re-rating, in: L. D. Bergman, A. Tuzhilin, R. D. Burke, A. Felfernig, L. Schmidt-Thieme (Eds.), Proceedings of the 2009 ACM Conference on Recommender Systems, RecSys 2009, New York, NY, USA, October 23-25, 2009, ACM, 2009, pp. 173–180.

[25] C. Leacock, M. Chodorow, Combining local context and wordnet similarity for word sense identification, in: C. Fellbaum (Ed.), WordNet: An Electronic Lexical Database, MIT Press, 1998, pp. 305–332.

[26] J. J. Jiang, D. W. Conrath, Semantic similarity based on corpus statistics and lexical taxonomy, arXiv preprint cmp-lg/9709008.

[27] P. Resnik, Using information content to evaluate semantic similarity in a taxonomy, in: Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI'95, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1995, pp. 448–453.
URL `http://dl.acm.org/citation.cfm?id=1625855.1625914`

[28] D. Lin, An information-theoretic definition of similarity, in: J. W. Shavlik (Ed.), Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconsin, USA, July 24-27, 1998, Morgan Kaufmann, 1998, pp. 296–304.

[29] Z. Wu, M. Palmer, Verbs semantics and lexical selection, in: Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics, ACL '94, Association for Computational Linguistics, Stroudsburg, PA, USA, 1994, pp. 133–138.