

# Rule- and OWA-based Semantic Similarity for User Profiling

Marek Z. Reformat and Seyed Koosha Golmohammadi

## Abstract

The ultimate goal of many Internet-related research activities is to “transform” the web into a user-friendly environment where users can easily find things they are looking for. In many cases, this translates into a task of finding information useful for a user in a quick and efficient way. In the growing amount of information available on the Internet this errand becomes a challenge. Systems targeting information retrieval activities on the web are being equipped with user profiling. User profiles represent users’ interests, and are used as a means to support extraction of relevant information. Continuous updating of profiles following users’ activities on the web is a very important aspect of the profiling.

In this paper, we propose a new method for updating a user profile. The method is based on analyzing user’s browsing behavior, and identifying the most relevant items that should be added to the profile. This process uses a semantic-based similarity measure. This measure is estimated using rules representing multiple facets of similarity. The rules are constructed using a domain ontology. Obtained similarities are aggregated using Ordered Weighted Averaging (OWA) operator. This allows for expressing different levels of “strictness” in estimating similarity, and utilization of linguistic quantifiers OR, SOME, and MOST for that purpose. The semantic similarity is then combined with the items’ importance measures in order to identify items that are of the highest relevance to the user interests. The proposed approach is used for updating a profile in the music domain. The details of the real-word experiment are described in the paper.

**Keywords:** *user profile, ontology, semantic similarity, if-then rules, ordered weighted averaging.*

## 1. Introduction

---

Corresponding Author: Marek Z. Reformat is with the Department of Electrical and Computer Engineering, University of Alberta, ECERF Building W6-023, Edmonton, T6G 2V4, Canada.

E-mail: Marek.Reformat@ualberta.ca

Manuscript received 13 Nov. 2009; revised 20 Mar. 2010; accepted 11 Apr. 2010.

Nowadays, the web is a huge repository of information easily accessible by anyone. Consequently, extracting relevant information from the web is a non-trivial task – multiple tries are required to find a desired piece of information.

Web personalization aims at helping users find relevant information and services by tailoring information retrieved from the web based on users’ individual needs. Systems that support users in their selection activities – so called recommender systems – use different types of information for a selection process: history of users’ purchases and selections (content and collaborative filtering systems [4]); or explicitly provided user’s needs (knowledge-based recommender systems [3]). The knowledge-based recommender systems evaluate alternatives based on needs and preferences representing user’s expectations set against the most desirable alternative [8, 28]. Those systems require mechanisms, built based on diverse multi-criteria decision-making approaches, for estimating how well a particular alternative meets each of user’s criteria [3, 24]. Regardless of the information filtering technique, a user profile plays a critical role in a process of identification of users’ points of view for the purposes of information accessing and retrieval.

A user profile represents user’s interests, as well as information and knowledge about domain that is relevant for a user. Representation of user preferences is a necessary factor for building effective and accurate recommender systems. Recommender systems compare user profiles to some reference profiles or item characteristics in order to predict user’s interests in considering items [10, 11]. The outcome of that process depends on ability to accurately identify and represent user’s interests. User profiles can be constructed using explicitly or implicitly collected information. In the example of explicit method, a user is asked to make a list of preferred items or rank/compare provided items. An implicit method, on the other hand, is based on analyzing user web access patterns. Additionally, user interests change quite often, and users are reluctant to specify all adjustments and modifications of their intents and interests. Therefore, techniques that leverage implicit approaches for gathering information about users are highly desired [21, 23].

Application of user profiles as a means for filtering information stored on the web is one of the most

desirable and effective ways of selecting pieces of information and services that fit users' needs and requirements. A very important aspect of that approach is a process of matching users' profiles against information retrieved from the web.

In this paper, we introduce a method for learning and updating a user profile automatically. The proposed method belongs to implicit techniques – it processes and analyzes behavioral patterns of user activities on the web, and modifies a user profile based on extracted information from user's web-logs. The method relies on analysis of web-logs for discovering concepts and items representing user's current and new interests. Those found concepts and items are compared with items from a user profile, and the most relevant ones are added to this profile. The mechanism used for identifying relevant items is build based on a newly introduced concept of ontology-based semantic similarity.

For illustrative purposes, we have implemented the proposed method to update a user profile in the music domain. We compared the estimated similarity values with the similarity values obtained using naïve Bayes classifier. All examples and experiments used to explain the proposed method are related to the music domain.

## 2. Background and Related Work

### A. Ontology and Ontology-based Rules

Ontology as defined by the Semantic Web community deals with a taxonomy of terms that describe a certain area of knowledge. In this context, the most popular definition says: “an ontology is a specification of a conceptualization” [6]. This definition indicates that ontology can be used for building conceptual nets equipped with a structure representing mutual relationships among the concepts [22].

The most important aspect of an ontology used for the Semantic Web applications is related to identifying two ontology layers: the ontology definition layer, and the ontology instance layer. The *ontology definition layer* represents a framework used for establishing a structure of ontology and for defining concepts (classes<sup>2</sup>) existing in a given domain. A structure of ontology is built based on a relation *is-a* between classes. The *ontology instance layer* is composed of concrete information represented as instances (individuals<sup>3</sup>) of ontology classes.

Ontology classes are defined using two types of the properties:

- *datatype property* – is used to represent attributes that can be expressed as values of such data types as

boolean, float, integer, string, and many more (for example, byte, date, decimal, time);

- *object property* – defines other than *is-a* relationships among classes; these relationships follow the notion of Resource Description Framework (RDF) [31] that is based on a triple *subject-predicate-object*, where: *subject* identifies what object the triple is describing; *predicate* (property) defines the piece of data in the object a value is given to; and *object* is the actual value of the property; for example, the triple “John likes books” has “John” as subject, “likes” as predicate and “books” as object.

Both types of properties are very important for defining ontology. The possibility of defining class attributes and any relations between classes creates a very versatile framework suitable for development of complex knowledge bases.

Once an ontology definition is constructed, its individuals can be formed – real data values are assigned to datatype properties, and links to individuals of other classes are assigned to object properties. A special ontology language called OWL [30] has been developed to specify definition as well as instance layers.

It has been shown [7] that the OWL has limitations in the case of representing relations between complex properties. This has been overcome by putting together OWL and a rule language. As the result of that, the Semantic Web Rule Language (SWRL) has been introduced [7]. It combines OWL with RuleML (the sub-language of Rule Markup Language).

In SWRL, a rule axiom consists of an antecedent (body) and a consequent (head). The basic element of both antecedent and consequent is an atom. SWRL identifies five basic atoms that are built based on concepts defined in ontology. The atoms are:

- $C(x)$  - used to check if a given individual  $x$  is an instance of concept  $C$ , for example,  $Track(Yesterday)$  checks if  $Yesterday$  is the instance of the concept  $Track$ ;
- $P(x,y)$  - allows for checking if two individuals  $x$  and  $y$  are related to each other via a property  $P$ , for example,  $genre(Yesterday, rock)$  is “looking” for the property  $genre$  between the individuals  $Yesterday$  and  $rock$ ;
- $Q(x,z)$  - verifies if a data property  $Q$  of an individual  $x$  has a value  $z$ ;
- $sameAs(x,y)$  - holds if individuals  $x$  and  $y$  are the same;
- $differentFrom(x,y)$  - holds if individuals  $x$  and  $y$  are different.

All atoms presented above can be used with variables instead of individuals. The atom  $P(x, y)$  can be used in the following way -  $genre(?t, rock)$ , and it would

<sup>2</sup> The term “ontology class” or “class” will be used throughout the paper.

<sup>3</sup> Recently, the term “instance” has been replaced with “individual”, and the “individual” will be used in the paper.

represent a question: what tracks belong to the genre rock?

### B. Semantic Similarity

There are multiple ways of calculating similarity of concepts/individuals in an ontology. The one that is efficient and matches the human intuition is based on ontology nodes (classes). In the node-based approach similarity is just a distance between the nodes that are being compared [17]. In the edge-based approach similarity is defined as the minimum number of edges between two concept nodes [16]. The main problem with these approaches is the assumption that links in the ontology are uniform. In other words, concepts  $A$  and  $B$  may have the same distance (number of edges or nodes between each other) as concepts  $C$  and  $D$  while the actual similarity of  $A$  and  $B$  may be far from being equal to the similarity of  $C$  and  $D$ . This can be addressed by a notion of weighted edges. Different methods of assigning weights are discussed in [8]:

- network density – greater the density (# of nodes in a part of an ontology) closer the distance between nodes;
- node depth – the distance shrinks when nodes are closer to the bottom of an ontology;
- type of link – type of the link affects calculating the edge weight, e.g., *is-a*, *part-of*, etc.;
- strength of each specific child link – this is defined to differentiate the weights of edges that connect a node with all its child nodes; can be viewed as the closeness of a specific child node to its parent comparing to closeness of its siblings to the parent node.

