The Conference Participant Advisor Service in a Virtual Information and Knowledge Environment Framework

M. Degemmis, P. Lops, P. Basile, and G. Semeraro

Dipartimento di Informatica
Università di Bari, Bari 70126, Italy
{degemmis,lops,basile,semeraro}@di.uniba.it

Abstract. Algorithms designed to support users in retrieving relevant information base their relevance computations on *user profiles*, in which representations of the users interests are maintained. The idea proposed in this paper is the integration of general linguistic knowledge in the process of learning *semantic* user profiles able to represent users' interests in a more effective way with respect to classical keyword-based profiles. Semantic profiles are obtained by integrating a naïve Bayes approach for text categorization with a word sense disambiguation strategy based exclusively on the lexical knowledge stored in the WordNet database. Semantic profiles are exploited by the "conference participant advisor" service developed in the VIKEF (Virtual Information and Knowledge Environment Framework) project in order to suggest papers to be read and talks to be attended by a conference participant. Experiments carried out on a dataset made of papers accepted to the previous editions of the International Semantic Web Conference and rated by real users show the effectiveness of the service.
Keywords: user profiling, text categorization, word sense disambiguation, WordNet.

1 Introduction

There has been a growing interest in augmenting traditional information filtering and retrieval approaches with machine learning techniques, that induce a structured model of the interests of a user, the *user profile*, from text documents [12]. These methods typically require users to label documents by assigning a relevance score, and automatically infer profiles exploited in the filtering/retrieval process to rank documents according to the user preferences. There are information access scenarios that cannot be solved through straightforward matching of queries and documents represented by keywords. For example, a researcher interested in retrieving "interesting scientific papers" cannot easily express this form of information need as a query suitable for search engines. In order to find relevant information in these problematic information scenarios, a possible solution could be to develop methods for discovering concepts that characterize documents the user has already labelled as relevant. Traditional keyword-based

approaches are unable to capture the *semantics* of the user interests. They are primarily driven by a string-matching operation: If a string, or some morphological variant, is found in both the profile and the document, a match is made and the document is considered as relevant. String matching suffers from problems of POLYSEMY, the presence of multiple meanings for one word, and SYNONYMY, multiple words having the same meaning.

The result is that, due to synonymy, relevant information can be missed if the profile does not contain the exact keywords in the documents while, due to polysemy, wrong documents could be deemed as relevant. These problems call for alternative methods able to learn more accurate profiles that capture concepts expressing users' interests from relevant documents. These *semantic* profiles will contain references to concepts defined in lexicons or, in a further step, ontologies. Although they clearly require additional knowledge and processing, methods for learning *semantic* profiles have potentially a number of advantages: For example, if a user likes documents about *robotics* and *machine learning*, a method with the ability to identify these concepts and to have access to the proper concept hierarchy could infer that the user is interested in *artificial intelligence*. Not only would this be a natural suggestion to the user, but it might also be useful in quickly capturing his/her real preferences and suggesting what additional information might be of interest. Moreover, the descriptions of the identified key concepts could help make the profile more intelligible to the user, which in turn could help establish trust. We propose a method to learn *semantic user profiles* by integrating a WSD algorithm based on WordNet [10, 3] with a naïve bayes method for text categorization.

The paper is organized as follows: After a short discussion about the main works related to our research, we describe in Section 3 our method to learn *semantic user profiles* obtained by integrating a Word Sense Disambiguation (WSD) algorithm based on WordNet [10] with a naïve bayes method for text categorization. A possible application scenario for semantic profiles is given in Section 4, which presents "Conference Participant Advisor", a service that supports participants to a conference in planning their attendance. Conclusions and future work are drawn in Section 5.

2 Related Work

Our work was mainly inspired by:

- *Syskill & Webert* [14], that suggests to learn user profiles as Bayesian classifiers;
- *LIBRA* [13], that adopts a Bayesian classifier to produce content-based book recommendations by exploiting product descriptions obtained from the Web pages of the Amazon on-line digital store. Documents are represented by using keywords and are subdivided into slots, each one corresponding to a specific section of the document (authors, title, astract ...);

- *SiteIF* [7], which exploits a sense-based representation to build a user profile as a semantic network whose nodes represent senses of the words in documents requested by the user;
- *OntoSeek* [4], a system designed for content-based information retrieval from online yellow pages and product catalogs which explored the role of linguistic ontologies in knowledge-retrieval systems. OntoSeek combines an ontology-driven content-matching mechanism based on WordNet with a moderately expressive representation formalism. The approach has shown that structured content representations coupled with linguistic ontologies can increase both recall and precision of content-based retrieval.

