2009 Ninth International Conference on Intelligent Systems Design and Applications

Content-based Filtering with Tags: the FIRSt System

Pasquale Lops, Marco de Gemmis, Giovanni Semeraro, Paolo Gissi, Cataldo Musto, Fedelucio Narducci
Department of Computer Science - University of Bari Aldo Morof
Via E. Orabona, 4 - I70126 Bari, Italy
{*lops,degemmis,semeraro,gissi,musto,narducci*}*@di.uniba.it*

Abstract **Basic content personalization consists in matching up the attributes of a user proCle, in which preferences and interests are stored, against the attributes of a content object. This paper describes a content-based recommender system, called FIRSt, that integrates user generated content (UGC) with semantic analysis of content. The main contribution of FIRSt is an integrated strategy that enables a content-based recommender to infer user interests by applying machine learning techniques, both on ofCcial item descriptions provided by a publisher and on freely keywords which users adopt to annotate relevant items. Static content and dynamic content are preventively analyzed by advanced linguistic techniques in order to capture the semantics of the user interests, often hidden behind keywords. The proposed approach has been evaluated in the domain of cultural heritage personalization.**

Keywords**-Web 2.0; Information Filtering; User Modeling; Recommender Systems;**

I. INTRODUCTION

The role of Web people has evolved from passive consumers of information to that of active contributors and, in this context, information Cltering systems must adapt their behavior to individual users by learning their preferences in order to construct user proCles to be exploited for selecting relevant items. Starting from a corpus containing all the informative content, Information Filtering techniques perform a progressive removal of non-relevant content according to information about user interests, previously acquired and stored in a user proCle [1]. Systems implementing the content-based approach analyze a set of documents, usually textual descriptions of the items previously rated by an individual user, and build a model or proCle of user interests based on the features of the objects rated by that user [2]. In this approach static content associated to items is used to learn a proCle. The proCle is then exploited to recommend new relevant items. The advent of the Web 2.0*f* has radically changed the role of people from passive consumers of information to that of active contributors who create and share new content. According to Tim O Reilly[1], the term Web 2.0*f* means putting the user in the center, designing software that critically depends on its users since the content, as in Flickr, Wikipedia, Del.icio.us, or YouTube, is contributed by thousands or millions of users. That is why Web 2.0 is also called the participative Web*f*. One of the forms of UGC is *folksonomy*, a taxonomy generated by users who collaboratively annotate and categorize resources of interests with freely chosen keywords called *tags*. Folksonomies provide new opportunities and challenges in the Cld of recommender systems. It should be investigated whether they might be a valuable source of information about user interests and whether they could be included in user proCles. Indeed, several difCculties of tagging systems have been observed, such as polysemy and synonymy of tags, or the different expertise and purposes of tagging participants that may result in tags at various levels of abstraction to describe a resource, or the chaotic proliferation of tags [3]. This paper presents an approach in which the process of learning user proCles is performed both on static content and UGC. In order to overcome some limitations of keyword-based approaches, semantic analysis*f* of content is proposed. The key idea is the adoption of knowledge bases, such as lexicons or ontologies, for both the annotation of the items and the proCle representation in order to obtain a more semantic*f* interpretation of the user information needs. Semantic analysis is the key to learn more accurate proCles that capture concepts expressing user interests from relevant documents. These *semantic proCles* contain references to concepts deCned in lexicons or ontologies.

II. GENERAL ARCHITECTURE

FIRSt (**F**olksonomy-based **I**tem **R**ecommender **Syst**em) is a system capable of providing recommendations, provided that descriptions of items are available in textual form. It is a semantic content-based recommender system which integrates static content describing the items with dynamic UGC in the process of learning user proCles. Item properties are represented in the form of *textual slots*. For example, a book can be described by three slots: *title*, *authors*, *abstract*. In order to train the system, users were requested to express their preferences for some of the items in the system repository. A preference was expressed as a numerical vote on a 5-point scale (1=strongly dislike, 5=strongly like). Moreover, users were left free to annotate the items with as many tags as they wished. The semantics in the recommendation process is introduced by the *Content Analyzer*, which identiCes relevant concepts. This process selects, among all the possible meanings (senses) of each polysemous word