Rodriguez et. al. [18] introduced a similarity function defined by weighted sum of three different similarities. A similarity of a class  $a$  of ontology  $p$  to a class  $b$  of ontology  $q$  is expressed as:

$$S(a^p, b^q) = w_w * S_w(a^p, b^q) + w_u * S_u(a^p, b^q) + w_n * S_n(a^p, b^q) \quad (1)$$

where  $S_w$  represents a word matching among synonym sets [9] denoted by classes  $a$  and  $b$ ,  $S_u$  is a feature matching over corresponding types of features of classes  $a$  and  $b$ , and  $S_n$  is a semantic-neighborhood matching comparing classes in semantic neighborhoods based on synonym sets or feature matching;  $w_s$  are respective weights of similarity components, for example,  $w_w$  is weight of the similarity between synonym sets. These weights depend on the characteristics of a given ontology. The semantic neighborhood for a given concept is the set of concepts with a distance lower than a non-negative integer. The similarity measures are defined in terms of a matching process:

$$S(a, b) = \frac{|A \cap B|}{|A \cap B| + \alpha(a, b)|A/B| + (1 - \alpha(a, b))|B/A|} \quad (2)$$

where  $A$  and  $B$  are description sets of classes  $a$  and  $b$ , i.e., synonym sets, sets of distinguishing features and a set of classes in semantic neighborhood;  $(A \cap B)$  and  $(A/B)$  represent intersection and difference respectively,  $| \cdot |$  is the cardinality of a set; and  $\alpha$  is a function that defines relative importance of non-common characteristics.

### C. User Profiling

A user profile is used to identify user's interests and preferences. There are several approaches suitable for constructing a user profile [1, 9, 25]. The most intuitive one is to develop a profile by asking a user multiple questions. However, the user might not be willing to give information regularly, and on top of that user's interests change constantly.

Processes of constructing user profiles can be divided into two categories: 1) knowledge-based, and 2) behavior-based [10]. The former considers a user model as static and uses questionnaires and interviews to match a user model to one of already existing models, while the latter constructs a model of a user based on the patterns discovered from user's behavior on the web applying machine-learning techniques.

Most recommender systems use a behavior-based approach and they model a user in a binary fashion. A binary profile is developed based on user evaluations of pages as interesting or uninteresting. Machine-learning techniques are applied to identify interesting pages [19]. The popular approaches – examples of knowledge-based methods – used by industry to construct user models include server-side accounts, and identity profiles. However, these methods are incapable of using and integrating information about a user.

Web usage mining – the process of discovering patterns from web data using data mining methods – strives to find user preferences based on the user web-logs that reside on servers. Web-logs<sup>4</sup> represent a website's usage – visitor's IP address, time and date of access, and accessed files.

A novel idea has been proposed in [23] for ontological user profile architecture considering short-term and long-term memory. Based on user's behavior, the method assigns interest scores to existing items of domain ontology. This approach is used for re-ranking the results of a search engine in order to provide personalized results.

In the case of personalized web agents, agents are able to learn user preferences and discover web information sources based on those preferences. Examples are WebWatcher [2] and Syskill & Webert [12]. WebWatcher uses both TFIDF (in learning from previous tours) and reinforcement learning (in learning from hypertext

<sup>4</sup> <http://www.w3.org/TR/WD-logfile.html>

structure) to suggest an appropriate link given an interest and webpage. Syskill & Webert utilizes a user profile and learns about “interestness” of Web pages using a Bayesian classifier.

D. Ordered Weighted Averaging

Aggregation of different pieces of information is a common aspect of any system that has to infer a single outcome from multiple facts. A very interesting class of aggregation operators is called the Ordered Weighted Averaging (OWA) [26]. In the simplest possible statement, this operator is a weighted sum over ordered pieces of information.

In a formal representation, the OWA operator, defined on the unit interval  $I$  and having dimension  $n$  ( $n$  arguments), is a mapping  $F_w: I^n \rightarrow I$  such that:

$$F_w(a_1, \dots, a_n) = \sum_{j=1}^n (w_j * b_j) \tag{3}$$

where  $b_j$  is the  $j^{th}$  largest of all arguments  $a_1, a_2, \dots, a_n$ , and  $w_j$  is a weight such that  $w_j$  is in  $[0, 1]$  and  $\sum_{j=1}^n w_j = 1$ .

If  $id(j)$  is the index of the  $j^{th}$  largest of  $a_i$  then  $a_{id(j)} = b_j$  and  $F_w(a_1, \dots, a_n) = \sum_{j=1}^n (w_j * a_{id(j)})$ . If  $\mathbf{W}$  is an

$n$ -dimensional vector whose  $j^{th}$  component is  $w_j$  and  $\mathbf{B}$  is an  $n$ -dimensional vector whose  $j^{th}$  component is  $b_j$  then  $F_w(a_1, a_2, \dots, a_n) = \mathbf{W}^T \mathbf{B}$ . In this formulation  $\mathbf{W}$  is referred to as the OWA weighing vector and  $\mathbf{B}$  is called the ordered argument vector.

The OWA operator is parameterized by the weighing vector  $\mathbf{W}$ . A number of interesting observations can be done when different  $\mathbf{W}$  are considered. For example, if  $\mathbf{W} = \mathbf{W}_*$  where  $w_n = 1$  and  $w_j = 0$  for  $j \neq n$  then  $F_w(a_1, a_2, \dots, a_n) = \text{Min}_j[a_j]$ . If  $\mathbf{W} = \mathbf{W}^*$  where  $w_1 = 1$  and  $w_j = 0$  for  $j \neq 1$  then  $F_w(a_1, a_2, \dots, a_n) = \text{Max}_j[a_j]$ . If  $\mathbf{W} = \mathbf{W}_N$

where  $w_n = \frac{1}{n}$  then  $F_w(a_1, \dots, a_n) = \frac{1}{n} \sum_{j=1}^n a_j$ , what

represents an arithmetic mean (average). Various other forms can be described. In general, it can be said that different values of weights  $w_j$  control the level of contribution of single pieces of information towards the final outcome.

At the beginning of eighties, Zadeh has introduced the concept of linguistic quantifiers. Those quantifiers describe a proportion of objects. According to Zadeh, a person knows a vast array of terms that are used to express information about proportions. Some examples are *most*, *at least half*, *all*, and *about 1/3*. The important issue is to formally represent those quantifiers.

In the mid-nineties, Yager showed how we can use a linguistic quantifier to obtain a weighing vector

associated with an OWA aggregation. He has introduced parameterized families of the Regular Increasing Monotone (RIM) quantifiers. These quantifiers are able to guide aggregation procedures by verbally expressed concepts in a description independent dimension. A RIM quantifier is a fuzzy subset  $Q$  over  $I = [0, 1]$  in which for any proportion  $r \in I$ ,  $Q(r)$  indicates the degree to which  $r$  satisfies the concept indicated by the quantifier  $Q$ . A fuzzy subset  $Q$  represents a RIM quantifier if:

- 1)  $Q(0) = 0$
- 2)  $Q(1) = 1$
- 3) if  $r1 > r2$  then  $Q(r1) > Q(r2)$  (monotonic)

For example, let us take a look at the parameterized family  $Q(r) = r^p$ , where  $p \in [0, \infty)$ . Here if  $p=0$  we obtain the *existential (max)* quantifier; when  $p \rightarrow \infty$  we have the quantifier *for all (min)*, and when  $p=1$  we have  $Q(r) = r$  and we deal with the quantifier *some*. In addition for the case  $p=2$ ,  $Q(r) = r^2$ , we obtain one possible interpretation of the quantifier *most*. These quantifiers are shown in Figure 1.

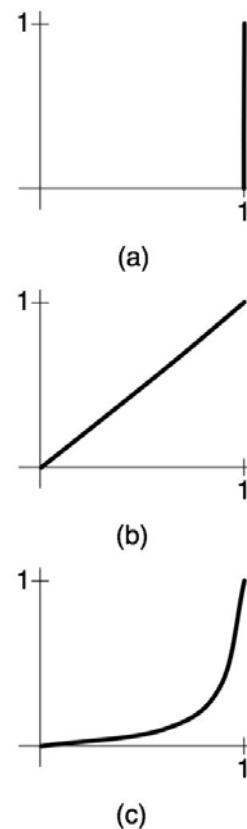


Figure 1. Linguistic quantifiers: for *all* (a), *some* (b), and *most* (c).

Assuming a RIM quantifier, we can associate with  $Q$  an OWA weighing vector  $\mathbf{W}$  such that for  $j=1$  to  $n$ :

$$w_j = Q\left(\frac{j}{n}\right) - Q\left(\frac{j-1}{n}\right) \tag{4}$$

where  $n$  is a number of pieces of information to be aggregated. This expression indicates that the weighing

vector  $\mathbf{W}$  is a manifestation of the quantifier underlying the aggregation process, and  $Q(k) = \sum_{j=1}^k w_j$ , where  $k \leq n$ .

(The relations between a quantifier  $Q$  and a weighing vector can be illustrated by the following observation:  $Q(0) = 0$  if a decision-maker is absolutely not satisfied (no criteria satisfied);  $Q(n) = 1$  if he is completely satisfied, i.e., if all criteria are satisfied.) Using this expression the values of the weighing vector can be obtained directly from the expression representing the quantifier. For example, the quantifier *some* can be expressed by the formula  $Q(r) = r$ . For this quantifier we have:

$$w_j = \frac{j}{n} - \frac{j-1}{n} = \frac{1}{n}$$

and this gives us a simple average. Such process is illustrated in Figure 2.

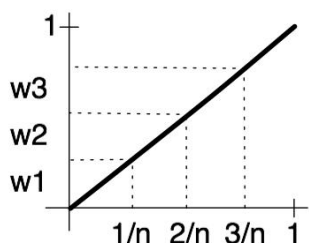


Figure 2. Obtaining weights from a quantifier.

