Serdica J. Computing **11** (2017), No 3–4, 297–320

Serdica
Journal of Computing

Bulgarian Academy of Sciences
Institute of Mathematics and Informatics

SEMANTIC-ORIENTED ARCHITECTURES AND USE OF ONTOLOGY FOR ORGANIZING ADAPTIVE SEARCH IN DIGITAL LIBRARIES*

Albena Turnina

Abstract. Over time, the number of knowledge systems is growing and gaining popularity in various fields such as business, science, training, and etc. Their use has been particularly beneficial for the learning process. The use of a digital library is growing over time because the information stored and delivered is reliable, structured and ready to use. Learners not always have necessary knowledge and skills to search for information in a digital library, as their experience in the information space is limited. This necessitates the development of functionalities that facilitate the process of search in a digital library, like semantic-based search, multi-criteria search, contextual search, adaptive search, and etc. The article presents the architecture of a Functional Module for Adaptive Search in a digital library and its main components. The basic scheme of adaptive

---

ACM Computing Classification System (1998): H.3.7, K.3.1.

Key words: semantic based knowledge systems, digital libraries.

*This article presents the principal results of the author's doctoral thesis *Semantic-oriented architectures and use of ontology for organizing adaptive search in Digital libraries*, successfully defended at the Faculty of Mathematics and Informatics, St. Kliment Ohridski University of Sofia, on December 2, 2016.

and customizable logic is also presented. The article ends with directions for future development of the architecture.

1. Introduction. The increasing number of information resources in Digital Library creates difficulties for users to search and find a needed content. Adaptive Search is type of search in which the system facilitates users by selecting the results most relevant for them and/or rearranging those results according to the preferences of a particular user, group of users, or all users of the system. Adaptive search is implemented through a number of methods, including: adapting the query, reordering of results, or evaluating the results. At the core of the adaptive search is a model describing user knowledge. Semantic-based architectures make it possible to use a unified vocabulary for describing objects in the domain based on ontology. Ontology can describe both the metadata used for annotation of the data and the data itself. The users of the system can be presented by an individual model or by a group model (stereotype) which represents the characteristics of a given group of users. In many cases detailed knowledge is not required from the users, as the same adaptive strategies can be applied to a group of users. In those cases the use of a group model is more suitable, because of its simplicity and efficiency. Stereotypes are used to describe a group of users with similar characteristics and preferences over which the common rules of adaptation can be applied. The use of stereotypes for user modeling finds its widest application in expert systems and in training systems [15].

User data in the system may be insecure or inaccurate, which requires the use of methods and techniques to address that uncertainty. Key technologies used for inference under uncertainty are Bayesian Networks (BN) and Fuzzy Logic. The BN model allows for inference in both diagnostic and prognostic directions. This ability of BN for inference in both directions is very useful in modeling learner knowledge. The inference over user knowledge via Bayesian models is the most common in adaptive learning systems.

In a typical system, the search interface provides a starting point for users from where they begin their search in the system. It is known that users usually start their search with very short queries, inspect the results, and then modify the queries in a gradual refinement cycle [1]. Therefore, the main task of the adaptive search is to help the user get the right resources from the start.

Simple keyword queries are useful when the users have a clear idea of what they are looking for and know what keywords to use when searching. In knowledge systems the level of knowledge of the subject area is often low. This makes it necessary to transform the original user's query into a query that recognizes and uses the established semantics in the field. These problems can find their solution by building such knowledge systems that actively support users by using leading user modeling techniques and adaptation strategies for inference over user model.

2. Theoretical Definition. Adaptability is related to the overall ability of the system to adapt to a user. Adaptability can be explicitly controlled by a user or provided implicitly by the system by analyzing user behavior and interaction with the system [17]. Adaptability is implemented through a set of methods and technologies for collecting and analyzing knowledge for the user and implementation of adaptive strategies. This is done by: (1) collecting user statistics, (2) building a user model, (3) analyzing the user model, (4) determining how the system can best adapt to the user, and (5) the adaptability process itself.

Group adaptability is adaptability that affects interaction with a particular user, but is influenced by interactions of many users. In group adaptability users are split into groups based on one or more of their characteristics.

In general, adaptability is achieved by applying adaptive methods and strategies over a user model. There are different approaches to user modeling, the chief among which are: (1) classification of users into stereotype groups (stereotypes); (2) use of techniques for learning the user model based on their behavior in the system; (3) identification of similarities between a new user and existing users in the system; etc. Adaptability is applied in various types of systems, such as: intelligent learning systems, information content access systems, electronic catalogs, health care systems, recommender systems, and more. In hypermedia systems adaptability is implemented as: content adaptation, adaptation of presentation or adaptation of the structure [11]. The two main personalization approaches are content-based and collaborative-based. The process of adaptability and personalization in knowledge systems aims at identifying and presenting to users the information resources that are the most relevant to them [20]. In e-learning systems, adaptability is geared towards identifying key learning resources for learners. In these systems, personalization is mainly understood as personalized access to learning resources. Among the systems for which adaptability and personalization are essential are digital libraries.

Digital libraries are online knowledge systems in which digital objects are collected, classified, and managed in accordance with established principles [21; 24].