According to these successful works, we conceived our ITem Recommender system as a text classifier able 1) to deal with a sense-based document representation obtained by exploiting a linguistic ontology and 2) to learn a bayesian profile from documents subdivided into slots. The strategy we propose to shift from a keyword-based document representation to a sense-based document representation is *to integrate lexical knowledge in the indexing step of training documents*. Several methods have been proposed to accomplish this task. Scott and Matwin [15] proposed to include WordNet information at the feature level by expanding each word in the training set with *all* the synonyms for it in WordNet, including those available for each sense, in order to avoid a WSD process. This approach has shown a decrease of effectiveness in the obtained classifier, mostly due to the word ambiguity problem. The work by Scott and Matwin suggests that some kind of disambiguation is required. Subsequent works tried to investigate whether embedding WSD in document classification tasks improves classification accuracy. Bloedhorn and Hotho [1] compared three strategies to map words to senses: No WSD, most frequent sense as provided by WordNet, WSD based on context. They found positive results on the Reuters 25178, the OSHUMED and the FAODOC corpus. In [18], a WSD algorithm based on the general concept of Extended Gloss Overlaps is used and classification is performed by a Support Vector Machine classifier applied to the two largest categories of the Reuters 25178 corpus and two Internet Movie Database movie genres[1]. The relevant outcome of this work is that, when the training set is small, the use of WordNet senses combined with words improves the performance of the classifier.

3 Learning User Profiles as a Text Categorization Problem

The machine learning techniques generally used in the task of inducing content-based profiles are those that are well-suited for text categorization [16]. In the machine learning approach to text categorization, an inductive process automatically builds a text classifier by learning from a set of *training documents* (documents labeled with the categories they belong to), the features of the categories. We consider the problem of learning user profiles as a binary text categorization

[1] www.imdb.com

task: Each document has to be classified as interesting or not with respect to the user preferences. Therefore, the set of categories is $C = \{c_+, c_-\}$, where c_+ is the positive class (user-likes) and c_- the negative one (user-dislikes). We present a method able to learn profiles for content-based filtering. The accuracy of the keyword-based profiles inferred by this method will be compared to semantic user profiles obtained by the same method, but exploiting an indexing procedure based on WordNet. In this paper, we first present a content-based profiling system named ITem Recommender (ITR), able to induce user profiles as a naïve Bayesian classifier [11]. Then, we describe an intelligent service able to support a conference participant in planning his talk attendance, built upon this system.

3.1 Document Representation

In the classical *bag of words* (BOW) model, each feature used to represent a document corresponds to a single word found in the document [16]. We extend the classical BOW model to a model in which the senses corresponding to the words in the documents are considered as features. This sense-based document representation can be exploited by the learning algorithm to build *semantic user profiles*. Here "word sense" is used as a synonym of "word meaning". The filtering phase could take advantage of the word senses to recommend new items (documents) with high semantic relevance with respect to the user profile. There are two crucial issues to address: First, a repository for word senses has to be identified. Second, any implementation of sense-based text classification must solve the problem that, while words occur in a document, meanings do not, since they are often hidden in the context. Therefore, a procedure is needed for assigning senses to words. This task is known as Word Sense Disambiguation (WSD) and consists in determining which of the senses of an ambiguous word is invoked in a particular use of the word [8].

As for sense repository, we have adopted WordNet (version 1.7.1), a large lexical database for English, which is freely available online[2] and has been extensively used in NLP research [17]. WordNet was designed to establish connections between four types of Parts of Speech (POS): Noun, verb, adjective, and adverb. The basic building block for WordNet is the SYNSET (SYNonym SET), which represents a specific meaning of a word. The specific meaning of one word under one type of POS is called a sense. Synsets are equivalent to senses, which are structures containing sets of words with synonymous meanings (words that are interchangeable in some contexts). Each synset has a gloss, a short textual description that defines the concept represented by the synset. For example, the words *night*, *nighttime* and *dark* constitute a single synset that has the following gloss: "the time after sunset and before sunrise while it is dark outside".