The concept that is associated with OWA operator is the measure of its “orness”. If we have  $\mathbf{W}_*$  (see above for explanation of  $\mathbf{W}_*$ ) then an OWA operator  $F$  is a pure “and” operator, while if we have  $\mathbf{W}^*$ , then  $F$  is a pure “or” operator. We can further observe the closer the total weight is to being in  $w_1$ , the closer the  $F$  function is to being a pure “or” operator, while the closer it is to being in  $w_n$ , the function is closer to an “and.” The formal definition of “orness” is as follows.

Assume  $F$  is an OWA aggregation operator with weighing vector  $\mathbf{W} = [w_1, w_2, \dots, w_n]$ . The degree of “orness” associated with this operator is defined as

$$orness(W) = \frac{1}{n-1} * \sum_{i=1}^n ((n-i) * w_i) \quad (5)$$

The “orness” of the parameterized family  $Q(r) = r^p$ , where  $p \in [0, \infty)$  is approximated by  $1/(p+1)$ . So, for the quantifiers *for all*, *some*, and *most* “orness” is equal to 0, 1/2, and 1/3 respectively.

### 3. Finding Relevant Items

#### A. Concept

In this paper, we propose a method for identification of new items that can be of interest for users, and to

update their profile with items that are the most relevant to them. The proposed method updates the user profile, originally provided by the user, without asking the user to explicitly provide any information related to their changing interests. This is achieved by processing data (web-logs) representing user’s web access behavior. The method uses an ontology-based semantic similarity to compare items browsed by a user on the web with the items from a user’s profile. Additionally, importance of the browsed items is evaluated. The importance is combined with similarity in order to obtain a level of relevance.

The overview of the proposed method is illustrated in Fig. 3. A process of finding relevant items that can be added to a user profile is performed in multiple steps.

The semantic similarity is estimated based on a domain ontology that contains different relationships existing between items. The similarity is estimated for each pair of items where one item is taken from a user profile, while the other one from a set of items found on a web page. This similarity expresses a common view of similarity of items as defined in an ontology – i.e., it expresses a view of anyone involved in construction and maintenance of a given domain ontology. A set of web page items that are similar to items from the user profile is considered as a set of items that can be added to this profile. However, in order to reflect a user’s point of view, i.e., to select items a user is interested in – an item importance measure is introduced. This measure is calculated based on a number of times a web page, with a particular item, has been seen by a user. The last step in identifying relevant items is a simple combination of both measures – semantic similarity and item importance.

The initial step is an extraction of URIs from web-log files created during a single web session. Additionally, besides URIs, the process identifies how many times each page has been visited. The extracted addresses of web pages are used to download those pages. Each page is processed in order to identify domain-related words, called hereafter items, considered for addition to the user’s profile. A bag of words representing a single page is obtained via a simple string matching process. The process uses the items from the domain ontology to find potentially related words on the page. It means, all the items that are found on the page do exist in the ontology, and assortment of those items depends on diversity of words that ontology contains. In the case of the music domain, we use ontology populated with words from MusicBrainz [29], and the items are identified using the UIMA (Unstructured Information Management Architecture) library developed by IBM [5]. Once domain-related items are recognized on the page and linked to the domain ontology, we evaluate their

relevance to user's interests.

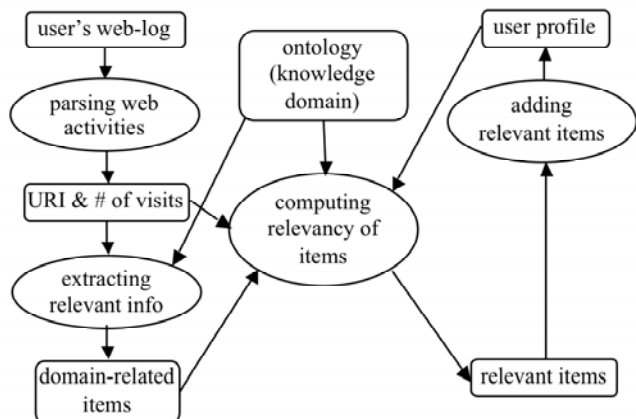


Figure 3. The process of updating user profile.

A process of evaluating and selecting the most relevant items follows a special procedure, Fig. 4. This procedure has three distinguished steps: computing semantic similarity of items, computing importance of each item, and combining computed similarity with importance.

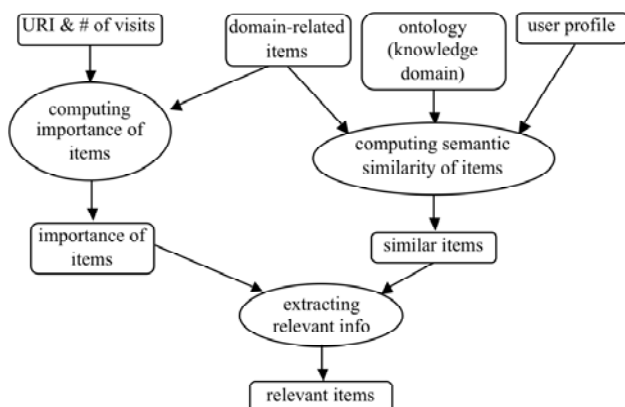


Figure 4. The process of computing relevancy of items.

### B. Semantic Similarity of Items

The difficulty of finding pairs of similar items lays in the fact that items that we deal with do not exist in a numerical space. Therefore, there is no possibility of identifying a distance between two items. To evaluate similarity we propose a novel technique that estimates similarity between items in a non-numerical space. The technique uses a domain ontology and rules built based on this ontology. Simply stated, similarity is represented via a set of rules whose body and head consist of atoms built from ontology concepts (Section 2.A).

In the proposed method for evaluating similarity of non-numerical items, an ontology represents a knowledge network containing relations between ontology classes. Those relationships could be of two

types: *is-a* – representing superclass-subclass relationships, and *object properties* – representing relationships existing between classes as recognized and defined by an ontology developer – those relationships represent semantic relations existing between classes. In Section II.A, we showed that rules could be built using relationships defined in an ontology. The rules that can represent different levels of similarity between involved classes are of special interests for us. For example, let us assume we have two classes *Artist* and *Work*, with a relationship between them *made*. If we build an antecedent:

**IF** *Artist\_A* made *Work\_I* and *Work\_II* **THEN** ...

then we can use it to construct a rule expressing a level of similarity between two different works. In other words, we can say that if both *Work\_I* and *Work\_II* are in the relationship *made* with the same *Artist\_A* then some level of similarity exists between them. To determine a level of similarity, we average similarity estimations provided by multiple individuals (Section 4.B). The individuals who performed those evaluations could be experts in the domain or/and users who will use the approach to update their profiles. The involvement of experts will ensure subjectivity of similarity evaluations, while the involvement of users will ensure that the similarity estimations reflect their own perception of similarity.

The process of evaluating similarity between two classes starts with construction of a number of rules that take into account different types of relations existing between these two classes. A person inspects each rule, and determines a similarity level existing between two classes based on their own subjective opinion. This level is assigned to the rule. For example, the rule presented above could be in the form:

**IF** *Artist\_A* made *Work\_I* and *Work\_II*  
**THEN** *similarity\_is\_at\_level\_K*

This indicates that for the person who inspected this rule if both *Work\_I* and *Work\_II* are in the relationship *made* with the same *Artist\_A* then similarity between these works is at the level *K*. In general, the similarity between two classes can be expressed by multiple rules. The rules are created at the very beginning and they are kept for the whole process of evaluation. It is possible for the user/expert to modify the rules as well as similarity levels assigned to them at any time during utilization of the proposed method.

A set of rules (their antecedents) that can be used to express similarity in the domain of human creativity – art (paintings, sculpture, theatrical plays), and music (classical, and modern) is shown below. The rules have been grouped into three categories: the rules that represent different relationships between two artists

(Artist-Artist Group), the rules that illustrate how two works can be correlated (Work-Work Group), and the rules that express associations between an artist and a work (Work-Artist Group). Some examples of rule from each category are presented below:

#### Artist-Artist

- IF *artists* belong to the same *group\_of\_artists*<sup>5</sup>  
THEN similarity between *artists* is at level ...
- IF *artists* collaborated with each other  
THEN similarity between *artists* is at level ...
- IF *artists* collaborated with another (the same) *artist*  
THEN similarity between *artists* is at level ...
- IF *artists* made *works* that belong to the same *style*<sup>6</sup>  
THEN similarity between *artists* is at level ...
- IF *artists* belong to the same *style*  
THEN similarity between *artists* is at level ...

#### Work-Work

- IF *works* belong to the same *collection\_of\_works*<sup>7</sup>  
THEN similarity between *works* is at level ...
- IF *works* are made by the same *artist*  
THEN similarity between *works* is at level ...
- IF *works* belong to the same *style*  
THEN similarity between *works* is at level ...
- IF *works* are made by *artists* who belong to the same *style*  
THEN similarity between *works* is at level ...

#### Work-Artist

- IF *work* and *artist* belong to the same *style*  
THEN similarity between *them* is at level ...
- IF *artist* and *artist* who made work belong to the same *style*  
THEN similarity between *them* is at level ...
- IF *artist* collaborated with *artist* who made work  
THEN similarity between *them* is at level ...
- IF *work* is made by *artist*  
THEN similarity between *them* is at level ...