[1]http://radar.oreilly.com/archives/2006/12/web-20-compact.html, Accessed on March 18, 2009

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.84

that either occurs in the description of the item, the correct one, according to the context in which the word occurs. In this way the Content Analyzer attempts to overcome the problems due to natural language ambiguity. The final outcome of the preprocessing step is a repository of disambiguated items. This semantic indexing is strongly based on natural language processing techniques, such as Word Sense Disambiguation (WSD) [4], and heavily relies on linguistic knowledge stored in the *WordNet* lexical ontology [5]. In order to involve folksonomies in the profile learning process, tags (words chosen by the user during the training phase to describe an item) are stored in an additional slot, different from those containing static content. The generation of the user profile is performed by the *Profile Learner*, which infers the profile as a binary text classifier. The Profile Learner implements a supervised learning technique for learning a probabilistic model of user interests from disambiguated documents, rated in the training phase by that user. Finally the *Recommender* exploits the user profile to suggest relevant items by matching concepts contained in the semantic profile against those contained in documents to be recommended.

III. Content Analyzer: Semantic Indexing of Documents

Semantic indexing of documents is performed by the Content Analyzer, which relies on META (Multi Language Text Analyzer) [6], a natural language processing tool able to deal with documents in English or Italian. The goal of the semantic indexing step is to obtain a concept-based document representation. To this purpose, the text is first tokenized, then for each word, possible lemmas as well as their morpho-syntactic features are collected. Part of speech ambiguities are solved before assigning the proper sense (concept) to each word. This last step requires the identification of a repository for word senses and the design of an automated procedure for performing word-concept association. In the semantic indexing module *WordNet* 2.0 has been embodied. The basic building block for WordNet is the synset (SYNonym SET), a structure containing sets of words with synonymous meanings, which represents a specific meaning of a word. META implements a WSD algorithm, called JIGSAW [7], that takes as input a document encoded as a list of h words in order of their appearance, and returns a list of k WordNet synsets ($k \leq h$), in which each synset s is obtained by disambiguating the target word w based on the *semantic similarity* of w with the words in its context. Note that $k \leq h$ because some words, such as most proper names, might not be found in WordNet, or because of bigram recognition. Semantic similarity computes the relatedness of two words using the Leacock-Chodorow measure [8], which is based on the length of the path between concepts in an IS-A hierarchy [9]. The WSD procedure allows to obtain a synset-based vector space representation, called bag-of-synsets (BOS), that is an extension of the classical bag-of-words (BOW) model. In the BOS model, a synset vector, rather than a word vector, corresponds to a document. FIRSt is able to suggest potentially relevant items to users, as long as an item having *m properties* can be represented in form of *m textual slots*. The text in each slot is represented by the BOS model by counting separately the occurrences of a synset in the slots in which it appears. In other words, every document d is structured in m bags of synsets:

$$d = \langle d^1, d^2, \ldots, d^m \rangle$$

where

$$d^i = \langle s^i_1, s^i_2, \ldots, s^i_{L(i)} \rangle$$

s^i_k is the k^{th} synset in slot i, and $L(i)$ is the total number of synsets in slot i of document d. For each bag of synsets d^i, the corresponding synset-frequency vector is computed:

$$f^i = \langle f^i_1, f^i_2, \ldots, f^i_{L(i)} \rangle$$

Note that the adoption of slots does not jeopardize the generality of the approach since the case of documents not structured into slots corresponds to have just a single slot in our document representation strategy. In order to involve folksonomies in the processing performed by META, *static* content describing the document is integrated with *dynamic* content (tags), collected during the training step. The set of tags provided by all the users who rated a document d is denoted as *SocialTags*(d), while the set of tags provided by a specific user U on d is denoted by *PersonalTags*(U,d). The distinction between personal and social tags aims at evaluating whether including either just personal tags or social tags in user profiles produces beneficial effects on the recommendations. By applying WSD to tags allows us to enhance the document model from representing tags as mere keywords or strings, to exploiting tags as pointers to WordNet synsets (semantic tags). Indeed, META applied to *SocialTags*(d) produces *SemanticSocialTags*(d), the synset-based folksonomy corresponding to *SocialTags*(d). Moreover, *SemanticPersonalTags*(U,d) is the set of synsets obtained by disambiguating the tags given by U on d, thus it is the result of invoking META on *PersonalTags*(U,d). Since the intent is to exploit a reliable context for WSD, whether the target tag occurs in one of the static slots, the text in that slot is used as a context, otherwise we are forced to accept the other tags as a context.