Synsets are connected through a series of relations: Antonymy (opposites), hyponymy/hypernymy (IS-A), meronymy (PART-OF), etc. We addressed the WSD problem by proposing an algorithm based on semantic similarity between synsets

[2] `http://wordnet.princeton.edu`

computed by exploiting the hyponymy relation, which serves to form the lexicon into a hierarchical structure.

The WSD procedure is fundamental to obtain a synset-based vector space representation that we called Bag-Of-Synsets (BOS). In this model, a synset vector corresponds to a document, instead of a word vector. Another key feature of the approach is that each document is represented by a set of *slots*, where each slot is a textual field corresponding to a specific feature of the document, in an attempt to take into account also the structure of documents. For example, in our application scenario, in which documents are scientific papers, we selected three slots:

1. *title*, the title of the paper;
2. *authors*, the list of the names of the authors;
3. *abstract*, the short text that presents the main points of the paper;

The text in each slot is represented according to the BOS model by counting separately the occurrences of a synset in the slots in which it appears. More formally, assume that we have a collection of N documents. Let m be the index of the slot, for $n = 1, 2, ..., N$, the n-th document is reduced to three bags of synsets, one for each slot:

$$d_n^m = \langle t_{n1}^m, t_{n2}^m, \ldots, t_{nD_{nm}}^m \rangle$$

where t_{nk}^m is the k-th synset in slot s_m of document d_n and D_{nm} is the total number of synsets appearing in the m-th slot of document d_n. For all n, k and m, $t_{nk}^m \in V_m$, which is the vocabulary for the slot s_m (the set of all different synsets found in slot s_m). Document d_n is finally represented in the vector space by three synset-frequency vectors:

$$f_n^m = \langle w_{n1}^m, w_{n2}^m, \ldots, w_{nD_{nm}}^m \rangle$$

where w_{nk}^m is the weight of the synset t_{nk}^m in the slot s_m of document d_n and can be computed in different ways: It can be simply the number of times synset t_k appears in slot s_m or a more complex TF-IDF score.

3.2 A WordNet-based algorithm for WSD

The goal of a WSD algorithm is to associate the appropriate meaning or sense s to a word w in document d, by exploiting its *window of context* (or more simply *context*) C, that is a set of words that precede and follow w. The sense s is selected from a predefined set of possibilities, usually known as *sense inventory*. In the proposed algorithm, the sense inventory is obtained from WordNet. For example, let us consider the document d: "The white cat is hunting the mouse". The text in d is processed by two basic phases:

1. tokenization, stopword elimination, part-of-speech tagging (POS) and lemmatization;
2. synset identification by means of WSD.

Figure 1 shows how d is represented in each substep in the first phase. The original sentence (1) is tokenized and, for each token, part of speech ambiguities are solved (2). Reduction to lemmas (3)(for example, verbs are turned to their base form) is performed before deleting stopwords (4).

```
The    white    cat    is     hunting     the    mouse    (1)
The/DT white/JJ cat/NN is/VBZ hunting/VBG the/DT mouse/NN (2)
The/DT white/JJ cat/NN be/VB  hunt/VB     the/DT mouse/NN (3)
       white/JJ cat/NN        hunt/VB             mouse/NN (4)
```

Fig. 1. The preprocessing of sentence "The white cat is hunting the mouse". Each token is labeled with a tag describing its lexical role in the sentence. NN=noun, singular - VB=verb, base form - VBZ=verb, is - VBG=verb, gerund form - JJ=adjective, DT=determinative.

As for lemmatization and part-of-speech tagging we use the MontyLingua natural language processor[3] for English. After step (4) in Figure 1, document d is ready for the second phase of synset identification through WSD. The core idea behind the proposed WSD algorithm is to disambiguate w by determining the degree of *semantic similarity* among candidate synsets for w and those of each word in C. Thus, the proper synset assigned to w is that with the highest similarity with respect to its context of use.