The above rules are built using ontology classes and relations between them: classes are shown in italic, while relations are underlined. The fact that those rules are defined based on classes means that they are generic and can be applied to any individuals (instances) of those classes. The domain-relevant items found on web pages and items that constitute a user profile are “treated” as individuals (instances), thus the proposed technique can be used to evaluate similarity between them. All those rules are expressed using a SWRL format (Section 2.A).

In general, there may be multiple rules that are fired for the same two items being compared. This means that different “types” of relationships exist between those two items, as well as different levels of similarity. The approach proposed here combines those different similarities and obtains a final result using OWA (Section 2.D). The details are presented in Section 4.C.

Overall, the process of determining a similarity between items from the visited web pages to items from

the user profile is done at two levels – an individual level, and a general level. The individual level focuses on “comparison” of a single item from a visited page with each item from the profile. The comparison is performed using all rules. When several rules fire a number of similarity values is obtained. Those values are aggregated using OWA. This process finishes with several similarity values, where each value represents the similarity between the single web page item and one item from the profile. The general level similarity estimation takes all those values and combines them into one value that is the similarity of the single web page item to the whole profile. Section V contains a detailed example of this process.

#### C. Importance of Items

The semantic similarity measure proposed here represents a generic similarity measure for given items. The domain ontology with its relations, as well as rules built based on those relations are developed by experts. Therefore, obtained similarities are, up to some degree, reflections of expert knowledge embedded in the ontology. This does not mean that a user is interested in all items that are similar to the items from his profile. The specific user interests may not match the similarity measures obtained based on rules, i.e., obtained from logical hierarchy and relationships expressed in the ontology. To solve this we introduce an importance measure  $I(c_i)$  of a item  $c_i$  based on user’s web activities:

$$I(c_i) = \frac{\sum_{j=1}^{NP} N_{d_j}(c_i)}{\sum_{j=1}^{NP} N_{d_j}} \quad (6)$$

where  $N_{d_j}(c_i)$  is a number of occurrences of the item  $c_i$  on the web page  $d_j$ <sup>8</sup>,  $N_{d_j}$  is a total number of items on the page  $d_j$ , and  $NP$  represents a total number of pages visited by the user. This simple measure is an estimation of user’s interest in a given item – it is proportional to a total number of times the item appears on all pages divided by the total number of times all relevant items appear on all pages. A number of possible measures could be used here, however we have proposed and used a measure that “represents” item’s overall importance, and not its importance to a single document.

#### D. Calculating Relevance of Items

The level of relevancy of the web page items to the user profile items is calculated by combining the semantic similarity of items with their importance. This process is preformed using a fuzzy approach. In this case,

<sup>5</sup> They are artists that had the same teacher/trainer, spent time together, worked on similar things, or belonged to the same music band.

<sup>6</sup> It can be anything of that sort, i.e., genre = style = art movement = music movement.

<sup>7</sup> They are works created at the same time, the same place, located at the same place, or coming from the same music record.

<sup>8</sup> Items are words/phrases defined in a domain ontology. Web pages are annotated based on that ontology (Section III.A).

both semantic similarity and importance are fuzzified. A number of linguistic terms have been defined on the universes of discourse for both measures. For the proposed similarity measure the universe of discourse is in the range from 0 to 1, and three different linguistic labels have been defined: *small*, *medium* and *high*. The fuzzy sets associated with those labels are uniformly distributed across the universe of discourse. For the importance measure, the range is from 0 to 1, and three terms: *small*, *medium* and *high* have been identified. However, their distribution is not uniform – values above 0.5 represent the importance *high* (see Section 5.B for details).

Once the values of similarity and importance are fuzzified a single rule is used to induce the level of relevance of the web page item to the user profile item:

**IF** *similarity* is high and *importance* is high  
**OR** *similarity* is medium and *importance* is high  
**THEN** *level\_of\_relevance* is high<sup>9</sup>

The level of satisfaction of this rule represents the item's level of relevance. If a threshold value for the relevance is established – then the item with the relevance value above the threshold is added to the user profile. The inference is based on the min-max (Mamdani's) mechanism. It uses the min t-norm as the implication function, and the max s-norm as the aggregation operator. The result of the inference mechanism is de-fuzzified based on commonly used method called the center of gravity [13].

## 4. Musical Domain Application

### A. Music Ontology

Music Ontology (MO) is an initiative aiming at development of a formal specification of concepts and relationships describing objects in the music domain [15]. It is built on top of three ontologies: Timeline [14] for expressing temporal information, Event [20] for expressing events, and Functional Requirements for Bibliographic Records (FRBR) for concepts: Work (artistic creation), Manifestation (physical embodiment), Item (prototype of such manifestation), and Expression (realization of a work).

We populated our MO with over 500 objects in five classes: *SoloMusicArtist*, *MusicGroup*, *Track*, *Record*, and *Genre*. We employed MusicBrainz [29] to populate MO with individuals. MusicBrainz does not provide explicit genre of tracks, records and artists, therefore we retrieved this information from Wikipedia (300 genres in 13 classes).

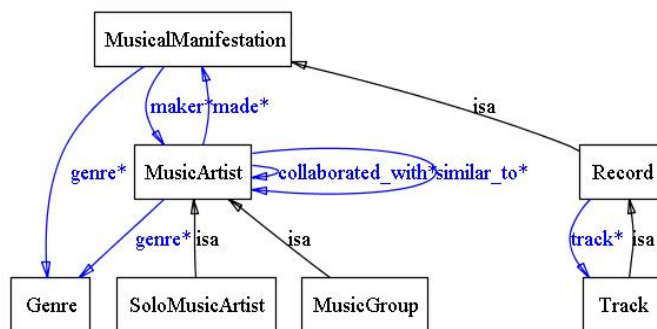


Figure 5. The object properties of Music Ontology.

Fig. 5 illustrates MO properties. The class *MusicArtist* is linked to the class *MusicalManifestation* through the object property *made* (the star indicates multiple links). This means that multiple *MusicalManifestations* can be made by a *MusicArtist* and a *MusicGroup* or a *SoloMusicArtist* which are of a type *MusicArtist*. A *MusiArtist* can be affiliated with multiple *Genres*, as well as a *Record* or a *Track* which are both a type of *MusicalManifestation*.

### B. Rule for Semantic Similarity in Music Domain

The SWRL rules are built based on rules presented in Section III.C. We have considered such classes as: *Track*, *Record*, *Genre*, *SoloMusicArtist* and *MusicGroup*, and such object properties (relations) as: *maker*, *made*, *genre*, *track*, *collaborated\_with*, and *similar\_to*. A total of 20 rules are used to assess the semantic similarity between items from web pages accessed by a user and items from a user's profile. Each rule assigns a specific similarity level. We asked different individuals to estimate a similarity level that should be induced by each rule. All rules have been sorted based on ratings obtained from the individuals, and values from 0 to 1 have been assigned to them. All rules with their similarity levels are presented below. Please note that all of them have been labeled. These labels will be used in the experimental section (Section 5) when details illustrating estimation of semantic similarity are presented.

#### Artist-Artist

- AA1:** IF *artists* belong to the same *music band*  
 THEN similarity between *artists* is at **level 0.9**
- AA2:** IF *artists* collaborated with each other  
 THEN similarity between *artists* is at **level 0.6**
- AA3:** IF *artists* collaborated with another (the same) *artist*  
 THEN similarity between *artists* is at **level 0.5**
- AA4:** IF *artists* made tracks that belong to the same *genre*  
 THEN similarity between *artists* is at **level 0.4**
- AA5:** IF *artists* belong to the same *genre*  
 THEN similarity between *artists* is at **level 0.3**

#### Track-Track

- TT1:** IF *tracks* belong to the same *record*  
 THEN similarity between *works* is at **level 0.9**
- TT2:** IF *tracks* are made by the same *artist*  
 THEN similarity between *works* is at **level 0.7**

<sup>9</sup> Level of relevance is also a fuzzy variable with the universe of discourse from 0 to 1, and uniformly distributed terms: *low*, *medium*, and *high*.

**TT3:** IF *tracks* belong to the same *genre*  
THEN similarity between *tracks* is at **level 0.5**

**TT4:** IF *tracks* are made by artists who belong to the same *genre*  
THEN similarity between *tracks* is at **level 0.3**

Track-Artist

**TA1:** IF *track* and *artist* belong to the same *genre*  
THEN similarity between *them* is at **level 0.6**

**TA2:** IF *artist* and *artist* who made track belong to the same *genre*  
THEN similarity between *them* is at **level 0.5**

**TA3:** IF *artist* collaborated with *artist* who made track  
THEN similarity between *them* is at **level 0.4**

**TA4:** IF *track* is made by artist  
THEN similarity between *them* is at **level 0.9**

Additionally to the basic rules presented in Section 3.B, we have added a number of rules useful for estimating similarity between items that belong to MO. We added three rules for expressing relationships between two records (Record-Record group), as well as two rules for expressing connections between record and artist (Record-Artist), and two for record and track (Record-Track) relationships.

