2009 Ninth International Conference on Intelligent Systems Design and Applications

Recommendations toward Serendipitous Diversions

Leo Iaquinta, Marco de Gemmis, Pasquale Lops, Giovanni Semeraro
Universitá degli Studi di Bari, Dipartimento di Informatica
via E. Orabona, 4 - 70126 - Italy
{*iaquinta, degemmis, lops, semeraro*}*@di.uniba.it*

***Abstract*—Recommenders systems are used with various purposes, especially dealing with e-commerce and information filtering tools. Content-based ones recommend items similar to those a given user has liked in the past. Indeed, the past behavior is supposed to be a reliable indicator of her future behavior. This assumption, however, causes the over-specialization problem. Our purpose is to mitigate the problem stimulating users and facilitating the serendipitous encounters to happen.**

This paper presents the design and implementation of a hybrid recommender system that joins a content-based approach and a serendipitous heuristic in order to provide also surprising suggestions. The reference scenario concerns with personalized tours in a museum and serendipitous items are introduced by slight diversions on the context-aware tours.

***Keywords*-Context-aware Recommender Systems; Serendipity; Cultural Heritage**

I. Background and motivation

Recommender systems (RSs) help overcome the *information overload* problem by exposing users to the most interesting items. Common expectations concern with relevance, novelty and surprise. Among different recommendation techniques proposed in the literature, the content-based filtering approach is one of the most widely adopted to date. A content-based RS analyzes a set of documents, usually textual descriptions of the items previously rated by an individual user, and build a model or profile of user interests based on the features of the items rated by that user [1]. The profile is then exploited to recommend new items of interest. Each type of filtering methods has its own weaknesses and strengths. Specifically, the content-based approach suffers from *over-specialization*. Indeed, the system recommend items that score highly against a user's profile and, consequently, the user is limited to being recommended for items similar to those already rated. This shortcoming is called *serendipity problem*.

It is useful to make a clear distinction between *novelty* and *serendipity*. As explained by Herlocker et al. in [2], novelty occurs when the system suggests to the user an unknown item that she might have autonomously discovered. On the other hand, a serendipitous recommendation helps the user to find a surprisingly interesting item that she might not have otherwise discovered (or it would have been really hard to discover).

Traditional accuracy metrics for RSs (e.g., MAE that measures the recommender algorithm performance by comparing the algorithm prediction against a user rating of an item) have difficulty to point what is actually useful for the user: a sensible recommendation (which is not always the most accurate one) [3]. The main problem of accuracy metrics follows from their design to judge the accuracy of individual item predictions. Indeed, they seldom judge the contents of entire recommendation lists, even if the users actually interact with these lists. Moreover, all recommendations are made in the context of the current recommendation list and the previous lists that the user has already seen.

Our objective is to try to feed the user also with recommendations that could possibly be serendipitous. Thus, we enrich the architecture of content-based RS with a component devoted to introduce serendipity in the recommendation process. The implementation of the serendipity-inducing module draws inspiration from the real-world situation when a person visits a museum and, while she is walking around, she finds something completely new that she has never expected to find, that is definitely interesting for her.

The demonstrative scenario concerns with personalized museum tours where the serendipitous suggested items are selected exploiting the learned user profile and causing slight diversions on context-aware tours. Indeed, the basic content-base recommender module allows to infer the most interesting items for the active user and, therefore, to arrange them according the spatial layout, the user behavior and the time constraint. The resulting tour potentially suffers from over-specialization and, consequently, some items can be found no so interesting for the user. Therefore the user starts to divert from the suggested path considering other items along the path with growing attention. On the other hand, also when the recommended items are actually interesting for the user, she does not move with blinkers, i.e. she does not stop from seeing artworks along the suggested path. These are accidental opportunities for serendipitous encounters. The serendipity-inducing module perturbs the optimal path with items that are programmatically supposed to be serendipitous for the active user.

The paper is organized as follows: Section II introduces the serendipity issue for information seeking and covers strategies to provide serendipitous recommendations; Section III provides a description of our recommender sys-

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.215

tem and how it discovers potentially serendipitous items in addition to content-based suggested ones; Section IV provides the description of the experimental session carried out to evaluate the proposed ideas; finally, Section V draws conclusions and provides directions for future work.