Several measures of similarity or relatedness have been proposed to determine the degree of semantic similarity between two words based on their relative position in a concept hierarchy like WordNet [2]. The measure of semantic similarity adopted in our work is the Leacock-Chodorow measure [6], which is based on the length of the path between concepts in an IS-A hierarchy. The idea behind this measure is that similarity between synsets a and b is inversely proportional to the distance between them in the WordNet *is-a* hierarchy, measured by the number of nodes in the shortest path (the path having minimum number of nodes) from a to b . The similarity is computed in algorithm 1 by the function `SinSim` (lines 24-28): the path length N_p is scaled by the depth D of the hierarchy, where depth is defined as the length of the longest path from a leaf node to the root node of the hierarchy. The proposed WSD procedure is described by using the sentence *"The white cat is hunting the mouse"* as example. Let w="cat" be the word to be disambiguated. The procedure starts by defining the context C of w as the set of words in the same slot of w having the same POS as w. In this case, the only *noun* in the sentence is "mouse", then $C = \{mouse\}$. Next, the algorithm identifies both the sense inventory for w, that is X_{cat} = {`01789046: feline mammal, 00683044: computerized axial tomography,`...}, and the sense inventory X_j for each word w_j in C. Thus, X_{mouse}= {`01993048: small rodents, 03304722: a hand-operated electronic device that controls the coordinates`

[3] http://web.media.mit.edu/hugo/montylingua

`of a cursor, ...` }. The sense inventory T for the whole context C is given by the union of all X_j (in this case, $X_j = T$, since C consists of a single word). After this step, we measure the similarity of each candidate sense $s_i \in X_w$ to that of each sense $s_h \in T$ and then the sense assigned to w is the one with the highest similarity score. In the example, `SinSim(01789046: feline mammal, 01993048: small rodents) = 0.806` is the highest similarity score, thus w is interpreted as "feline mammal".

Algorithm 1 The WordNet-based WSD algorithm

procedure WSD(w, d) ▷ finds the proper synset of a polysemous word w in document d
  $C \leftarrow \{w_1, ..., w_n\}$ ▷ C is the context of w. For example, $C = \{w_1, w_2, w_3, w_4\}$ is a window with radius=2, if the sequence of words $\{w_1, w_2, w, w_3, w_4\}$ appears in d
  $X_w \leftarrow \{s_1, ...s_k\}$ ▷ X_w is sense inventory for w, that is the set of all candidate synsets for w returned by WordNet
  $s \leftarrow null$ ▷ s is the synset to be returned
  $score \leftarrow 0$ ▷ $score$ is the similarity score assigned to s wrt to the context C
  $T \leftarrow \emptyset$ ▷ T is the set of all candidate synsets for all words in C
  for all $w_j \in C$ **do**
    if $POS(w_j) = POS(w)$ **then** ▷ $POS(y)$ is the part-of-speech of y
      $X_j \leftarrow \{s_{j1}, ...s_{jm}\}$ ▷ X_j is the set of m possible senses for w_j
      $T \leftarrow T \cup X_j$
    end if
  end for
  for all $s_i \in X_w$ **do**
    for all $s_h \in T$ **do**
      $score_{ih} \leftarrow$ SinSim(s_i, s_h) ▷ computing similarity scores between s_i and every synset $s_h \in T$
      if $score_{ih} \geq score$ **then**
        $score \leftarrow score_{ih}$
        $s \leftarrow s_i$ ▷ s is the synset $s_i \in X_w$ having the highest similarity score wrt the synsets in T
      end if
    end for
  end for
  return s
end procedure
function SinSim(a, b) ▷ The similarity of the synsets a and b
  $N_p \leftarrow$the number of nodes in path p from a to b
  $D \leftarrow$maximum depth of the taxonomy ▷ $D = 16$ in WordNet 1.7.1
  $r \leftarrow -log(N_p/2D)$
  return r
end function

Each document is mapped into a list of WordNet synsets by following three steps:

1. each monosemous word w in a slot of a document d is mapped into the corresponding WordNet synset;
2. for each pair of words $\langle$*noun,noun*$\rangle$ or $\langle$*adjective,noun*$\rangle$, a search in WordNet is made to verify if at least one synset exists for the bigram $\langle w_1, w_2 \rangle$. In the positive case, algorithm 1 is applied on the bigram, otherwise it is applied separately on w_1 and w_2; in both cases all words in the slot are used as the context C of the word(s) to be disambiguated;
3. each polysemous unigram w is disambiguated by algorithm 1, using all words in the slot as the context C of w.