To summarize, by invoking META on a text t, we get META(t) = $(\vec{x}, \vec{y})$, where $\vec{x}$ is the BOS containing the synsets obtained by applying JIGSAW on t, and $\vec{y}$ is the corresponding synset-frequency vector. When t is a list of tags, the resulting BOS can be seen as the intended meaning of those tags. BOS-indexed documents are used in a content-based information filtering scenario for learning accurate *sense-based* user profiles, as discussed in the following section.

IV. PROFILE LEARNER

The generation of the user profile is performed by the *Profile Learner*, which infers the profile as a binary text classifier [10] since each document has to be classified as interesting or not with respect to the user preferences. Therefore, the set of categories is restricted to c_+, the positive class (*user-likes*), and c_- the negative one (*user-dislikes*).

The induced probabilistic model is used to estimate the *a posteriori* probability, $P(c_j|d_i)$, of document d_i belonging to class c_j. The algorithm adopted for inferring user profiles is a Naïve Bayes text learning approach, widely used in content-based recommenders, which is not presented here because already described in [11]. What we would like to point out here is that the final outcome of the learning process is a probabilistic model used to classify a new document in the class c_+ or c_-. Given a new document d_j, the model computes the a-posteriori classification scores $P(c_+|d_j)$ and $P(c_-|d_j)$ by using probabilities of synsets contained in the user profile and estimated in the training phase by exploiting both ratings provided by users and synset frequencies. Each user provided ratings on items using a discrete scale ranging from MIN (strongly dislikes) to MAX (strongly likes). Items whose ratings are greater than or equal to (MIN+MAX)/2 are supposed to be liked by the user and included in the positive training set, while items with lower ratings are included in the negative training set.

The profile contains the user identifier and the *a-priori* probabilities of liking or disliking an item, apart from its content. Moreover, the profile is structured in two main parts: *profile_like* contains features describing the concepts able to deem items relevant, while features in *profile_dislike* should help in filtering out not relevant items. Each part of the profile is structured in slots, resembling the same representation strategy adopted for items. Each slot reports the features (WORDNET identifiers) occurring in the training examples, with corresponding frequencies computed in the training step. Frequencies are used by the Bayesian learning algorithm to induce the classification model (i.e. the user profile) exploited to suggest relevant items in the recommendation phase. A deeper analysis is required for the slot containing tags. Semantic tags are exploited by the Profile Learner to include information about tags in the user profiles. The profile learning process for user U starts by selecting all items (disambiguated documents) and corresponding ratings provided by U. Each item falls into either the positive or the negative training set depending on the user rating, in the same way as previously described in this section. Let TR_+ and TR_- be the positive and negative training set respectively for user U. Several options for generating the user profile can be chosen at this point, depending on the type of content involved in the process. If we would like to infer a user profile strictly related to personal preferences (one-to-one user profile), all the semantic tags obtained from personal tags provided by U on all items she rated should be exploited in the learning step. This means that, for each $d_j \in TR_+ \cup TR_-$, the additional slot for d_j is *SemanticPersonalTags*(U, d_j). On the other hand, if we would like to build a content-collaborative profile for U, semantic tags obtained from social tags provided by users on all items rated by U should be exploited in the learning step. This means that, for each $d_j \in TR_+ \cup TR_-$, the additional slot for d_j is *SemanticSocialTags*(d_j).

Therefore, given a new document d_j, the *recommendation step* consists in computing the a-posteriori classification scores $P(c_+|d_j)$, used to produce a ranked list of potentially interesting items, from which items to be recommended can be selected.