II. THE SERENDIPITY POINT OF VIEW

The term *serendipity* was coined in the 1754 by Horace Walpole to express the "making discoveries by accident and sagacity of things which one is not on quest of". Serendipitous encounters depend on personal characteristics, e.g. the open minded attitude, the wide culture and the curiosity [4]. Therefore, the subjective nature of serendipity becomes an issue to conceptualize, to analyze and to implement it [5] Anyway, programming for serendipity is feasible [6]. The surrounding objective is to allow users to expand their own knowledge and to preserve the opportunity of serendipitous discoveries. For instance, in the information searching, there are three kind of search [7]:

- seeking information about a well-defined object;
- seeking information about an object that cannot be fully described, but that will be recognized at first sight;
- acquiring information in an accidental, incidental, or serendipitous manner.

It is easy to realize that serendipitous happenings are quite useless for the first way of acquisition, but are extremely important for the last one.

Introducing serendipity in the recommendation process requires an operational strategy. Among different approaches which have been proposed, Toms suggests four strategies, from simplistic to more complex ones [7]:

1) Role of chance or 'blind luck', implemented via a random information node generator.
2) Pasteur principle ("chance favors the prepared mind"), implemented via a user profile.
3) Anomalies and exceptions, partially implemented via poor similarity measures.
4) Reasoning by analogy, whose implementation is currently unknown.

In this paper we propose to integrate the "Anomalies and exceptions" approach in a content-based RS to provide serendipitous recommendations alongside classical ones.

The basic assumption is that serendipity cannot happen if the user already knows what is recommended to her. Thus the lower is the probability that user knows an item, the higher is the probability that a specific item could result in a serendipitous recommendation. The probability that user knows something semantically near to what the system is confident she knows is higher than the probability of something semantically far. If we evaluate semantic distance with a similarity metric, like internal product which takes into account the item description to build a vector and compares it to other item vectors, it results that it is more

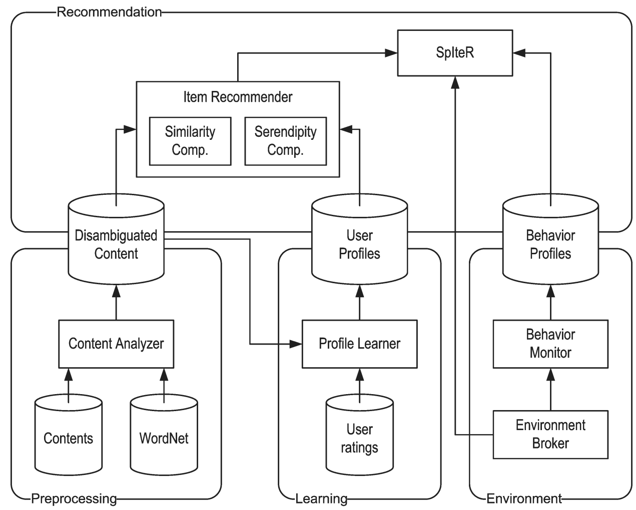

Figure 1. General system architecture

probable to get a serendipitous recommendation providing the user with something less similar to her profile.

According to this idea, items should not be recommended if they are too similar to something the user has already seen. Following this principle, the basic idea underlying the proposed architecture is to ground the search for potentially serendipitous items on the similarity between the item descriptions and the user profile, as described in the next section.

III. SERENDIPITY-PRONE RECOMMENDER SYSTEM

The starting point to provide serendipitous recommendations consists in a content-based RS developed at the University of Bari [8], [9]. The system is capable of providing recommendations for items in several domains (e.g., movies, music, books), provided that descriptions of items are available as text metadata (e.g. plot summaries, reviews, short abstracts). In the following, we will refer to documents as textual metadata about items to be recommended.

Figure 1 shows the general architecture of the system evolved to provide also serendipitous suggestions within the museum scenario. The recommendation process is performed in several steps, each of which is handled by a separate component. First, given a collection of documents, a preprocessing step is performed by the Content Analyzer, which uses the WordNet lexical database to perform Word Sense Disambiguation (WSD) to identify correct senses, corresponding to concepts identified from words in the text. Subsequently, a learning step is performed by the Profile Learner on the training set of documents, to generate a probabilistic model of the user interests. This model is the user profile including those concepts that turn out to be most indicative of the user preferences. Then, the Item Recommender component implements a naïve Bayes text categorization algorithm to classify documents as interesting or not for a specific user by exploiting the probabilistic model learned from training examples. In addition, the Item Recommender contains a sub-module implementing the heuristic to provide serendipity computation. Finally, the SpIteR (Spatial Item Recommender) module rearranges the suggested items in a personalized tour using information about environment and user behavior.