Record-Record

**RR1:** IF *records* are made by the same *artist*  
THEN similarity between *works* is at **level 0.8**

**RR2:** IF *records* belong to the same *genre*  
THEN similarity between *tracks* is at **level 0.4**

**RR3:** IF *records* are made by artists who belong to the same *genre*  
THEN similarity between *tracks* is at **level 0.3**

Record-Artist

**RA1:** IF *track* and *artist* belong to the same *genre*  
THEN similarity between *them* is at **level 0.5**

**RA2:** IF *record* is made by artist  
THEN similarity between *them* is at **level 0.8**

Record-Track

**RT1:** IF *record* and *track* are made by the same *artist*  
THEN similarity between *works* is at **level 0.8**

**RT2:** IF *record* and *track* belong to the same *genre*  
THEN similarity between *tracks* is at **level 0.4**

All those rules are “translated” into a SWRL format. In order to compare items that appear in the user profile and her web-log, we introduce two classes (from the top class – owl:Thing) to the ontology. One of them represents all items from the profile – owl:Profile, while the second class represents items from the user weblog – owl:Weblog. Therefore, the SWRL rules contain atoms built from classes and properties from MO, and both classes *Profile* and *Weblog*. For example, the SWRL format of the rule TT3 is:

$$\begin{aligned} &Track(?w) \wedge Track(?p) \wedge Profile(?p) \wedge Weblog(?w) \\ &\wedge genre(?w, ?z) \wedge genre(?p, ?z) \\ &\rightarrow level\_five(?w) \end{aligned}$$

The rule asserts that if a track in the user profile and a track in the user-accessed web page are both from the same genre then a similarity level between them is five.

### C. Application of OWA for Similarity Estimation based

#### on Multiple Rules

Two items – individuals of MO classes – can be related to each other in a number of ways. This fact is represented by the existence of multiple rules involving those items. For example, two *artists* can be related if they belong to the same band (rule AA1), if they collaborated with each other (rule AA2), if they collaborated with the same other *artist* (rule AA3), if they created *tracks* that belong to the same *genre* (rule AA4), or if they themselves belong to the same *genre* (rule AA5). Each of those relations (rules) represents a different aspect and level of similarity. Therefore, it is important to “merge all available evidences”, and combine the similarity levels obtained from all rules that fired for two given artists. It is very likely that both of them are related to each other in more than one of presented ways: AA1-AA5.

A process of combining different levels of similarity in order to obtain a single similarity estimation value can be done in multiple ways: simple averaging, weighted averaging, or even voting. However, what we would like to achieve is to mimic, up to some degree, a human way of combining multiple pieces of information. It can be stated that each rule represents a different piece of evidence that contributes to the overall similarity level. In real life, different people will “combine” pieces of evidence in a different way. For example, for one person the existence of a single evidence (firing of one rule) is enough to decide that two items are similar, another person needs a few evidences – firing of a few rules, yet another person needs most of evidences to be present (most of rules should fire) to be convinced that two items are similar. In order to have the ability to combine multiple pieces of evidence in numerous different ways we apply the ordered weighted averaging (OWA) operator (Section 2.D).

The OWA allows us to combine evidences in a way that is expressed with a linguistic quantifier. This makes the proposed approach more human like – the combination of evidences can be described with statements like: “similarity between two items is determined by ONE rule with the highest level of similarity among all fired rules”, or “if SOME rules fire then two items are similar”, or “if MOST rules fire then two items are similar”. The following subsections provide some details of the approach.

#### Existential Operator

The first statement above: “similarity between two items is determined by ONE rule with the highest level of similarity among all fired rules” is an example of application of an existential quantifier. In a formal way, this quantifier is represented by a word OR, and is equivalent to the operation “max”. It means that it takes

into consideration all available pieces of information and “selects” one with the highest value.

For the case of multiple similarity rules, the OR approach selects as a single value of similarity level equal to the highest similarity provided by all rules fired when two items are “compared”. For example, for the AA group of rules, if rules AA2 and AA4 are fired – the similarity is  $\max(0.4, 0.6)$ , i.e., it is 0.6.

In order to obtain such “behavior”, the OWA operator with the linguistic quantifier OR is used. The weights of OWA are calculated according to the formula (2), Section 2.D:

$$w_j = Q\left(\frac{j}{n}\right) - Q\left(\frac{j-1}{n}\right)$$

where  $n$  is a number of pieces of information to be aggregated. For OR, the weight values are:

$$w_1 = 1; \quad w_2 = 0; \quad w_3 = 0; \quad w_4 = 0; \quad w_5 = 0$$

for the group AA,

$$w_1 = 1; \quad w_2 = 0; \quad w_3 = 0; \quad w_4 = 0$$

for the groups TT, TA,

$$w_1 = 1; \quad w_2 = 0; \quad w_3 = 0$$

for the group RR, and

$$w_1 = 1; \quad w_2 = 0$$

for the groups RA and RT.

### Operator SOME

The statement “if SOME rules fire then two items are similar” involves a linguistic quantifier SOME. The OWA weights for this quantifier are calculated with  $Q(r) = r$ . So, the weight values are:

$$w_1 = \frac{1}{5}; \quad w_2 = \frac{1}{5}; \quad w_3 = \frac{1}{5}; \quad w_4 = \frac{1}{5}; \quad w_5 = \frac{1}{5}$$

for the group AA, and

$$w_1 = \frac{1}{4}; \quad w_2 = \frac{1}{4}; \quad w_3 = \frac{1}{4}; \quad w_4 = \frac{1}{4}$$

for the groups TT, TA,

$$w_1 = \frac{1}{3}; \quad w_2 = \frac{1}{3}; \quad w_3 = \frac{1}{3}$$

for the group RR, and

$$w_1 = \frac{1}{2}; \quad w_2 = \frac{1}{2}$$

for the groups RT and RA.

### Operator MOST

The third possibility considered here is the application of the quantifier MOST. It implies that most pieces of evidence should support a final conclusion – it means that more rules fire more “consistent” similarity is obtained. The OWA weights for this quantifier are calculated with  $Q(r) = r^2$ . So, the weight values are:

$$w_1 = \frac{1}{25}; \quad w_2 = \frac{3}{25}; \quad w_3 = \frac{5}{25}; \quad w_4 = \frac{7}{25}; \quad w_5 = \frac{9}{25}$$

for the group AA,

$$w_1 = \frac{1}{16}; \quad w_2 = \frac{3}{16}; \quad w_3 = \frac{5}{16}; \quad w_4 = \frac{7}{16}$$

for the groups TT, TA,

$$w_1 = \frac{1}{9}; \quad w_2 = \frac{3}{9}; \quad w_3 = \frac{5}{9}$$

for the group RR, and

$$w_1 = \frac{1}{4}; \quad w_2 = \frac{3}{4}$$

for the groups RT and RA.

### Example

In order to illustrate the application of the OWA operator and show differences when three different linguistic quantifiers are used, a simple example is presented. Let us assume that we have four rules from the TT group. They are listed below in the ordered – based on the similarity level – fashion:

- TT1: IF *tracks belong to* the same *record*  
THEN similarity between *tracks* is **at level 0.9**
- TT2: IF *tracks are made by* the same *artist*  
THEN similarity between *tracks* is **at level 0.7**
- TT3: IF *tracks belong to* the same *genre*  
THEN similarity between *tracks* is **at level 0.5**
- TT4: IF *tracks are made by artists who belong to* the same *genre*  
THEN similarity between *tracks* is **at level 0.3**

The general formula for similarity between two items is:

$$\text{Similarity}(item_1, item_2) = w_1 * b_1 + w_2 * b_2 + w_3 * b_3 + w_4 * b_4$$

When only two rules **TT1** and **TT3** fired we have **b1 = 0.9** (from TT1), **b2 = 0.5** (from TT3), and **b3=b4=0** (from the fact that TT2 and TT4 did not fire). The similarity calculated with OR quantifier is

$$\text{Similarity}(item_1, item_2) = 1 * 0.9 + 0 * 0.5 + 0 * 0 + 0 * 0 = 0.9.$$

This value is normalized using the value of similarity when all rules are fired. In this example, the normalization factor is 0.90, and the final value of *similarity* is 1.00. For the quantifier SOME we have

$$\begin{aligned} \text{Similarity}(item_1, item_2) &= \frac{1}{4} * 0.9 + \frac{1}{4} * 0.5 + \frac{1}{4} * 0 + \frac{1}{4} * 0 \\ &= \frac{1.4}{4} = 0.35. \end{aligned}$$

With the normalization factor (the value when all rules are fired) equal to 0.60, the *similarity* is 0.58. The quantifier MOST provides even more strict conditions to be satisfied. In the example when only two rules fired, the OWA formula is

$$\begin{aligned} \text{Similarity}(item_1, item_2) &= \frac{1}{16} * 0.9 + \frac{3}{16} * 0.5 + \frac{5}{16} * 0 + \frac{7}{16} * 0 \\ &= \frac{2.4}{16} = 0.15 \end{aligned}$$

and after the normalization process (the normalization factor – the value when all rules are fired – is equal to 0.48), the *similarity* is 0.31.

In the case, when TT2 and TT3 were fired –  $\mathbf{b1} = 0.7$  (from TT2),  $\mathbf{b2} = 0.5$  (from TT3), and  $\mathbf{b1}=\mathbf{b4}=0$  (from the fact that TT1 and TT4 did not fire), we would have:  $\text{similarityOR} = 0.78$ ,  $\text{similaritySOME} = 0.50$ , and  $\text{similarityMOST} = 0.19$ .

#### D. Application of Naïve Bayesian Classifier for Similarity Estimation

The proposed method for estimating similarity between a single item and items from the user’s profile can be compared to a method based on a well-known Bayes’ theorem. A naïve Bayes classifier is a simple probabilistic classifier that assumes independence of features used for discrimination purposes.