Our hypothesis is that the proposed indexing procedure helps to obtain profiles able to recommend documents semantically closer to the user interests. The difference with respect to keyword-based profiles is that synset unique identifiers are used instead of words.

3.3 A Naïve Bayes Method for User Profiling

Naïve Bayes is a probabilistic approach to inductive learning. The learned probabilistic model estimates the *a posteriori* probability, $P(c_j|d_i)$, of document d_i belonging to class c_j. To classify a document d_i, the class with the highest probability is selected. As a working model for the naïve Bayes classifier, we use the multinomial event model [9]:

$$P(c_j|d_i) = P(c_j) \prod_{w \in V_{d_i}} P(t_k|c_j)^{N(d_i,t_k)} \tag{1}$$

where $N(d_i, t_k)$ is defined as the number of times word or token t_k appears in document d_i. Notice that rather than getting the product of all distinct words in the corpus V we only use the subset of the vocabulary V_{d_i} containing the words that occur in the document d_i.

Since each document is encoded as a vector of BOS, one for each slot, Equation (1) becomes:

$$P(c_j|d_i) = \frac{P(c_j)}{P(d_i)} \prod_{m=1}^{|S|} \prod_{k=1}^{|b_{im}|} P(t_k|c_j, s_m)^{n_{kim}} \tag{2}$$

where $S = \{s_1, s_2, \ldots, s_{|S|}\}$ is the set of slots, b_{im} is the BOS in the slot s_m of the document d_i, n_{kim} is the number of occurrences of the synset t_k in b_{im}. ITR implements this approach to classify documents as interesting or uninteresting for a particular user.

To calculate (2), we only need to estimate $P(c_j)$ and $P(t_k|c_j, s_m)$ in the training phase of the system.

The documents used to train the system belong to a collection consisting of all the scientific papers accepted to the 2002-2004 editions of the International Semantic Web Conference (ISWC). Ratings on these documents, obtained from real users, were recorded on a discrete scale from 1 to 5 (see Section 4 for a

detailed description of the dataset). An instance labeled with a rating r, $r = 1$ or $r = 2$ belongs to class c_- (user-dislikes); if $r = 4$ or $r = 5$ then the instance belongs to class c_+ (user-likes); rating $r = 3$ is neutral. Each rating was normalized to obtain values ranging between 0 and 1:

$$w_+^i = \frac{r-1}{MAX-1}; \qquad w_-^i = 1 - w_+^i \tag{3}$$

where MAX is the maximum rating that can be assigned to an instance.

The weights in (3) are used for weighting the occurrences of a word in a document and to estimate the probability terms from the training set TR. The prior probabilities of the classes are computed according to the following equation:

$$\hat{P}(c_j) = \frac{\sum_{i=1}^{|TR|} w_j^i + 1}{|TR| + 2} \tag{4}$$

Witten-Bell smoothing [19] has been adopted to compute $P(t_k|c_j, s_m)$, by taking into account that documents are structured into slots and that word occurrences are weighted using weights in equation (3):

$$P(t_k|c_j, s_m) = \begin{cases} \frac{N(t_k,c_j,s_m)}{V_{c_j}+\sum_i N(t_i,c_j,s_m)} & \text{if } N(t_k, c_j, s_m) \neq 0 \\ \frac{V_{c_j}}{V_{c_j}+\sum_i N(t_i,c_j,s_m)} \frac{1}{V-V_{c_j}} & \text{if } N(t_k, c_j, s_m) = 0 \end{cases} \tag{5}$$

where $N(t_k, c_j, s_m)$ is the count of the weighted occurrences of the word t_k in the training data for class c_j in the slot s_m, V_{c_j} is the total number of unique words in class c_j, and V is the total number of unique words across all classes. $N(t_k, c_j, s_m)$ is computed as follows:

$$N(t_k, c_j, s_m) = \sum_{i=1}^{|TR|} w_j^i n_{kim} \tag{6}$$

In (6), n_{kim} is the number of occurrences of the term t_k in the slot s_m of the i^{th} document. The sum of all $N(t_k, c_j, s_m)$ in the denominator of equation (5) denotes the total weighted length of the slot s_m in the class c_j. In other words, $\hat{P}(t_k|c_j, s_m)$ is estimated as a ratio between the weighted occurrences of the term t_k in slot s_m of class c_j and the total weighted length of the slot. The final outcome of the learning process is a probabilistic model used to classify a new instance in the class c_+ or c_-. The model can be used to build a personal profile that includes those words that turn out to be most indicative of the user's preferences, according to the value of the conditional probabilities in (5).