Adaptability and personalization in digital libraries takes place in the following ways:

- adaptive search of information resources, based on the individuality of the user;
- personalized presentation of the content and the graphical interface of the system;
- recommended content according to user characteristics;
- personalization of the services offered by the system in accordance with the user's preferences;
- personalized grouping and aggregation of information resources on the basis of various attributes [18];
- adaptation based on the used devices—desktop, mobile, etc.;
- adaptation based on the location of the user and the proximity to objects of importance, such as art monuments, architecture, etc.;
- adaptation based on environmental features such as climate, time of day, etc.;
- adaptation based on information created or viewed by the user.

The user modeling process is related to a selection of his essential characteristics and their representation in a model. Such characteristics include: demographic data, such as age and gender; level of knowledge in the domain; cognitive and mental characteristics; way of perception and learning; etc. There are various approaches to selecting these characteristics; Brusilovsky's classification is widely known [3]. In adaptive e-learning systems and intelligent tutoring systems, essential user characteristics are: way of learning, ease of learning and cognitive abilities. There are two main standards for learner modeling—IEEE PAPI Learner [6] and IMS LIP [7]. These standards serve to create a unified learner model and ensure compatibility and reusability of models in different e-learning systems.

SAPS (Systems for Adaptive and Personalized Search) usually employ the user's searches for their modeling. In them the user's model is populated with indirectly collected data, consisting of terms of his previous searches. Other systems also use that approach. There are systems that use explicit data provided by the user in the system to build the user's model. These different approaches can be combined. The simplest representation of a user's model in SAPS is an n-dimensional vector composed of search terms. In this type of model, terms are usually characterized by a certain weight that indicates the importance of the term for the particular user. The vector model uses a simple structure in which there are no meaningful relations between the individual terms. The simplest implementation of this model uses only one vector that contains all the terms that are relevant to the user modeling. In more sophisticated implementations different aspects of the user are represented by separate vectors.

The advantage of the vector model compared to other types of models is its simplicity, computational efficiency and proven performance. The representation of a user model as a vector of terms is closely related to the vector-space model known from Information retrieval. In the vector-space model each document and each user are represented as a set of terms or as n-dimensional vectors. A vector-space model is used for measuring the similarity between vectors of resources and vectors of users. If the similarity of the vectors is within a given threshold, it is assumed that they are similar. The vector-space model finds a broad application in measuring similarity between a search query and the available documents in a given resource collection. One of the most popular methods for measuring the similarity between two vectors in the field of Information retrieval is the so-called cosine similarity [19].

The stereotype is considered an aggregate model of a group of users. This means that if two users belong to the same stereotype, they belong to a common group, behave similarly and have similar interests. Stereotypes present a set of characteristics that a group is assumed to possess. Stereotypes were originally introduced and developed by Rich [15, 16] and subsequently widely used for modeling a user in heterogeneous types of systems. A stereotype can be considered as a collection of features designed to describe a frequently occurring situation. The use of stereotypes is common in information filtering systems as a method of dividing users into groups based on their common characteristics. In most cases, user modeling with stereotypes is used when the system does not have enough user data, for example when the user is new to the system. In other cases, stereotypes are used to divide users into groups over which common adaptive rules are applied.

Adaptive search aims at building systems capable of providing an individual collection of results for different users. The basis of these systems is a model describing user knowledge. Adaptive search is the process of selecting the most relevant results and their ranking depending on a particular user, group of users or all users of the system. It is implemented through a set of approaches, including: query adaptation, reordering of results and evaluation of results. Adaptive search is based on a set of methods, techniques and strategies such as: search history, user models, collaborative approaches, clustering of results, hypertext content and current context [12].

The main methods of adaptive search are: (1) adaptation of the query; (2) reordering of returned results and (3) a combination of both. Adaptation in multi-lingual systems also includes translation of the query and result across languages [13].

The adaptation of a query leads to a new query that is considered to better reflect the user's needs and interests. This method of adaptation is useful when the user lacks knowledge of the subject area.

The main methods for query adaptation are: (1) modifying the query, (2) expanding the query with new terms and (3) assigning the weights to terms in the query. Query modification is implemented by replacing original terms with terms that better represent the user's interests. Query expansion is related to addition of new terms to the original query. In this way, short queries consisting of only few terms are complemented with new, more precise terms. The query can be adapted not only by terms but also by concepts or categories. In the most common case, those terms and categories are taken from the user model. They can also come from reference dictionaries, taxonomies, or domain ontologies.

The main technologies used for inference under uncertainty are BN (Bayesian Networks) and Fuzzy Logic. Bayesian networks are capable of expressing a relationship between a state and its symptoms or predicting the most likely outcome of a situation or event. This ability of BN to infer in both directions is very useful in modeling learner knowledge. BN is defined as a directed acyclic graph represented by the ordered pair (N, E), where N is a multitude of nodes and E is a multitude of arcs connecting the nodes. Each node of N contains a variable name and a CPT (conditional probability table) that shows the dependence of this variable on its predecessors. There are arcs only between nodes corresponding to variables that are not conditionally independent [23]. According to the definition by Pearl [14], BN is a directed acyclic graph in which each node presents a random variable and each arc represents a probable relationship between two variables. If there is an arc between two nodes, this means that there is dependence between the two variables, which would be expressed by a suitable implication.