V. EXPERIMENTAL EVALUATION

The dataset considered for the experiments consists of 45 paintings chosen from the collection of the Vatican picture-gallery. In particular, for each element in the dataset an image of the artifact was collected, along with four textual properties: title, artist, description (static content), and tags (dynamic content). 30 non-expert users and 10 expert users according to the *availability sampling strategy* voluntarily took part in the experiments: *expert users* are supposed to have specific knowledge in the art domain, such as museum curators, while *non-expert users* are supposed to be naïve museum visitors. The goal of the experimental evaluation was to measure the predictive accuracy of FIRSt when different types of content are used in the training step. Users were requested to interact with a web application, in order to express their preferences for all the 45 paintings in the collection. A preference was expressed as a numerical vote on a 5-point scale (1=strongly dislike, 5=strongly like). Moreover, users were left free to annotate the paintings with as many tags as they wished. For the overall 45 paintings in the dataset, 4300 tags were provided by non-expert users, while 1877 were provided by expert users. The average number of tags associated with each painting is about 95 for non-expert users and 41 for expert users, thus experiments relied on a sufficient number of user annotations. Since FIRSt is conceived as a text classifier, its effectiveness can be evaluated by classification accuracy measures, namely Precision and Recall [10], where Precision (Pr) is defined as the number of relevant selected items divided by the number of selected items, and Recall (Re) is defined as the number of relevant selected items divided by the total number of relevant items.

An overall measure of predictive accuracy F_β is used in [12] for the evaluation of recommender systems. F_β is defined in the following manner:

$$F_\beta = \frac{(1+\beta^2) \cdot Pr \cdot Re}{\beta^2 \cdot Pr + Re}$$

where β sets the relative degree of importance attributed to Pr and Re. Since users should trust the recommender, it is important to reduce false positives. It is also desirable to provide users with a short list of *relevant* items (even if not all the possible relevant items are suggested), rather than a long list containing a greater number of relevant items mixed-up with *not relevant* ones. Therefore, we set $\beta = 0.5$. These specific measures were adopted because we are interested in measuring how *relevant* a set of recommendations is for a user. In the experiment, a painting is considered *relevant* for a user if the rating is greater than or equal to 4, while FIRSt considers a painting relevant for a user if the a-posteriori probability of the class *likes* is greater than 0.5.

We organized three different experimental sessions, each one with the aim of evaluating the accuracy of FIRSt for a specific community of users:

1) SESSION#1: NON-EXPERT USER COMMUNITY ˘ All paintings are rated and tagged by 30 non-expert users, for whom recommendations are computed.
2) SESSION#2: WHOLE USER COMMUNITY ˘ All paintings are rated and tagged both by expert and non-expert users. Recommendations are provided for the whole set of 40 users.
3) SESSION#3: NON-EXPERT USER COMMUNITY SUPPORTED BY EXPERTS TAGS ˘ In this session we evaluate whether tags provided by experts have positive effects on recommendations generated for non-expert users. All paintings are rated solely by non-expert users, but tags used for generating non-expert user profiles are provided by expert users.

For SESSION#1 and SESSION#2, 5 different experiments were designed, depending on the type of content used for training the system:

- Exp#1: Static Content - only title, artist and description of the paintings, as collected from the official website of the Vatican picture-gallery
- Exp#2: SemanticPersonalTags(U,d)
- Exp#3: Static Content+SemanticPersonalTags(U,d)
- Exp#4: SemanticSocialTags(d)
- Exp#5: Static Content+SemanticSocialTags(d)

For example, SemanticSocialTags(d) in SESSION#1 includes the set of synsets obtained by disambiguating tags provided by all non-expert users who rated d, while in SESSION#2 it includes the set of synsets obtained by disambiguating tags provided by both expert and non-expert users who rated d.