4 The "Conference Participant Advisor" Service in the VIKEF Project

VIKEF (Virtual Information and Knowledge Environment Framework)[4] is an application-oriented Integrated Project. VIKEF is dedicated to advanced semantic-enabled support for ICK (Information, Content, and Knowledge) production, acquisition, processing, annotation, sharing and use by empowering information and knowledge environments for scientific and business communities. The VIKEF results are integrated into an open and flexible advanced software framework, the VIKE-Framework, which will enable new forms of content use and community support in sectors like community events (e.g. trade fairs and congresses) and scientific publishing. The benefit of the framework will be made measurable by building a representative application solution for the VIKEF crystallization application domains, trade fairs and scientific congresses. In this section, we present a service realized in the context of the applications to support scientific congress organization.

4.1 Description of the service

The main target of personalization in a virtual information and knowledge environment is the reduction of information overload. The user modeling and tracking activities provide the basis for a wide range of services that reuse the semantic information describing properties of the user. Such personalization services contribute to a more targeted information access and dissemination. Typical applications of user profiles in an information environment are information filtering, personalized information recommendations, and targeted notification about changes in the information space. Among different recommendation techniques proposed in the literature, the collaborative filtering approach is the most successful and widely adopted to date. The success of a collaborative filtering approach depends on the availability of a critical mass of users: it is not wise to use collaborative filtering in domains with a few number of users, and where personalization can be based on the analysis of the content that is interesting and not to users. The collaborative filtering approach is not the optimal choice in VIKEF because in both the scenarios (trade fairs, scientific congress) there is not a critical mass of users able to provide some feedback on the items. Moreover, the idea underlying the Semantic Web is to use a model in which the knowledge is explicitly represented: Collaborative filtering implements a black-box model in which users do not have any explanation of the rules the system used for producing recommendations. Users must trust information produced by the system, and this is not possible if they are not able to understand the reasoning process used by the system to produce recommendations. On the other hand, content-based methods must be extended in order to introduce specific methodologies for semantic analysis of content. In the proposed approach, semantic profiles are used to plan the conference visit. The "Conference Advisor Service" aims to show the

[4] `www.vikef.net`

potential of semantic tools integrated in ITR, providing useful services for conference participation planning. The prototype has been realized in the context of the "International Semantic Web Conference 2004", by adding to the conference homepage (it is a local copy of the official web site) a login/registration form to access recommendation services. The conference participant can register providing a valid email address and can browse the whole document repository or search for papers presented during 2002 and 2003 ISWC events, in order to provide ratings. The user could specify in which slots (different parts of a document) the search should be performed. Each retrieved paper can be rated and, given a sufficient number of ratings, the system builds the participant profile (at present the threshold is 20). ISWC 2004 papers are classified using the learned profile to obtain a personalized list of recommended papers and talks which is sent by email to the participant. Recommended talks are highlighted in yellow in the personalized electronic program (Fig. 2).

TECHNICAL PROGRAM		
Tuesday 9th		
Opening Ceremony	09.00-09.30	[Room 1]
Invited Speaker: Edward Feigenbaum	09.30-10.30 [chair: F. van Harmelen]	[Room 1]
The Semantic Web Story -- It's already 2004. Where are we?		
Coffee Break	10.30-11.00	[Room 4]
Session 1:Integration and Interoperability	11.00-12.30 [chair: M.C. Rousset]	[Room 1]
An Extensible Directory Enabling Efficient Semantic Web Service Integration	Ion Constantinescu, Walter Binder, Boi Faltings	
Opening Up Magpie via Semantic Services	Martin Dzbor, Enrico Motta, John Domingue	
Working with Multiple Ontologies on the Semantic Web	Bernardo Cuenca Grau, Bijan Parsia, Evren Sirin	
Session 2: Searching and Querying	11.00-12.30 [chair: R. Studer]	[Room 2]
A Comparison of RDF Query Languages	Peter Haase, Jeen Broekstra, Andreas Eberhart,Raphael Volz	
Generating On the Fly Queries for the Semantic Web: The ICS-FORTH Graphical RQL Interface (GRQL)	Nikos Athanasis, Vassilis Christophides, Dimitris Kotzinos	
Information Retrieval Support for Ontology Construction and Use	Willem van Hage, Maarten de Rijke, Maarten Marx	
Lunch	12.30-14.00	