Let us assume we have a class – *Similar Items* ( $C_{SI}$ ) – of items similar to the profile provided by the user. Using Bayes’ theorem we can write the probability that a given  $item_k$  belongs to the class  $C_{SI}$  as:

$$P(C_{SI} / item_k) = \frac{P(C_{SI})}{P(item_k)} * P(item_k / C_{SI}) \quad (7)$$

where  $P(C_{SI})$  is the probability of the class  $C_{SI}$ ,  $P(item_k)$  is the probability of the  $item_k$ , and  $P(item_k / C_{SI})$  is the probability of  $item_k$  given the class  $C_{SI}$ . We assume that  $P(item_k)$  is the same for all items  $item_k$ , therefore:

$$P(C_{SI} / item_k) \propto P(item_k / C_{SI})$$

The  $item_k$  is a single word and we cannot “divide” it into smaller pieces, as it happens in the case of a document classification process where  $doc_k$  from  $P(doc_k / class)$  is represented by a set of words, and:

$$P(doc_k / class) = \prod_i P(word_{k,i} / class)$$

therefore expressing  $P(item_k / C_{SI})$  as a product of probabilities is not doable. However, the combination of the concept of the extension of Bayesian classifier [26] and the concept of rules representing different levels of similarity provides the following solution:

$$P(item_k / C_{SI}) = \prod_i P(rule_{k,i} / C_{SI}) \quad (8)$$

where  $rule_{k,i}$  represents an  $i$ -th rule that fires for  $item_k$ , and the probability  $P(rule_{k,i} / C_{SI})$  of firing  $rule_{k,i}$  when  $item_k$  belongs to  $C_{SI}$  is equivalent to the similarity level associated with the  $rule_{k,i}$ . In other words, this probability is the level of similarity “defined” by the rule  $rule_{k,i}$ . So,

$$P(C_{SI} / item_k) \propto \prod_i P(rule_{k,i} / C_{SI}).$$

For example, let us consider rules TT1 to TT4 from the example in Section IV.C. If rules TT1 and TT4 fire for a given  $item_k$ , then

$$P(C_{SI} / item_k) \propto P(TT_1 / C_{SI}) * P(TT_4 / C_{SI}) = 0.9 * 0.3.$$

## 5. Results and Discussion

### A. Experiment Overview

In our experiments we used an example of a real-life scenario in which the user provides her initial profile, and then browses music-related web pages. We retrieve the URIs and number of pages visited during three different sessions. The music-related items are extracted from the user-accessed pages. All these web page items are compared with items from the user profile. Each item is labeled with a relevance value calculated based on semantic similarity and importance values. The items with relevancy values above a defined threshold (0.5) are added to the user profile.

### B. User Profile and Web Page Items

The user profile includes a number of music-related items: two music artists, two records, and four tracks, Table 1. The items from the web pages visited during the first session are shown in Table 2.

Once a set of items from eight user-accessed pages is identified, each item is compared with the item from the user portfolio. Multiple rules are fired for each item-to-item comparison. All rules fired for the items from the first session are shown in Table 3. It can be easily identified that in most cases more than one rule fired.

Table 1. Music items from the user profile (A-artist, T-track, R-record, the name in italic represents an artist associated with a given track or record).

Madonna (A)   Mariah_Carey (A)   Die_Another_Day (T, <i>Madonna</i> )   For_the_Record (T, <i>Mariah_Carey</i> )   Hard_Candy (R, <i>Madonna</i> )   I_Got_U (T, <i>Jenifer_Lopez</i> )   Lonely (T, <i>Britney_Spears</i> )   Circus (R, <i>Britney_Spears</i> )	initial user's profile
--	------------------------

Table 2. Music items from user’s first session.

<b>page 1</b> [no of visits: 2]	Nobody_Knows_Me (1)   American_Life (1)   Justin_Timberlake (1)   Kanye_West (1)   Pink (1)   Madonna   Celine_Dion
<b>page 2</b> [no of visits: 1]	Britney_Spears (3)   Madonna   pop   Mariah_Carey
<b>page 3</b> [no of visits: 1]	Nobody_Knows_Me (1)   Britney_Spears (1)   American_Life (1)   Timberland (1)   Madonna (1)   Hard_Candy (1)   The_Emancipation_of_Mimi (1)   Cher (1)   Back_to_Basics (1)
<b>page 4</b> [no of visits: 3]	Christina_Aguilera (1)   Back_to_Basics (1)
<b>page 5</b> [no of visits: 2]	The_Emancipation_of_Mimi (1)   Mariah_Carey (1)   Queen (1)   pop (1)   pop-rock (1)
<b>page 6</b> [no of visits: 1]	The_Emancipation_of_Mimi (1)   Mariah_Carey (1)     Britney_Spears (1)
<b>page 7</b> [no of visits: 1]	Britney_Spears (1)   Christina_Aguilera (1)   Madonna (1)
<b>page 8</b> [no of visits: 1]	Britney_Spears (2)   Christina_Aguilera (2)   Back_to_Basics (2)

Table 3. Rules fired for the items from the visited pages when compared against the items from the user profile (A-artist, T-track, R-record, the name in italic represents an artist associated with a given track or record).

web page items	items from user's profile							
	Madonna (A)	Mariah Carey (A)	Die Another Day (T) <i>Madonna</i>	For the Record (T) <i>Carey</i>	Hard Candy (R) <i>Madonna</i>	I Got U (T) <i>Lopez</i>	Lonely (T) <i>Spears</i>	Circus (R) <i>Spears</i>
<b>Nobody Knows Me</b> (2)(T) <i>Madonna</i>	TA1,4	TA1,2	TT1,2,3,4	TT3,4	RT1,2	TT3,4	TT3,4	RT2
<b>Britney Spears</b> (5)(A)	AA2,3,4,5	AA4,5	TA1,2,3	TA1,2	RA1	TA1	TA1,4	RA1,2
<b>American Life</b> (4)(R) <i>Madonna</i>	RA1,2	RA1	RT1,2	RT2	RR1,2	RT2	RT2	RR2,3
<b>Christina Aguilera</b> (3)(A)	AA4,5	AA4,5	TA1,2	TA1,2	RA1	TA1,2	TA1,2	RA1
<b>The Emancipation of Mimi</b> (3)(R)*	RA1	RA1,2	RT2	RT1,2	RR2,3	RT2	RT2	RR2,3
<b>Back to Basics</b> (3)(R) <i>Aguilera</i>	RA1	RA1	RT2	RT2	RR2,3	RT2	RT2	RR2,3
<b>Pink</b> (1)(A)	AA4,5	AA4,5	TA1,2	TA1,2	RA1	TA1,2	TA1,2	RA1
<b>Justin Timberlake</b> (1)(A)	AA2,3,4,5	AA4,5	TA1,2,3	TA1,2	RA1	TA1,2	TA1,2,3	RA1
<b>Kanye West</b> (1)(A)	AA2,4,5	AA2,4,5	TA1,2,3	TA1,2	RA1	TA1,2	TA1,2	RA1
<b>Timberland</b> (1)(A)	AA2,4,5	AA4,5	TA1,2,3	TA1,2	RA1	TA1,2	TA1,2	RA1

\* Mariah Carey

The similarity levels associated with fired rules are aggregated at the individual level, as well as at the global level (Section 3.B). The application of three different linguistic quantifiers at both levels has led to nine different combinations. The obtained similarity values are presented in Table 4. The OR quantifier used at the individual level (columns 1, 2, and 3) leads to a very “forgiving” results. Even the application of any other quantifier at the global level – OR, SOME, MOST – has not brought any discrimination among web page items. All similarity values are above the threshold 0.5. Columns 4, 5 and 6 of Table 4 contain similarity values when SOME quantifier is used at the individual level. In this case, we see more pronounced differences between similarity values. A very similar situation occurs when

MOST is used at the individual level – columns 7, 8 and 9. A close comparison of the results, and a number of discussions involving experts and users have led us to the selection of MOST at the individual, and OR at the global level as the two quantifiers to be applied during further experiments. The selected similarity values for the items from the first session are presented in Table 5.

The Naïve Bayes approach (Section 4.D) has been also applied to the items from eight user-accessed pages. In this case, we use the following formula

$$\prod_i \text{similarity\_rule}_{k,i}$$

for  $k=1, \dots, 10$ . The product of similarities associated with the rules fired for a single item (a row from Table 3) is used to estimate the similarity between a given item and the user's profile.