A. Content Analyzer

It allows introducing semantics in the recommendation process by analyzing documents in order to identify relevant concepts representing the content. This process selects, among all the possible meanings (senses) of each polysemous word, the correct one according to the context in which the word occurs. In this way, documents are represented using concepts instead of keywords, in an attempt to overcome the problems due to natural language ambiguity. The final outcome of the preprocessing step is a repository of disambiguated documents. This semantic indexing is strongly based on natural language processing techniques and heavily relies on linguistic knowledge stored in the WordNet lexical ontology [10].

The core of the Content Analyzer is a procedure for Word Sense Disambiguation (WSD), called JIGSAW [11]. WSD is the task of determining which of the senses of an ambiguous word is invoked in a particular use of that word. The set of all possible senses for a word is called sense inventory that, in our system, is obtained from WordNet.The basic building block for WordNet is the *synset* (synonym set), i.e. a group of synonymous words that represents a concept. Since it is not the focus of the paper, the procedure is not described here. What we would like to underline here is that the WSD procedure allows to obtain a synset-based vector space representation, called bag-of-synsets (BOS), that extends of the classical bag-of-words (BOW) model. In the BOS model a synset vector, rather than a word vector, corresponds to a document. Our idea is that BOS-indexed documents can be used for learning accurate sense-based user profiles, as discussed in the following section.

B. Profile Learner

It implements a supervised learning technique for learning a probabilistic model of the interests of the active user. The disambiguated synset-based documents representation allows to have a *semantic* profile with concepts that turn out to be most indicative of the user preferences. We consider the problem of learning user profiles as a binary *Text Categorization* task [12], 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 POS, that represents the positive class (user-likes), and NEG the negative one (user-dislikes). The induced probabilistic model is used to estimate the a-posteriori probability, $P(X|d)$, of document d belonging to class X. The algorithm adopted for inferring user profiles follows a Naïve Bayes text learning approach, widely used in content-based recommenders [1]. More details are reported in [9]. What we would like to point out here is that the final outcome of the learning process is a text classifier able to categorizes a specified item in two classes: POS and NEG.

C. Item Recommender

It exploits the user profile to suggest relevant documents, by matching concepts contained in the semantic profile against those contained in documents to be recommended. The module devoted to discover potentially serendipitous items has been included in this component, in addition to the module which is responsible for the similarity computation between items and profiles.

In order to integrate Toms' "poor similarity" within the recommender, a heuristic has been included in the module for serendipity computation. The module devoted to compare items with profiles (Similarity Computation) produces a list of items ranked according to the a-posteriori probabilities. That list will contain on the top the most similar items to the user profile, i.e. the items high classification score for the class POS. On the other hand, the items for which the a-posteriori probability for the class NEG is higher, will ranked lower in the list. The items on which the system is more uncertain are the ones for which difference between the two classification scores for POS and NEG tends to zero. We could reasonably assume that those items are not known by the user, since the system was not able to clearly classify them as relevant or not. Therefore, the heuristic included in the serendipity module takes into account the absolute value of the difference of the probability of an item to belong to the two classes: $|P(POS|d) - P(NEG|d)|$.

D. SpIteR

It dynamically arranges the suggested items to make the user experience more enthralling. Indeed, the Item Recommender is able to provide a static ordered list of items according to the user assessed interests, but it does not rely on the user interaction with environment. Besides, if the suggested tour simply consists of the enumeration of ranked items, the path is too tortuous and with repetitive passages that make the user disoriented, especially under a time constraint. Finally, different users interact with the environment in different manner, e.g. they travel with different speed, they spend different time to admire artworks, they divert from the suggested tour. Consequently, the suggested personalized tour must be dynamically updated and optimized according to contextual information on the user interaction with the environment.