BN is able to calculate the *a posteriori* probabilities of uncertain variables based on evidence obtained from related variables. This process is known as distribution of evidence. This property, along with BN's ability to model complex relationships between uncertain variables, makes it very suitable for building diagnostic models. The Bayesian model allows not only an accurate diagnosis of the learner's current cognitive level but also an informed recommendation of the most appropriate next activity (exercises, tests) to be undertaken. The inference over user knowledge via Bayesian models is the most common in Adaptive Learning Systems.

In real-world situations there are cases when multiple causes can cause the same effect [8]. To overcome these situations, Henrion [9, 10] proposes a model in which, apart from the variables of the effect Y and the causes $\{X_1, X_2, ..., X_n\}$, intermediate variables $\{I_1, I_2, ..., I_n\}$ also exist which represent the independent contribution of each cause to the effect. In this way, the effect Y is presented as a deterministic function of these intermediate variables.

The size of CPT presents a problem in big BN, because it increases exponentially to the number of variables preceding the given one. To face this problem new canonical models have been developed that approximate the CPT tables and require fewer parameters. Complexity of reasoning is avoided by using canonical interactions such as NoisyOR, NoisyMAX and NoisyAdder, which limit the expressiveness of the model [14]. The NoisyOR gateway can only be used for binary variables. To overcome this limitation the NoisyMAX gateway has been developed [5, 10]. Henrion [10] proposes an extension of NoisyOR to model the cases where the effect Y can occur even though all possible causes are absent. This extension is called LEAK and reflects the phenomenon that the effect Y may occur spontaneously in the absence of any concrete cause.

Another model representing the cause-effect interaction is NoisyAdder. This model consists of binary variables $\{X_1, X_2, ..., X_n\}$ and integer variables $\{I_1, I_2, ..., I_n\}$ such that I_i can take values from -1 to $+1$.

Existing Bayesian models are of varying complexity. In diagnostic models, the simplest model is based on the assumptions that only one malfunction can occur at a time and causes are conditionally independent. More realistic Bayesian models are those with more than one fault variable.

3. Architecture of a functional module for adaptive search in a digital library. A functional module for adaptive search facilitates users in their searches for information resources in the system. The presented architecture offers automated and transparent adaptation of the user's queries. This is implemented by expanding the user's query with terms taken from *Model for adaptation and personalization*. The Model is a 3-layer BN, described in detail in the section below. The adaptive logic behind the architecture is presented as well.

3.1. Basic scheme of adaptive and personalization logic for adaptive search. The basic scheme is presented in Fig. 1. It shows the main activities of the user and respective responses of the system.

As a first step the user submits data about himself (through a registration form), such as: goals in the system, language/s, areas of knowledge, institution, role, etc. Possible data values are predefined on the basis of elements of the TEO ontology [2; 22]. Those elements are used in the registration form as data lists and drop-down menus. In the second step a user model (user vector)