Table 4. Semantic similarity values for the items from the web pages (all possible combinations of OR, SOME, and MOST applied at individual and global levels).

web page items	similarity to user's profile								
	individual aggregation			individual aggregation			individual aggregation		
	OR			SOME			MOST		
	global aggregation:			global aggregation:			global aggregation:		
	_OR_	SOME	MOST	_OR_	SOME	MOST	_OR_	SOME	MOST
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	
<b>Nobody Knows Me</b> (2)(T) <i>Madonna</i>	1	.73	.62	1	.55	.41	1	.42	.26
<b>Britney Spears</b> (5)(A)	1	.72	.80	1	.53	.58	1	.39	.43
<b>American Life</b> (4)(R) <i>Madonna</i>	1	.70	.62	1	.58	.50	1	.47	.37
<b>Christina Aguilera</b> (3)(A)	.67	.60	.65	.46	.39	.42	.26	.22	.24
<b>The Emancipation of Mimi</b> (3)(R)*	1	.64	.65	1	.54	.49	1	.45	.39
<b>Back to Basics</b> (3)(R) <i>Aguilera</i>	.63	.53	.65	.47	.38	.39	.37	.25	.27
<b>Pink</b> (1)(A)	.67	.60	.65	.46	.39	.42	.26	.22	.24
<b>Justin Timberlake</b> (1)(A)	.67	.63	.65	.67	.48	.47	.58	.34	.31
<b>Kanye West</b> (1)(A)	.67	.66	.65	.63	.47	.45	.51	.29	.27
<b>Timberland</b> (1)(A)	.67	.63	.65	.63	.44	.43	.51	.27	.26

\* Mariah Carey

The values obtained for Bayes approach are shown in Table 6 (the second column represents the values normalized by the maximum value). The sequence of items is the same as in Table 5, and it can be observed that the ranking of items based on the similarity values obtained with Bayes-based approach is different then the ranking obtained using the proposed method (Table 5). However, we can see that the item *American Life* obtained the highest score using both methods, while the similarities of the rest of items are quite different (ranking-wise) between the methods. The Bayes-based approach seems to be more “demanding” – only high values of similarity measures for all fired rules would lead to the high similarity between the item and the use’s profile. This is fully justified when we realize that the Bayes-based approach as a special case of OWA [27] with ALL as the linguistic quantifier – and this means that all similarities associated with the rules have to be high to obtain a high value of the overall similarity.

Table 5. The items from the web pages sorted based on similarity values: OR(MOST) approach.

web page items	similarity
Nobody Knows Me	1
Britney Spears	1
American Life	1
The Emancipation of Mimi	1
Justin Timberlake	.58
Kanye West	.51
Timberland	.51
Back to Basics	.37
Christina Aguilera	.26
Pink	.26

Once the similarity levels of items from user-accessed web pages are estimated (Tables 5 and 6), the process focuses on estimating importance of those items. Table 7 represents numbers of occurrences of the items on the visited pages. The last column contains calculated – based on the formula (4) from Section 3.C – importance of the items. It can be observed that except four items, the values of importance are below or equal to 0.20. A possible reason for that is existence of two multi-item pages – page #1 which was visited twice, and page #3.

Table 6. The items from the web pages with their similarity values: naïve Bayes classifier.

web page items	similarity $\times 10^{-3}$	similarity (normalized by max 0.15729)
Nobody Knows Me	.02204	.1401
Britney Spears	.01008	.0641
American Life	.15729	1
The Emancipation of Mimi	.02916	.1854
Justin Timberlake	.05898	.3750
Kanye West	.09216	.5859
Timberland	.02916	.1854
Back to Basics	.00140	.0089
Christina Aguilera	.00420	.0267
Pink	.00700	.0445

Both estimated values – similarity and importance are inputs to a very simple fuzzy inference system. The rule used in this inference is (Section 3.B):

**IF** *similarity* is high and *importance* is high  
**OR** *similarity* is medium and *importance* is high  
**THEN** *level\_of\_relevance* is high

Table 7. Number of occurrences of items on the visited pages, and importance of those items (please note that pages #1, #4 and #5 are visited multiple times, the values in the cells represent a number of occurrences times a number of visits).

web page items	visited page								importance
	#1 (x2)	#2	#3	#4 (x3)	#5 (x2)	#6	#7	#8	
Nobody Knows Me	2(1)		1						3/16 (0.19)
Britney Spears		3	1			1	1	2	8/19 ( <b>0.42</b> )
American Life	2(1)		1						3/16 (0.19)
The Emancipation of Mimi			1		2(1)	1			4/10 ( <b>0.40</b> )
Justin Timberlake	2(1)								2/10 (0.20)
Kanye West	2(1)								2/10 (0.20)
Timberland			1						1/6 (0.17)
Back to Basics			1	3(1)				2	6/18 ( <b>0.33</b> )
Christina Aguilera				3(1)			1	2	6/14 ( <b>0.43</b> )
Pink	2(1)								2/10 (0.20)
#items per page	10	3	6	6	2	2	2	6	

The membership functions associated with linguistic labels used in the rule are presented in Figure 6. The user determines the parameter values of those functions. The values reflect user’s perception of *medium* and *high* similarity, *high* importance, and *high* relevance. In the shown example for the *high* importance (Figure 6 b), the

“cut-off” value is 0.3 what means that any item with the value below 0.3 (Table 7) is not even considered for a possible addition to the profile. Similar thing can be said about relevance (Figure 6 c). The values 0.3 and 0.9 should make the user comfortable with identifying relevant items. In the experiments described here, we

“obtained” user’s approval for those values. The rules represent a “common sense” approach for expressing what it means high importance or high relevance. The user can make adjustments, so the rules reflect her perception of those concepts.

The final results – once the similarity and importance are “pushed” through the inference mechanism – are presented in Table 8. It shows that only four items have been identified as relevant. The most interesting aspect of that recommendation is the exclusion of some top items (*Nobody Knows Me*, and *American Life*), and inclusion of some less similar items (*Back to Basics*, and *Christina Aguilera*). Such results are a combination of similarity and importance. In the case of the last two items (*Back to Basics*, and *Christina Aguilera*), their importance values have been quite high, and the fuzzy rule (Section 3.B) includes the antecedent: “*similarity* is medium and *importance* is high”.

For the similarities obtained using Bayes-based approach none of items obtained non-zero relevancy value. The items with the highest Bayes-based similarity values: *American Life*, *Justin Timberlake* and *Kanye West* have the values of importance below 0.3.

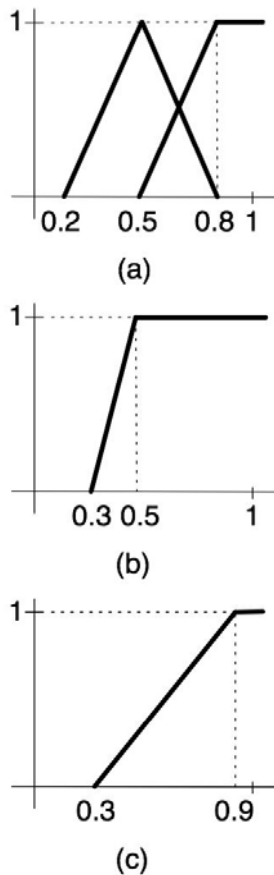


Figure 6. Fuzzy membership functions for: *medium* and *high* similarity (a), for *high* importance (b), and for *high* relevance (c).

Table 8. Relevancy values for items from visited web pages.

web page items	relevancy
Nobody Knows Me	0
<b>Britney Spears</b>	<b>0.73</b>
American Life	0
<b>The Emancipation of Mimi</b>	<b>0.73</b>
Justin Timberlake	0
Kanye West	0
Timberland	0
<b>Back to Basics</b>	<b>0.68</b>
<b>Christina Aguilera</b>	<b>0.68</b>
Pink	0

### C. Further Results and Discussion

Similar process is performed for the second and third session, Table 9 and Table 10. Each session “finds” a smaller number of relevant items: the second session results in three, and the third in two items added, Table 11. The user has verified additions, and indicated that she would do the same choices. Similar results were obtained with other users.

Table 9. Music items from user’s second session.

<b>page 1</b> [no of visits: 1]	Candy_Shop   Beat_Goes_on   Die_Another_Day   Four_Minutes   Shes_Not_Me   Devil_Wouldnt_Recognize_You   Spanish_Lesson   Miles_Away   Hung_Up   Give_It_to_Me
<b>page 2</b> [no of visits: 2]	The_Emancipation_of_Mimi   Mariah_Carey   Madonna   Voices
<b>page 3</b> [no of visits: 3]	50_Cent   Beyonce   Mariah_Carey   Snoop_Dogg   Ludacris   Nelly   hip-hop   Justin_Timberlake   Kid_Rock   Travis_Barker   Christina_Aguilera   Fred_Durst   Hilary_Duff   Taylor_Swift   JC_Chasez
<b>page 4</b> [no of visits: 3]	Madonna   Hard_Candy   Rock_and_Roll   Mariah_Carey   4_Minutes   Justin_Timberlake   Me_Against_the_Music   Britney_Spears   Timbaland   Pharrell_Williams   Kanye_West   E=MC2   Leona_Lewis   Lyfe_Jennings   Lyfe_Change   Spirit   Leona_Lewis   Rising_Down   the_Roots   Third   Portishead   Mudcrutch   Greatest_Hits   Tim_McGraw

Table 10. Music items from user’s third session.