For SESSION#3, 2 different experiments were designed, depending on the type of content used for training the system:

- Exp#1: SemanticSocialTags(d) ˘ SemanticSocialTags(d) includes the set of synsets obtained by disambiguating tags provided by all experts on d. In this way tags provided by experts contribute to the profiles of non-expert users. The aim of the experiment is to measure whether accuracy of recommendations for non-expert users is improved by tags provided by expert users
- Exp#2: Static Content+SemanticSocialTags(d) ˘ SemanticSocialTags(d), as intended in Exp#1 in this session, are combined with static content.

All experiments were carried out using the same methodology, consisting in performing one run for each user, scheduled as follows:

1) select the appropriate content depending on the experiment being executed;
2) split the selected data into a training set *Tr* and a test set *Ts*;
3) use *Tr* for learning the corresponding user profile;
4) evaluate the predictive accuracy of the induced profile on *Ts*.

The methodology adopted for obtaining *Tr* and *Ts* was the K-fold cross validation [13], with K = 5. Given the size of the dataset (45), applying a 5-fold cross validation technique means that the dataset is partitioned into 5 disjoint sets, each containing 9 paintings. The learning of profiles and the test of predictions were performed in 5 steps. At each step, 4 sets were used as the training set *Tr*, whereas the remaining set was used as the test set *Ts*. The steps were repeated until each of the 5 disjoint sets was used as the *Ts*. Results were averaged over the 5 runs.

VI. RESULTS

The first outcome of experiments in SESSION#1 (Table I) is that the integration of tags (social or personal) causes an increase of *precision* in the process of recommending artifacts to users. More specifically, *Precision* of profiles learned from both static content and tags (hereafter, augmented profiles) outperformed the *Precision* of profiles learned from either static content (hereafter, content-based profiles) or just tags (hereafter, tag-based profiles). The improvement of augmented profiles with personal tags (Exp#3) is 1.62 with respect to content-based profiles (Exp#1), while it is about 1 with respect to tag-based profiles (Exp#2 and Exp#4). Lower improvements are observed by comparing results of Exp#5 with those of Exp#4.

The increase in precision of augmented profiles corresponds to a slight and physiological loss of recall. Lowest recall has been observed for Exp#2. This result is not surprising since personal tags summarize cultural interests and represent them in a deeper and more precise way compared to static content, which, on the other hand, allows covering a broader range of user preferences.

To sum up, by observing the F_β figures, we can conclude that for non-expert users, the highest accuracy is achieved by augmented profiles with personal tags.

Table I
RESULTS OF SESSION #1

Exp.	Precision	Recall	F_β
Exp#1	77.01	93.54	79.83
Exp#2	77.63	86.57	79.27
Exp#3	**78.63**	92.79	**81.11**
Exp#4	77.40	91.87	79.92
Exp#5	**77.78**	93.35	**80.46**

Similar results are observed in SESSION#2 (Table II), where the community also includes expert users. It is interesting to compare results of Exp#1, Exp#2 and Exp#3 in SESSION#1 with those of same experiments in SESSION#2, in order to evaluate the accuracy of recommendations provided by content-based proCles, tag-based proCles built using just personal tags, and augmented-proCles with personal tags in both communities. The values of F_β in SESSION#2 are lower than those observed in SESSION#1, thus we can conclude that it is more difCcult to provide recommendations for expert users.

Another interesting Cnding regards proCles built by using social tags (Exp#4). A comparison between results obtained in SESSION#1 and SESSION#2 highlights a signiCcant loss both in precision and recall when expert users are included in the community. Since social tags represent the lexicon of the community, this result might be interpreted as the fact that tagging with more speciCc and technical lexicon does not bring a signiCcant improvement of system predictive accuracy.

Table II
RESULTS OF SESSION #2

Exp.	Precision	Recall	F_β
Exp#1	75.17	92.63	78.11
Exp#2	76.60	89.86	78.93
Exp#3	77.31	90.61	**79.65**
Exp#4	74.91	89.93	77.50
Exp#5	76.60	91.58	**79.19**

SESSION#3 provides a more insight on the impact of the lexicon introduced by expert users on recommendation provided to non-expert users (Table III).