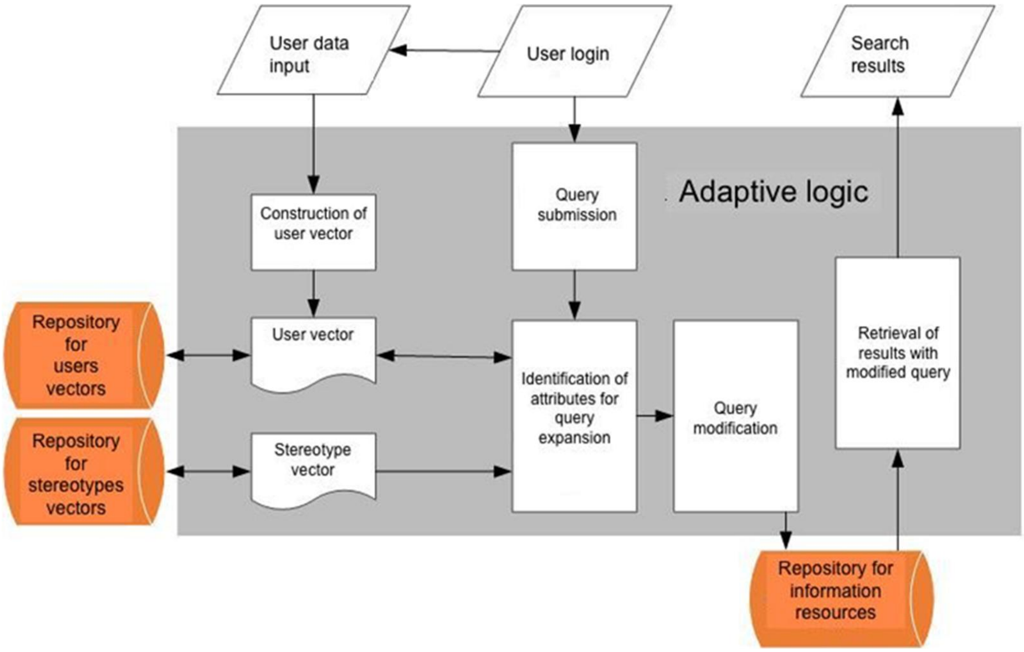

Fig. 1. Basic scheme of adaptive and personalization logic for adaptive search

is created from the submitted data. This vector consists of five elements that can have either a value provided by the user or no value at all (in case of absence of data). The vectors of all users are stored in the system. In the next step the stereotype's vectors are loaded from the repository. Each stereotype is represented as a weighted vector composed of the same type of elements as the user's vectors in the system. This representation allows a comparison between stereotype's vectors and user's vector. As a result of that comparison each user of the system is classified to given stereotype/s. In the next step the stereotype/s to which the user is classified is/are submitted to the *Model for adaptation and personalization* for inference. The inference is based on providing evidence (True/False) to the nodes of the higher layer of the model (which represents stereotypes) and running clustering algorithm for distribution of that evidence to the lower layer of the model. As a result of the inference a list of terms for query adaptation is returned. The provided terms are saved in the user model. Another important part of the scheme is submission of a simple search query by a user. There is a process of replacing the user's original query with an adapted query. The adapted query is sent to the system instead of the original query. As a result, the system returns results that are tailored to the user. In the last step the retrieved search results are presented to the user.

3.2. Components in the architecture of a functional module for adaptive search in a digital library. The components in the architecture of the Functional Module for Adaptive Search are a set of main modules and their sub-modules that act together and consistently to achieve the intended goal of implementing adaptive search. The main modules and their sub-modules are as follows:

- Module for presentation and management of knowledge for the user. This module consists of three sub-modules:
    - Sub-module for storage of the stereotypes in the system. This module ensures the storage of stereotypes in the system.
    - Sub-module for presentation of knowledge for the user.
    - Sub-module for calculating similarity between user and stereotypes (classification of user to stereotype/s). The vector-space model is used and the similarity is measured as cosine similarity between two vectors.
- Module for implementation of adaptive and personalization logic. A central component of this module is the *Model for adaptation and personalization.*
- Module for applying adaptive search. This module receives as an input submitted user query and expansion terms, provided by the *Model for adaptation and personalization.* It produces a modified query and returns personalized search results to the user.

A brief description of the components is presented in the section below.

3.2.1. Module for implementation of adaptive and personalization logic. This module is the component of the architecture of the Functional Module containing adaptive logic. The central component in it is the *Model for adaptation and personalization* who serves as a machine for inference over user knowledge.

Model for adaptation and personalization

The Model is a three-layer BN, composed of nodes and directed, weighted arcs. The variables of each layer are independent of each other and therefore are not linked by arcs. The variables of the top layer represent the *stereotypes* in the Model, the variables of the medium layer are the *needs* and the variables of the bottom layer are the *attributes*. The variables of the types *stereotypes*, *needs*, and *attributes* are binary, and therefore can receive only true or false values. Initially, *a priori* probabilities with p (x = true) = 0.2 and p (x = false) = 0.8 are assigned to the variables of the top layer (*stereotypes*). These *a priori* probabilities serve to pre-determine the conditional probabilities of the variables of the other two layers in the Model. Subsequently, the *a priori* probabilities are replaced by evidence with true or false values. Based on that evidence and using the Clustering algorithm, the Model calculates the probability values of the nodes of the bottom layer (*attributes*).

The canonical interaction NoisyOR is used to reduce the conditional probabilities (CPT tables) in the Model. The NoisyAdder method is used to calculate the probabilities of the nodes of the middle and bottom layers—*needs* and *attributes*.

The weight of an arc in the Model shows the significance of this relationship. The initial assumption underlying the Model is that all relations are equally significant and therefore of equal value. Fig. 2 presents a layout of the *Model for adaptation and personalization*.

The *Model for adaptation and personalization* was developed on the basis of the initial scheme, serving as a guideline for future system development in the Share.TEC project [2]. This initial scheme only identifies the guidelines and gives a "top view" but does not present a ready solution.

In the presented architecture, a total of 51 *stereotypes*, 40 *needs* and 20 *attributes* have been developed and implemented. The LEAK node participates in the implementation of the Model. Its use was introduced by Henrion [10] to reflect the phenomenon that an event could occur spontaneously in the absence of a concrete cause.