Fig. 2. The personalized program sent to the user

4.2 Experimental Session

An experimental evaluation of semantic profiles is carried out on ISWC papers. The goal of the evaluation is to estimate if the BOS version of ITR improves the performance with respect to the BOW one. Experiments were carried out on a collection of 100 papers (42 papers accepted to ISWC 2002, 58 papers accepted to ISWC 2003) rated by 11 real users, that we called *ISWC dataset*. Papers are rated on a 5-point scale mapped linearly to the interval [0,1]. Tokenization, stopword elimination and stemming have been applied to obtain the BOW. Documents indexed by the BOS model have been processed by WSD

procedure, obtaining a 14% feature reduction (20,016 words vs. 18,627 synsets - see Table 1). This is mainly due to the fact that bigrams are represented using only one synset and that synonym words are represented by the same synset. Classification effectiveness was evaluated by the classical measures *precision*, *recall* [16]. We adopted also the Normalized Distance-based Performance Measure (NDPM) [20] to measure the distance between the ranking imposed on papers by the user ratings and the ranking predicted by ITR, that ranks papers according to the a-posteriori probability of the class *likes*. Values range from 0 (agreement) to 1 (disagreement). In the experiments, a paper is considered *relevant* if the user provided a rating for it greater or equal than 3, while ITR considers an item as relevant if $P(c_+|d_i) \geq 0.5$, computed as in equation (2). We executed one experiment for each user. Each experiment consisted in 1) selecting the ratings of the user and the papers rated by that user; 2) splitting the selected data into a training set *Tr* and a test set *Ts*; 3) using *Tr* for learning the corresponding user profile; 4) evaluating the predictive accuracy of the induced profile on *Ts*, using the aforementioned measures. The methodology adopted for obtaining *Tr* and *Ts* was the 5-fold cross validation [5]. The results of the comparison between the profiles obtained from documents represented using the two indexing approaches, namely BOW and BOS, are reported in Table 2. We can notice an improvement both in precision (+1%) and recall (+2%). The BOS model outperforms the BOW model specifically for users 7 and 10. By the way, the general indication is that it is difficult to reach a strong improvement both in precision and recall by using the BOS model. Even if a higher level of precision is reached (users 5 and 7), recall has not been improved. Only on user 10 we observed a general improvement of both measures. NDPM has not been improved, but it remains acceptable. It could be noticed from the NDPM values that the relevant/not relevant classification is improved without improving the ranking. The general conclusion is that the BOS method has improved the classification of

Table 1. The ISWC dataset used in the experiments

Id user	Rated Papers	% POS	% NEG	n. words	n. synsets
1	37	59	41	2,702	2,546
2	22	54	46	1,597	1,506
3	27	63	37	1,929	1,792
4	27	44	56	1,830	1,670
5	29	59	41	2,019	1,896
6	22	82	18	1,554	1,433
7	26	58	42	1,734	1,611
8	28	61	39	2,034	1,901
9	23	57	43	1,442	1,374
10	22	59	41	1,335	1,258
11	25	48	52	1,740	1,640
	288	59	41	20,016	18,627

Table 2. Performance of the BOW - BOS profiles

Id User	**Precision**		**Recall**		**NDPM**	
	ITR BOW	ITR BOS	ITR BOW	ITR BOS	ITR BOW	ITR BOS
1	0.57	0.55	0.47	0.50	0.60	0.56
2	0.73	0.55	0.70	0.83	0.43	0.46
3	0.60	0.57	0.35	0.35	0.55	0.59
4	0.60	0.53	0.30	0.43	0.47	0.47
5	0.58	0.67	0.65	0.53	0.39	0.59
6	0.93	0.96	0.83	0.83	0.46	0.36
7	0.55	0.90	0.60	0.60	0.45	0.48
8	0.74	0.65	0.63	0.62	0.37	0.33
9	0.60	0.54	0.63	0.73	0.31	0.27
10	0.50	0.70	0.37	0.50	0.51	0.48
11	0.55	0.45	0.83	0.70	0.38	0.33
Mean	0.63	0.64	0.58	0.60	0.45	0.45

items whose score (and ratings) is close to the relevant/not relevant threshold, thus items for which the classification is highly uncertain (thus minor changes in the ranking have not modified the NDPM values). A Wilcoxon signed ranked test, requiring a significance level $p < 0.05$, has been performed in order to validate these results. We considered each dataset as a single trial for the test. Thus, 11 trials have been executed. The test confirmed that there is a statistically significant difference in favor of the BOS model with respect to the BOW model only as regards recall, but not precision.