By analyzing results of Exp#1, we observed that precision and recall of tag-based proCles do not outperform those obtained in Exp#4 in SESSION#1, thus suggesting that the speciCc lexicon adopted by expert users does not positively affect recommendations for non-expert users. Anyway, the slight improvement in recall (+0.53) suggests that the more technical tags adopted by experts might help to select relevant items missed by proCles built with simple tags.

Even integrating social tags provided by experts with content does not improve accuracy of recommendations for non-expert users. Indeed, precision and recall observed in Exp#2 do not signiCcantly change compared to results of Exp#5 in SESSION#1.

Table III
RESULTS OF SESSION #3

Exp.	Precision	Recall	F_β
Exp#1	76.98	92.40	79.64
Exp#2	77.47	93.51	80.22

VII. CONCLUSIONS

This paper presented a technique to infer user proCles from both static content, as in classical content-based recommender systems, and tags provided by users to freely annotate items. The main outcome of this work is that the integration of tags causes an increase of the prediction accuracy in the process of CltEring relevant items for users. Furthermore, experiments have shown that the expertise of users contributing to the folksonomy does not actually affect the accuracy of recommendations.

We are currently working on the integration of FIRSt in an adaptive platform for multimodal and personalized access to museum collections. In this context, speciCc recommendation services, based upon augmented proCles, are being developed. Moreover we are trying other methodologies to exploit expert knowledge of users to increase the prediction accuracy of the recommendation system.

REFERENCES

[1] U. Hanani, B. Shapira, and P. Shoval, Information CltEring: Overview of issues, research and systems,*f User Model. User-Adapt. Interact.*, vol. 11, no. 3, pp. 203˘259, 2001.

[2] D. Mladenic, Text-learning and related intelligent agents: a survey,*f IEEE Intelligent Systems*, vol. 14, no. 4, pp. 44˘54, 1999.

[3] S. Golder and B. A. Huberman, The Structure of Collaborative Tagging Systems,*f Journal of Information Science*, vol. 32, no. 2, pp. 198˘208, 2006.

[4] C. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*. Cambridge, US: The MIT Press, 1999, ch. 16: Text Categorization, pp. 575˘608.

[5] G. Miller, WordNet: An On-Line Lexical Database,*f International Journal of Lexicography*, vol. 3, no. 4, 1990, (Special Issue).

[6] P. Basile, M. de Gemmis, A. Gentile, L. Iaquinta, P. Lops, and G. Semeraro, META - MultilanguagE Text Analyzer,*f* in *Proceedings of the Language and Speech Technnology Conference - LangTech 2008, February 28-29, 2008, Rome, Italy*, 2008, pp. 137˘140.

[7] P. Basile, M. de Gemmis, A. Gentile, P. Lops, and G. Semeraro, UNIBA: JIGSAW algorithm for Word Sense Disambiguation,*f* in *Proceedings of the 4th ACL 2007 International Workshop on Semantic Evaluations (SemEval-2007), Prague, Czech Republic*. Association for Computational Linguistics, June 23-24, 2007, pp. 398˘401.

[8] C. Leacock, M. Chodorow, and G. Miller, Using corpus statistics and wordnet relations for sense identification, *Computational Linguistics*, vol. 24, no. 1, pp. 147–165, 1998.

[9] G. Semeraro, M. Degemmis, P. Lops, and P. Basile, Combining Learning and Word Sense Disambiguation for Intelligent User Profiling, in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, M. M. Veloso, Ed., 2007, pp. 2856–2861, iSBN 978-I-57735-298-3.

[10] F. Sebastiani, Machine Learning in Automated Text Categorization, *ACM Computing Surveys*, vol. 34, no. 1, 2002.

[11] M. de Gemmis, P. Lops, G. Semeraro, and P. Basile, Integrating Tags in a Semantic Content-based Recommender, in *Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys 2008, Lausanne, Switzerland, October 23-25, 2008*, 2008, pp. 163–170.

[12] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, Evaluating Collaborative Filtering Recommender Systems, *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, Jan. 2004.

[13] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in *Proc. of IJCAI-95*, 1995, pp. 1137–1145. [Online]. Available: citeseer.ist.psu.edu/kohavi95study.html