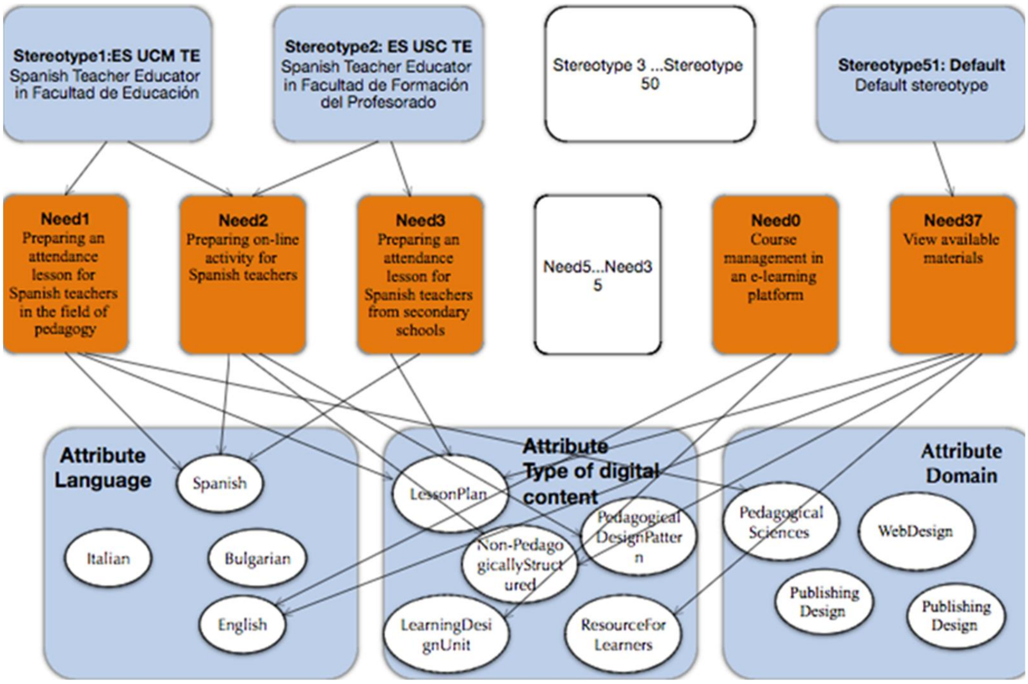

Fig. 2 Model for adaptation and personalization

Nodes in the Model

Top layer nodes (stereotypes). The binary variables of the top layer in the Model represent stereotypes in the system. In the presented work, 51 stereotypes (Stereotype 1–Stereotype 51) have been developed, including a default stereotype. After classification of a user to a stereotype/s, the variable/s of this/these stereotype/s in the Model get/s a true value, while the variables of all other stereotypes get a false value.

Medium Layer nodes (needs). Recommender systems base their functionality on the user's characteristics and needs. The needs in these systems are described as a constraint on the type of resources the user is likely to want to see. The binary variables of the medium layer in the Model represent the needs of the user in the system. Within the presented work, 40 needs (Need0–Need39) have been developed to describe the different needs of users in the system. The number of needs is smaller than the number of stereotypes, because in some cases, the same needs are used by several stereotypes. Each stereotype variable in the Model is related to one or more need variables. Each variable representing a need in the Model is related to a set of attributes that indicate what type of information content will satisfy that need.

Bottom layers nodes (attributes). The binary variables of the bottom layer in the Model represent the attributes in the system. The attributes and their possible values are taken from the TEO ontology and participate as elements in the CMM metadata model [21] of the Share.TEC digital library. This approach ensures that attributes derived from the Model can be used to identify resources whose metadata match these attributes. The attributes are a total of 20 and are grouped into three groups: “Language”, “Digital Content Type” and “Knowledge Area”. They are all elements in the TEO ontology.

- Attributes in the Language group are: English (EN), Spanish (ES), Italian (IT), and Bulgarian (BG).
- The attributes of the Digital Content Type group are: Learning Unit, Lesson Plan, Pedagogical Model of Structure, Non-Structured Pedagogical, and Learning Resources. Descriptions of the attributes taken from [22] are presented below:
  - LearningDesignUnit—a reusable unit that models the structure and flow of the learning process, including actors, resources, tools, activities and methods.
  - LessonPlan—a resource that describes the lesson through its goals, strategies, tools, resources, possible uses, and so on. It takes place in a given area of knowledge and targets a particular target group.
  - Pedagogical Design Pattern (PedagogicalDesignPattern)—uses strategies or techniques to make good practices portable. It is designed to provide solutions to typical educational problems.
  - Non-Pedagogically Structured—general-purpose auxiliary material that does not contain a specific pedagogical structure or direction. This category includes materials that are not

developed for pedagogical purposes, but can nevertheless be used in the learning process as an information resource.

- Resources for Learners (ResourceForLearners)—pedagogically structured content designed to be used for learning.

- The attributes of the Knowledge Area group are: PedagogicalSciences, TeacherTrainingPhisicalTraining, TeacherTrainingInMusic, Religion, ComputerScience, Educational Management EducationalManagement, CareerAdvising, WebDesign, Arts, LibraryInformationArchive, and PublishingDesign [22].

3.2.2. Module for presentation and management of knowledge for the user. This module is a component of the architecture of the Functional Module serving to represent a user as vector/s and classify it to a stereotype/s. The module performs the following main functions: (1) takes the user registration data; (2) uses that data to construct vector/s of the user; (3) compares that vector with the predefined vectors of stereotypes and (4) selects the most similar stereotype/s for a given user.

For the purpose of classifying a user into a stereotype a vector-space model is used. Both user and stereotypes are presented as 5-dimensional vectors with weights. Their similarity is calculated using the cosine similarity method, which reduces the task of classifying a user to a stereotype to a simple vector comparison. This approach allows each user to be classified to at least one stereotype. In cases where the similarity between the user and all stereotypes in the system is below a defined threshold, the user is classified to the default stereotype.

Vectors—structure and definitions. Similarity between two vectors

Let U_i be a user in the system, such that $U_i \in U$, where U is the set of all users in the system. Let S_i be a stereotype in the system, such that $S_i \in S$, where S is the set of all stereotypes in the system.

Let L_i be the language of a user U_i, such that $L_i \in L$, where L is the set of all languages in the system.

Let G_i be the target of a user U_i, such that $G_i \in G$, where G is the set of all goals.

Let R_i be the role of user U_i, such that $R_i \in R$, where R is the set of all roles.

Let I_i be the institution of user U_i, such that $I_i \in I$, where I is the set of all institutions in the system.

Let K_i be a domain of user U_i such that $K_i \in K$, where K is the set of all areas of knowledge in the system.