<b>page 1</b> [no of visits: 1]	Queen   The_cosmos_Rocks   rock   Brian_May   Roger_Taylor   Innuendo   The_Miracle   Sheer_Heart_Attack   C-lebrity   Small   Voodoo   Warboys   Call_Me
<b>page 2</b> [no of visits: 3]	Mariah_Carey   Christina_Aguilera   Miley_Cyrus   Kanye_West   Alicia_Keys   Rihanna   New_Kids_on_the_Block   Pussycat_Dolls   Taylor_Swift   Pink   The_Fray   Annie_Lennox   Colbie_Caillat   Flo_Rida   Jonas_Brothers   Paramore   The_Dream
<b>page 3</b> [no of visits: 2]	Hard_rock   Timbaland   rock   Chris_Cornell   R&B   Hip-hop   Justin_Timberlake   Nelly_Furtado   Madonna   Ryan_Tedder   hard-rock
<b>page 4</b> [no of visits: 1]	88_Keys   Kanye_West   The_Death_of_Adam   Pink

## 6. Conclusions

The paper focuses on the issue of automatic updating of a user profile. The method used for that purpose is based on combining two measures: 1) similarity of items

from the user-accessed web pages to the items form the user profile, and 2) importance of the web items. The approach treats both factors – similarity and importance – as equally vital for estimating the relevancy of items to user’s interests. The proposed semantic similarity measure is defined based on a set of rules built using a domain ontology. The similarity levels associated with fired rules, for a given pair of items, are further combined using OWA operator. The weights of OWA are derived from different linguistic quantifiers – like OR, SOME, MOST – making the approach more human-like. The importance of items is estimated based on statistical information obtained from user web-access data. The proposed relevance measure includes the general view of the knowledge domain (similarity as perceived by a domain ontology), user’s perception of similarity (linguistic quantifiers OR, SOME, MOST), and the user’s estimated interests (statistical information obtained from the contents of web pages browsed by the user).

The relevancy measure is applied to the process of updating a user profile in the music domain. The music domain knowledge base has been created by customizing the well-known music ontology, and populated it using online music databanks (MusicBrainz, and Wikipedia). The results of the real-world experiment using the proposed and Bayes-based approaches together with their detailed descriptions are presented.

Table 11. Music items added to user’s profile after each session.

Madonna   Mariah_Carey   Die_Another_Day   For_the_Record   Hard_Candy   I_Got_U   Lonely   Circus	initial user's profile
Britney Spears Back_to_Basics Christina_Aguilera The_Emanicipation_of_Mimi	added after session 1
4_Minutes Justin_Timberlake E=MC <sup>2</sup>	added after session 2
Kanye_West Pink	added after session 3

### Acknowledgment

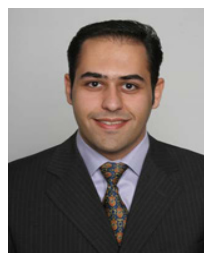
The authors would like to acknowledge the support of Natural Sciences and Engineering Research Council of Canada (NSERC).

### References

- [1] G. Adomavicius, G., and A. Tuzhilin, A., “Personalization technologies: A process-oriented perspective,” *Communications of the ACM*, vol. 48. no.10, pp. 83-9, 2005.
- [2] R. Armstrong, D. Freitag, T. Joachims, and T. Mitchell, “WebWatcher: A learning apprentice for the world wide web,” in *Proc. of AAAI Spring Symp. on Inf. Gathering from Heterogeneous, Dist. Environments*, AAAI Press, pp. 6-12, 1995.
- [3] R. Burke, “Hybrid Recommender Systems: Survey and Experiments,” *User Modeling and User-Adapted Interaction*, vol. 12, no. 4, pp. 331-370, 2002.
- [4] J. Cho, K. Kwon, and Y. Park, “Collaborative Filtering Using Dual Information Sources,” *IEEE Intelligent Systems*, vol. 22, no. 3, pp. 30-38, 2007.
- [5] D. Ferrucci and A. Lally. Uima, “An architectural approach to unstructured information processing in the corporate research environment,” *Natural Language Engineering*, vol. 10 no.3-4, pp. 327-348, 2004.
- [6] T. R. Gruber, “A translation approach to portable ontology specifications,” *Knowledge Acquisition*, vol. 5, pp. 199-220, 1993.
- [7] I. Horrocks, and P. F. Patel-Schneider, “Proposal for OWL Rule Language,” *13th Int. WWW Conference*, pp. 723-731, 2004.
- [8] J. J. Jiang, and D. W. Conrath, “Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy,” in *Proc. ROCLING*, Taiwan, 1997.
- [9] K. Lakiotaki, P. Delias, and V. Sakkalis, N. F., “User profiling based on multi-criteria analysis: The role of utility functions,” *Operational Research*, vol. 9, no. 1, pp. 3-16, 2009.
- [10] S. E. Middleton, N. Shadbolt, and D. C. De Roure, “Ontological user profiling in recommender systems,” *ACM Transactions on Information Systems*, vol. 22, pp. 54-88, 2004.
- [11] V. Loia, W. Pedrycz, S. Senatore, and M. I. Sessa, “Interactive Knowledge Management for Agent-assisted Web Navigation,” *International Journal of Intelligent Systems*, vol. 22, no. 10, pp. 1101-1122, 2007.
- [12] M. Pazzani, J. Muramatsu and D. Billsus, “Syskill & webert: Identifying interesting web sites,” in *Proc. of AAAI Spring Symposium on Machine Learning in Information Access*, Portland, Oregon, 1996.
- [13] W. Pedrycz, and F. Gomide, “Fuzzy Systems Engineering: Toward Human-Centric Computing,” *Wiley-IEEE Press*, 2007.
- [14] Y. Raimond, and S. A. Abdallah, The timeline ontology, OWL-DL ontology, <http://purl.org/NET/c4dm/timeline.owl>, 2006.
- [15] Y. Raimond, S. Abdallah, M. Sandler, and F. Giasson, “The music ontology,” in *Proc. of the Intern. Conf. on Music Information Retrieval*, pp. 417-422, September 2007.

- [16] P. Resnik, "Development and application of a metric on semantic nets," *IEEE Trans. on SMC*, vol. 19, pp. 17-30, 1989.
- [17] P. Resnik, "Semantic Similarity in a Taxonomy: An Information-Based," *J of Artificial Intelligence Research*, vol. 11, pp. 95-130, 1999.
- [18] M. A. Rodriguez, and M. J. Egenhofer, "Determining Semantic Similarity among Entity Classes from Different Ontologies," *IEEE Trans. on Knowledge and Data Engineering*, vol. 15, pp. 442-456, 2003.
- [19] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, pp. 1-47, 2002.
- [20] M. Shanahan, "The event calculus explained, in Artificial Intelligence Today," *Lecture Notes in AI no. 1600*, Springer, pp. 409-430, 1999.
- [21] X. Shen, B. Tan, and C. Zhai, "Ucair: Capturing and exploiting context for personalized search," in *Proc. of the Information Retrieval in Context Workshop, SIGIR IRiX*, Brazil, 2005.
- [22] A. Sheth, I. B. Arpinar, and V. Kashyap, "Relationships at the heart of Semantic Web: modeling, discovering and exploiting complex semantic relationships," in *Enhancing the Power of the Web*, Heidelberg: Springer, pp. 63-94, 2004.
- [23] A. Sieg, B. Mobasher, R. Burke, "Web Search Personalization with Ontological User Profiles," in *Proc. of the 16<sup>th</sup> ACM Conf. on information and knowledge management*, Portugal, pp. 525-534, 2007.
- [24] B. Towle, and C. Quinn, "Knowledge Based Recommender Systems Using Explicit User Models, *Knowledge-Based Electronic Markets*," the *AAAI Workshop*, Menlo Park, CA: AAAI Press, pp. 74-77, 2000.
- [25] J. Trajkova, and S. Gauch, "Improving ontology-based user profiles," in *Proc. of RIAO*, Vaucluse, France, pp. 380-389, 2004.
- [26] R. R. Yager, "On ordered weighted averaging aggregation operators in multi-criteria decision making," *IEEE Trans. SMC*, vol. 18, pp. 183-190, 1988.
- [27] R. R. Yager, "An Extension of the Naïve Bayesian Classifier," *Information Sciences*, vol. 176, pp. 577-588, 2006.
- [28] Q. Zhang, J.C.H. Chen, and P.P Chong, "Decision consolidation: criteria weight determination using multiple-preference formats," *Decision Support Systems*, vol. 38, pp. 247-258, 2004.
- [29] MusicBrainz: <http://musicbrainz.com/>
- [30] OWL: [http://www.w3.org/2007/OWL/wiki/OWL\\_Working\\_Group](http://www.w3.org/2007/OWL/wiki/OWL_Working_Group)
- [31] RDF: <http://www.w3.org/RDF/>

in the area of applications of Computational Intelligence techniques, as well as probabilistic and evidence theories applied to intelligent data analysis leading to the transformation of data into coherent pieces of evidence. He is also interested in the development of ontology and ontology-based forms of knowledge representation. He actively pursues research in combining ontology with fuzziness, utilization of ontology for similarity estimation, and ontology-based search mechanisms. Dr. Reformat has been a member of program committees of several conferences related to Computational Intelligence, evolutionary computing, and software engineering. He is actively involved in North American Fuzzy Information Processing Society (NAFIPS). He is a member of the IEEE and ACM.



#### Seyed Koosha Golmohammadi

received his BSc degree from Azad University, Iran, and his MSc degree from University of Alberta. Currently he is working towards his PhD degree in Computing Science Department at University of Alberta. His research interests are Intelligent Agents, Fuzzy Logic, and Semantic Web Services. He was actively involved in RoboCup Competitions since late 2004. He participated in almost all competitions since then and won many awards in soccer simulation league. He is a member of ACM and IEEE Computational Intelligence Society.



**Marek Z. Reformat** received his M.Sc. degree (with honors) from Technical University of Poznan, Poland, and his Ph.D. from University of Manitoba, Canada. He is with the Department of Electrical and Computer Engineering at the University of Alberta. His interests lie