5 Conclusions and Future Work

We presented a system exploiting a Bayesian learning method to induce *semantic* user profiles from documents represented using WordNet synsets. Our hypothesis is that replacing words with synsets in the indexing phase produces a more accurate document representation that could be successfully exploited to learn more accurate user profiles. Semantic profiles are used in the context of planning the talk to be followed and the papers to be read by a conference participant. Our hypothesis is confirmed by the experiments conducted in order to evaluate the effectiveness of the service and can be explained by the fact that synset-based classification allows the preference of documents with a high degree of semantic coherence, not guaranteed in case of word-based classification. As a future work, we plan to exploit not only the WordNet hierarchy but also domain ontologies in order to realize a more powerful document indexing.

Acknowledgments

This research was partially funded by the European Commission under the 6^{th} Framework Programme IST Integrated Project VIKEF No. 507173, Priority 2.3.1.7 Semantic Based Knowledge Systems.

References

[1] S. Bloedhorn and A. Hotho. Boosting for text classification with semantic features. In *Proceedings of 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Mining for and from the Semantic Web Workshop*, pages 70–87, 2004.

[2] A. Budanitsky and G. Hirst. Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures. In *Proceedings of the Workshop on WordNet and other Lexical Resources, Second Meeting of the North American Chapter of the Association for Computational Linguistics*, 2001.

[3] C. Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[4] N. Guarino, C. Masolo, and G. Vetere. Content-based access to the web. *IEEE Intelligent Systems*, 14(3):70–80, 1999.

[5] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 1137–1145. San Mateo, CA: Morgan Kaufmann, 1995.

[6] C. Leacock and M. Chodorow. Combining local context and wordnet similarity for word sense identification. In C. Fellbaum, editor, *WordNet: An Electronic Lexical Database*, pages 305–332. MIT Press, 1998.

[7] B. Magnini and C. Strapparava. Improving user modelling with content-based techniques. In *Proc. 8th Int. Conf. User Modeling*, pages 74–83. Springer, 2001.

[8] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*, chapter 16: Text Categorization, pages 575–608. The MIT Press, Cambridge, US, 1999.

[9] A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification. In *Proceedings of the AAAI/ICML-98 Workshop on Learning for Text Categorization*, pages 41–48. AAAI Press, 1998.

[10] G. Miller. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3(4), 1990. (Special Issue).

[11] T. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.

[12] D. Mladenic. Text-learning and related intelligent agents: a survey. *IEEE Intelligent Systems*, 14(4):44–54, 1999.

[13] R. J. Mooney and L. Roy. Content-based book recommending using learning for text categorization. In *Proceedings of the 5^{th} ACM Conference on Digital Libraries*, pages 195–204, San Antonio, US, 2000. ACM Press, New York, US.

[14] M. Pazzani and D. Billsus. Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27(3):313–331, 1997.

[15] S. Scott and S. Matwin. Text classification using wordnet hypernyms. In *COLING-ACL Workshop on usage of WordNet in NLP Systems*, pages 45–51, 1998.

[16] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1), 2002.

[17] M. Stevenson. *Word Sense Disambiguation: The Case for Combinations of Knowledge Sources*. CSLI Publications, Stanford, CA, USA, 2003.

[18] M. Theobald, R. Schenkel, and G. Weikum. Exploting structure, annotation, and ontological knowledge for automatic classification of xml data. In *Proceedings of International Workshop on Web and Databases*, pages 1–6, 2004.

[19] I. Witten and T. Bell. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory*, 37(4), 1991.

[20] Y. Y. Yao. Measuring retrieval effectiveness based on user preference of documents. *Journal of the American Society for Information Science*, 46(2):133–145, 1995.