Let W_{tj} be the weight of element j in the user model t, such that $W_{tj} \in W$, where $W \in [0, 1]$.

Then each U_i user vector consists of the elements $(L_i, G_i, R_i, I_i, K_i)$ and can be represented by the vector:

$$\overrightarrow{u_t} = (W_{tl}, W_{tg}, W_{tr}, W_{ti}, W_{tk})$$

Let L_j be the language of the stereotype S_i, such that $L_j \in L$, where L is the set of all languages in the system.

Let G_j be the goal of a Si stereotype, such that $G_j \in G$, where G is the set of all goals.

Let R_j be the role of a S_i stereotype, such that $R_j \in R$, where R is the set of all roles.

Let I_j be the institution of a stereotype S_i, such that $I_j \in I$, where I is the set of all institutions in the system.

Let K_j be the area of knowledge of the S_i stereotype, such that $K_j \in K$, where K is the set of all areas of knowledge in the system.

Let W_{ij} be the weight of element j in stereotype i and $W_{ij} \in W$, where $W \in [0, 1]$.

Then each stereotype vector S_i consists of the elements $(L_j, G_j, R_j, I_j, K_j)$ and can be represented by the vector:

$$\overrightarrow{s_\iota} = (W_{il}, W_{ig}, W_{ir}, W_{ii}, W_{ik})$$

To determine the similar stereotypes for a given user, a calculation of the similarity between the vector/s of the user and the vectors of all stereotypes in the system is performed by the cosine similarity method.

Using the cosine similarity method, we can calculate the similarity between a user and the stereotypes by:

$$\mathrm{sim}(\overrightarrow{s_\iota}, \overrightarrow{u_t}) = \cos \boldsymbol{\vartheta} = \frac{\overrightarrow{s_\iota} . \overrightarrow{u_t}}{|\overrightarrow{s_\iota}||\overrightarrow{u_t}|} = \frac{\sum_j w_{ij} . w_{tj}}{\sqrt{\sum_j {w_{ij}}^2} . \sqrt{\sum_j {w_{tj}}^2}}$$

The equation has values:

- sim($\overrightarrow{s_\iota}$, $\overrightarrow{u_t}$) = 1, when $\overrightarrow{s_\iota} = \overrightarrow{u_t}$;
- sim($\overrightarrow{s_\iota}$, $\overrightarrow{u_t}$) = 0, when $\overrightarrow{s_\iota}$ and $\overrightarrow{u_t}$ have no common element.

The elements of user vectors and stereotypes are instances of the ontological classes *Language*, *ActorGoal*, *Role*, *(EducationalInstitution InformationCulturalAgency)*, and *KnowledgeAreaElement* of the TEO ontology. Because of this approach, the possible values of the elements are limited to the finite set which makes the task trivial. The structure of a vector and its relationship to the ontology TEO is presented in Fig. 3.

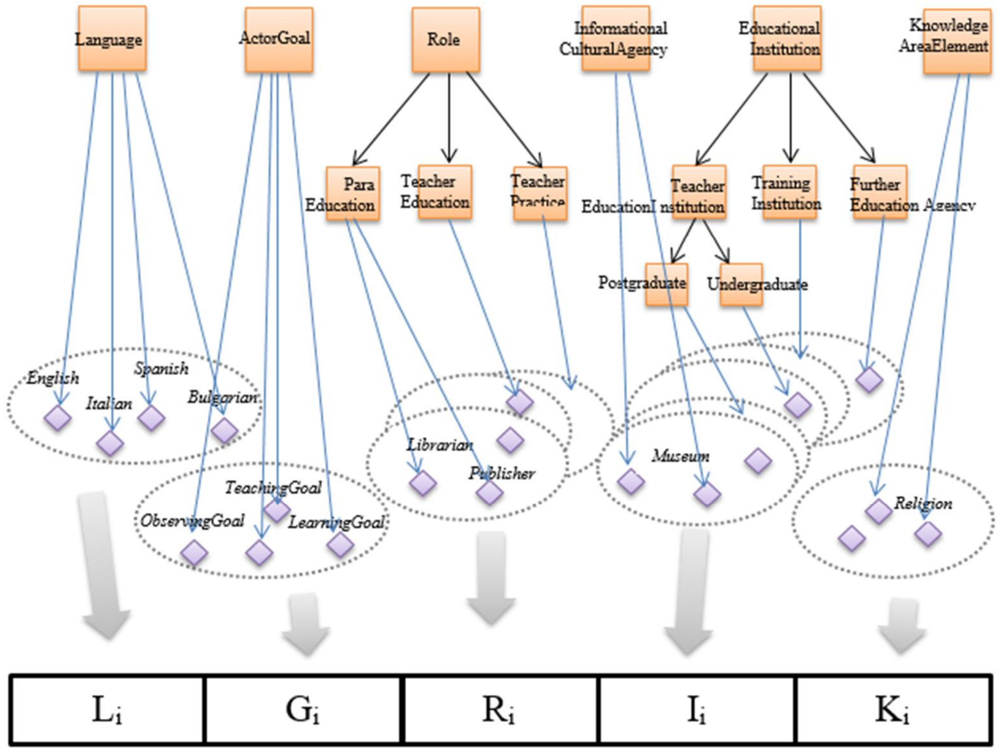

Fig. 3. Structure of a vector and its relationship with the ontology TEO

The first element of the vectors has a set of possible values limited to the instances of the ontological class *Language*, which are: *English-EU, Spanish, Bulgarian, Italian.*

The second element of the vectors has a set of possible values limited to the instances of the ontological class ActorGoal, which are: TeachingGoal, LearningGoal, ObservingGoal, LearnerVisualizationGoal.