The tour suggestion task requires knowledge about the item layout. We propose to basically represent items as nodes of an Euclidean graph. Thus the museum tour is quite similar to the classical *Traveling Salesperson Problem*

(TSP). TSP is a well known combinatorial optimization problem and has been studies extensively in many variants. Some aspects of the genetic methods used by SpIteR are shortly presented. Genetic algorithms (GAs) are search algorithms that work via the process of natural selection. They begin with a sample set of potential solutions which then evolves toward a set of more optimal solutions. A solution (i.e., a chromosome for GAs) for the museum tour is a sequence of suggested items. GAs require that a potential solution can be break into discrete parts, namely genes, that can vary independently. It's necessary to evaluate how "good" a potential solution is with respect to other potential solutions. The fitness function is responsible for performing this evaluation. In the museum tour scenario, the fitness function relies on a user-sensitive time constraint, the user behavior (i.e., speed and stay times), the user learned preferences and the item layout.

Further information about museum tour items can be exploited to obtain the problem solution. Usually, few items are placed in rooms and each room is connected with some other rooms. A sample layout of rooms is shown in Figure 2 with the schematic representation of Vatican Pinacoteca. Rooms provide a simpler perspective from combinatorial complexity point of view, but, while a tour visits each item at most once (like in k-TPS), each room can be visited more times.

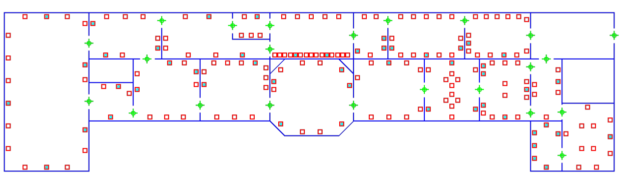

Figure 2. Schematic representation of Vatican Pinacoteca layout

The user profile can be exploited to further reduce nodes involved in the TSP for the museum scenario. Indeed the suggested tour should consist of recommended items, i.e., the k most interesting items for the active user. The question about k value selection arises. Intuitively, the k value depends on how long should be the personalized tour, e.g., the preferred tour duration and the user behavior must be counted. Figure 3 shows a sample tour consisting of the k most interesting items sorted according to the ranking provided by the Item Recommender: bend lines are only a graphical expedient to avoid hidden segments.

Speed and stay times are parameters related to the user behavior. At the beginning, they are estimated on the basis of a stereotypical user profile [13] and then updated according to data collected during the tour from the actual user behavior by the Behavior Monitor module. When the user behavior requires too many significant updates to the behavior profile or the user skips a recommended item or she stays in front of un-recommended items, the tour is planned again taking into account the previous user behavior and the actual viewed items.

Once the personalized tour is achieved starting from the k most interesting items, as shown in Figure 4, serendipitous disturbs are applied. Indeed, the ranked list of serendipitous items is obtained from the Item Recommender module and the previous personalized tour is augmented with some serendipitous items along the path. The resulting solution most likely has a worse fitness value and then a further optimization step is performed. However, the further optimization step should cut away exactly the disturbing serendipitous items, since they compete with items that are more similar with the user tastes. Therefore serendipitous items should be differently weighed from the fitness function, for instance changing their stay time. Indeed, the supposed serendipitous items should turn out not so serendipitous and the user should reduce her stay time in front of such items. Figure 5 shows a "good enough" personalized tour consisting of the most interesting items and the most serendipitous ones. It is amazing to note that some selected serendipitous items are placed in rooms otherwise unvisited by the active user under the her actual time constraint.

IV. Experimental session

The goal of the experimental evaluation is to evaluate the serendipity augmenting effects on personalized tours. The dataset was collected from the official website of the Vatican picture-gallery and it consists of of 45 paintings. For each item, an image of the artifact and three textual metadata (title, artist, and description) were collected. All artworks were laid in the schematic environment model of the Pinacoteca in order to deal with the spatial layout influence on recommending.

We involved 30 users who voluntarily took part in the experiments. The average age of the users was in the middle of twenties. None of the users was an art critic or expert.

In order to evaluate the serendipity augmenting effects on personalized tours, the learned profiles were used to obtain personalized tours with different time constraints and different serendipitous disturbs. Five time constraints were chosen so that tours consisted approximately of 10, 15, 20, 25, 30 items. Serendipitous items ranged from 0 to 7. Table I reports the average of sums and means of *POS* values of tours. The serendipitous item augmenting causes the exploiting of items less similar to the user tastes according to her profile and this effect is particularly evident when there are too many serendipitous items. On the other hand, there is also a decrease when many items are selected according to the user profile, since they are progressively less interesting. When there are many items, the serendipitous item augmenting seems to have no effects over POS mean, but probably this comes from the not very large dataset used.