The third element of the vectors has a set of possible values limited to the instances of the ontological class Role and his sub-classes ParaEducation, TeacherEducation and TeacherPractice. These instances have values: ContentDeveloper, Librarian, Publisher, ServiceProvider, StudentTeacher, TeacherEducator, TeacherTrainer, Coordinator, HeadOfSchool, Mentor-InductionMentor, Teacher, TeachingAssistant, Technician, Trainer and Tutor.

The fourth element of the vectors has a set of possible values limited to the instances of the ontological classes EducationalInstitution and InformationalCulturalAgency and their respective sub-classes. Some of the values of the instances are: FurtherEducationAgency—FurtherEducationAgencyES_4, FurtherEducationAgencyES_5, ..., TrainingInstitutionIT_4, GovernmentAgency, InformationCulturalAgencyES_1, ..., ResearchCenter.

The fifth element of the vectors has a set of possible values limited to the instances of the ontological class KnowledgeAreaElement, some of which are: PedagogicalSciences, TeacherTraining-PhysicalTraining, TeacherTrainingInMusic, ..., Religion.

Stereotype vector

The elements in stereotype vectors have a predefined weight equal to one for each element of the vector. Part of the stereotypes in the system have an empty set for one or more elements. This is in cases where this element/s is/are not essential for a given stereotype. In those cases the element of the stereotype vector takes the value of the respective element of the user vector, as its weight gets reduced by 2. Fig. 4 shows a stereotype vector of this type before the user vector comparison process.

Вектор на стереотип

EN 1	TeachingGoal 1	TeacherEducator 1	PostgraduateIE_2 1	∅

Fig. 4. A stereotype vector with an empty set element

The process of setting the value of an empty set element when compared to a user vector is shown in Fig. 5.

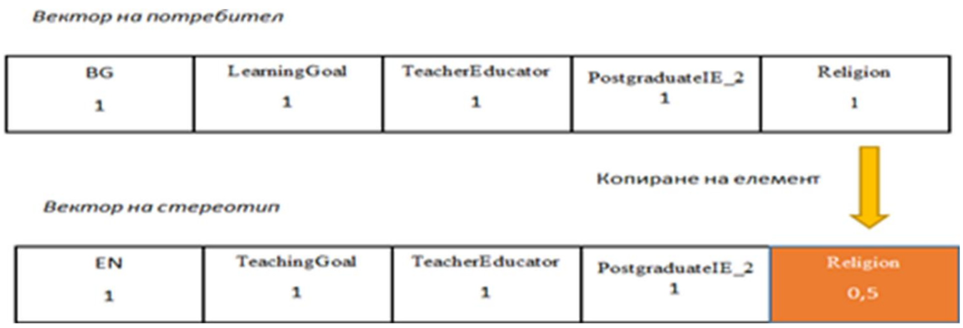

Fig. 5. Setting the value of the empty set element when compared to a user vector

The vectors of stereotypes are created in advance and stored in a stereotype store in the system. Examples of stereotype vectors are shown in Table 1.

User vector

For weighing the elements of a user vector is chosen approach in which all elements get the highest possible weight, equal to one. This is because all values of elements in a user vector are set directly by the user himself and the system has confidence in the user's choice.

In the future development of the system, it is possible to indirectly collect a part of the user vector data automatically by the system. In this case, the weights of the indirectly collected values of the elements should be less than the ones directly provided by users. In a user vector in case of absence of data for particular element that element gets a weight of 0.

3.2.3. Module for applying adaptive search. This module is a component of the Architecture of a Functional Module for Adaptive Search in a Digital Library. The module accepts as input the original user query submitted by the user on one hand and terms for expansion of the query from the *Module for implementation of adaptive and personalization logic* on the other hand. It modifies the query and retrieves results with the modified query.

Table 1. Stereotype Vectors in the system

Stereotype	**Description of stereotype**	**Stereotype vector**
Stereotype1: ES UCM TE	Spanish Teacher Educator at Facultad de Educación (Máster y Doctorado)	ES(1), TeachingGoal (1), TeacherEducator (1), PostgraduateES_1(1), 0
Stereotype2: ES USC TE	Spanish Teacher Educator at Facultad de Formación del Profesorado	ES(1), TeachingGoal(1), TeacherEducator(1), PostgraduateES_2(1), 0
Stereotype3: IR SST TE	Irish Teacher Educator of postgraduate teachers	EN(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIE_1(1), 0
Stereotype4: IR OTT TE	Irish Teacher Educator for on-line education	EN(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIE_2(1), 0
Stereotype5: IT SSIS TE	Italian TeacherEducator in Scuola di Specializzazione per l'Insegnamento Secondario	IT(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIT_1(1), PedagogicalScience(1)

Essential for the success of the implemented adaptive search is the existing metadata model which annotates information resources in the system. In the digital library Share.TEC resources are annotated with metadata from LOM-based Common Metadata Model (CMM). The Metadata Model CMM includes an expanded set of metadata specifically designed to describe the pedagogical resources [21]. CMM is closely related to the ontology TEO, which serves as a conceptual basis for describing the elements and objects in the domain of teacher education. The search engine in the system is set to use these metadata fields in the search.