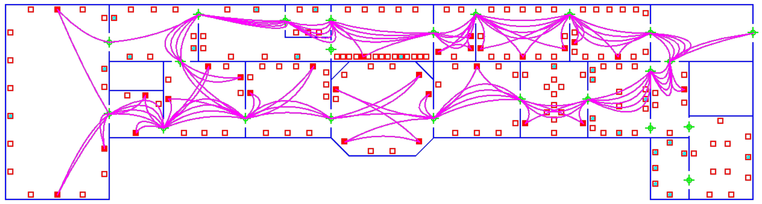

Figure 3. A sample tour consisting of the ranked *k* most interesting items

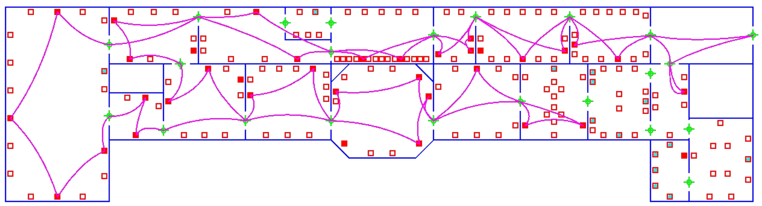

Figure 4. The optimized version of personalized tour in Figure 3

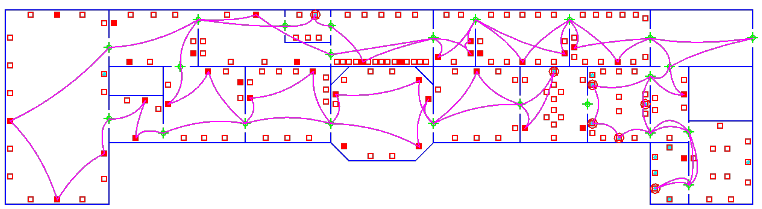

Figure 5. The optimized version of personalized tour whit serendipitous items

	T1		**T2**		**T3**		**T4**		**T5**	
S0	7.18	0.711	10.69	0.714	14.02	0.705	17.21	0.679	19.94	0.671
S1	7.15	0.708	10.61	0.709	14.00	0.704	17.20	0.679	19.89	0.670
S2	7.12	0.705	10.59	0.708	13.98	0.702	17.20	0.679	19.88	0.669
S3	7.08	0.701	10.60	0.708	13.96	0.702	17.19	0.679	19.87	0.669
S4	7.03	0.696	10.58	0.707	13.96	0.701	17.19	0.678	19.87	0.669
S5	6.88	0.681	10.52	0.703	13.95	0.701	17.17	0.678	19.85	0.668
S6	6.54	0.647	10.42	0.696	13.90	0.698	17.11	0.676	19.75	0.665
S7	6.17	0.611	10.19	0.681	13.76	0.692	16.99	0.671	19.64	0.661
Items	10.10		14.97		19.90		25.33		29.70	

Table I
SUMS AND MEANS OF POS VALUES OF TOURS

Table II reports percentages of walking time over the tour. Data show that, increasing the time constraint, less time is spent to walk. Indeed, if few items are selected, they are scattered around (proportionally) many rooms and the user visits room with very few and even no one suggest item. The serendipitous item augmenting seems to increase the walking time. This result is quite amazing according to the selection serendipitous item strategy, i.e., items that are along to a previously optimized path. Actually, the walking time percentage mainly increases because serendipitous items are introduced as new genes of a "good enough" chromosome (solution). However, the augmented chromosome tends to evolve toward the previous one. Thus the new genes should be promoted with a benefit over the fitness function: the reduction in their supposed stay time. This approach is simple and intuitive, but it makes difficult the interpretation of expected walking time percentage. Indeed the variation on walking time becomes from path variations, but the total tour time is also changed on account of the technical issue about the GA fitness function.

Moreover, the effects of serendipitous items on expected walking time are analyzed with respect to the starting optimized tours (S0), i.e. the previously discussed drawback is partially cut off. Table III shows that few disturbs cause

	T1	T2	T3	T4	T5	
S0	39.9	34.0	34.6	31.6	30.2	**34.1**
S1	42.6	36.3	36.0	32.8	31.3	**35.8**
S2	45.0	38.1	37.4	34.0	32.2	**37.4**
S3	49.7	40.1	38.3	34.5	33.5	**39.2**
S4	52.7	42.0	39.9	36.3	34.6	**41.1**
S5	56.0	45.5	41.9	37.8	35.9	**43.4**
S6	60.0	47.5	43.7	39.7	37.2	**45.6**
S7	65.2	51.7	45.6	41.7	39.0	**48.6**

Table II
PERCENTAGES OF WALKING TIME

a quite uniform increase of the walking time percentage: the ground becomes from the slight deviations on S0 tour. On the other hand, growing the number of serendipitous items, the deviations are amplified. This is more evident for the shortest S0 tours, since many serendipitous items can encourage the "exploration" of rooms untouched by S0, about Figure 5.

	T1	T2	T3	T4	T5
S1	106	106	104	103	103
S2	112	112	108	107	107
S3	124	119	112	110	111
S4	131	126	117	115	115
S5	141	136	123	121	120
S6	150	143	130	127	125
S7	164	155	135	134	131

Table III
INCREMENT OF WALKING TIME FOR TOURS WITH SERENDIPITOUS ITEMS

V. CONCLUSIONS AND FUTURE WORK

This paper presents a beginning effort to apply some ideas about serendipity to information retrieval and information filtering systems, especially in recommenders, to mitigate the over-specialization issue. The museum scenario is particularly interesting because items are arranged in a physical space and users interact with the environment. Thus disregarding context facets makes useless recommendations.

Similar remarks are still valid in domains (different from cultural heritage fruition) in witch a physical or virtual space is involved and it represents a pragmatic justification to explain (supposed) serendipitous recommendations.

As future work, we expect to carry out more extensive experimentation with more users and wider item collections. We plan also to gather user feedback and feeling by questionnaires focused on qualitative evaluation of the recommendations and the idea of getting suggestions that should surprise them. That is really important for the need to understand the effectiveness of the module in finding unknown items rather the ones that result best rated. Experimentation with users with different cultural levels and with different information seeking tasks are also important to find out which kind of user would like most serendipitous recommendations and to whom they are more useful.

We expect also to implement the other suggestions given by Toms [7] and to develop further the heuristic proposed (maybe padding a parameter factor that multiplies the probabilities in order to balance better between categories) or also introduce new heuristics and make an experimental comparison.

REFERENCES

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

[2] J. Herlocker, J. Konstan, L. Terveen, and J. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, 2004.

[3] S. M. McNee, J. Riedl, and J. Konstan, "Being accurate is not enough: how accuracy metrics have hurt recommender systems," in *the 2006 Conference on Human Factors in Computing Systems (CHI 2006)*, G. M. Olson and R. Jeffries, Eds. Montral: ACM, 2006, pp. 1097–1101.

[4] R. M. Roberts, *Serendipity: Accidental Discoveries in Science*. New York: John Wiley & Sons, Inc, 1989.

[5] A. Foster and N. Ford, "Serendipity and information seeking: an empirical study," *Journal of Documentation*, vol. 59, no. 3, pp. 321–340, 2003.

[6] J. Campos and A. de Figueiredo, "Searching the unsearchable: Inducing serendipitous insights," in *the Workshop Program at the 4th International Conference on Case-Based Reasoning (ICCBR 2001)*, 2001, pp. 159–164.

[7] E. Toms G., "Serendipitous information retrieval," in *DELOS Workshop: Information Seeking, Searching and Querying in Digital Libraries*, Zurich - Switzerland, 2000.

[8] M. Degemmis, P. Lops, and G. Semeraro, "A content-collaborative recommender that exploits wordnet-based user profiles for neighborhood formation," *User Modeling and User-Adapted Interaction: The Journal of Personalization Research (UMUAI)*, vol. 17, no. 3, pp. 217–255, 2007.

[9] G. Semeraro, M. Degemmis, P. Lops, and P. Basile, "Combining learning and word sense disambiguation for intelligent user profiling," in *the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007)*, M. M. Veloso, Ed., Hyderabad - India, 2007, pp. 2856–2861.

[10] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[11] P. Basile, M. Degemmis, A. L. Gentile, P. Lops, and G. Semeraro, "Uniba: Jigsaw algorithm for word sense disambiguation," in *the 4th ACL 2007 International Worshop on Semantic Evaluations (SemEval-2007)*, Prague, 2007, pp. 398–401.

[12] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.

[13] B. Shapira, P. Shoval, and U. Hanani, "Stereotypes in information filtering systems," *Information Processing and Management*, vol. 33, no. 3, pp. 273–287, 1997.