The attributes of the Model used for query expansion are instances of ontology classes from TEO ontology. The same instances are used in metadata elements of the CMM model.

In the Share.TEC portal there is functionality for so-called Advanced search. It is based on manual input from the user. The presented architecture takes advantage of this functionality but upgrades it to a higher level. In our approach the system populates the field of Advanced search template automatically. The fields are populated with attributes taken from the *Model for adaptive search and personalization.*

3.3. Implementation of the Model for adaptive search and personalization. The implementation of *Model for adaptive search and personalization* is based on the development environment GeNIe and Library SMILE. They are provided by Decision Systems Laboratory at Pittsburgh University and were made available to the general public in 1998. GeNIe and SMILE are distributed as open source software and have no limitations on their use. To date, GeNIe and SMILE licenses have been purchased by BayesFusion LLC (in 2015), which continues the tradition of the University of Pittsburgh and provide the software free of charge for academic purposes [4].

Fig. 6 shows an implementation of the Model for adaptive search and personalization in the graphical development environment GeNIe 2.0 [4].

4. Future development of the system. This paper presents a defended PhD thesis.

The main scientific contributions of the thesis are:

- Development of an Architecture of a Functional Module for Adaptive Search in a Digital Library and its main components.
- Development of a Basic scheme of adaptive and personalization logic for adaptive search. The implementation of the adaptive logic is based on an inference in the Bayesian Network model.
- Use of the domain ontology TEO for user and stereotype presentations. The same ontology was used to define a part of the metadata that annotates resources in the Share.TEC digital library.

The main applied contributions are:

- Development of a list of Stereotypes (51 in all, including the default stereotype), list of the needs of the users and list of attributes for query expansion.

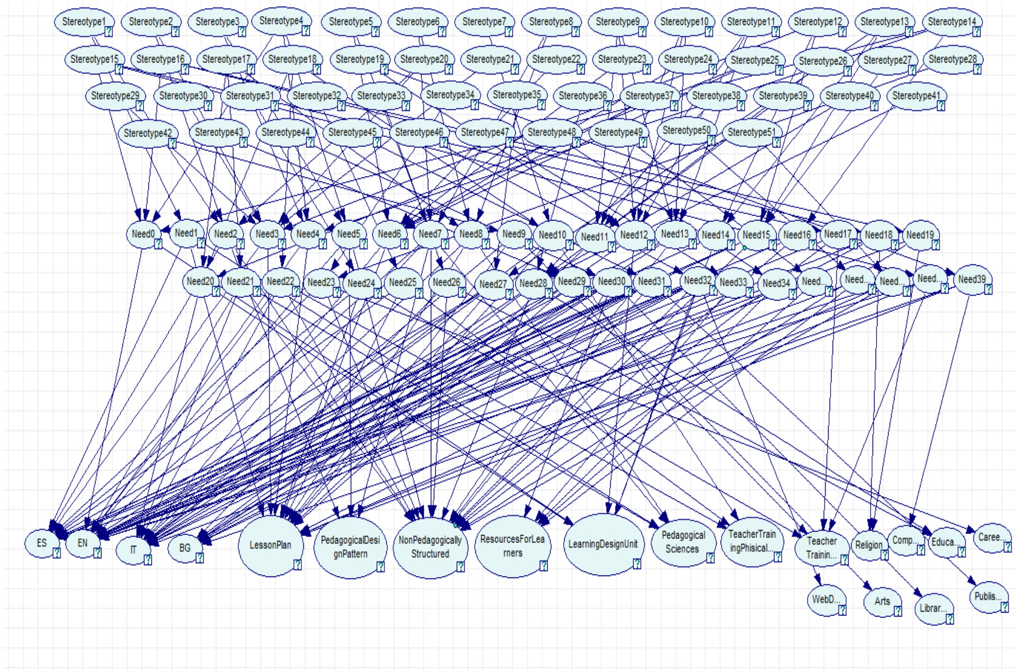

Fig. 6. Model for adaptive search and personalization in graphical development environment GeNIe 2.0 [4]

- Inclusion of lists of Stereotypes, needs, and attributes into developed Bayesian model.
- Development of the Bayesian model, with its three-layer structure.
- Proposal of a new approach for user classification, based on a vector-space model and representation of both users and stereotypes as vectors.
- Proposal of a new query expansion method, based on use of the domain ontology TEO and currently implemented metadata model in the library.

In the future, it is possible to extend the Bayesian model by adding new nodes to each layer, as well as creating new relations. Modifications to the parameters in Bayesian model like as setting weights of arcs, defining different probability values, and so on.

Another direction for future development is to modify the weights of the vector elements so that they reflect the relations in the TEO ontology.

REFERENCES

[1] BAEZA-YATES R., B. RIBEIRO-NETO. Modern Information Retrieval. Addison-Wesley Publishing Co., 1999.

[2] BOYTCHEV P., S. BOCCONI, K. STEFANOV. An Ontological Approach for Multicultural Digital Repositories. 2010. `http://hdl.handle.net/10867/77`, 30 October 2018.

[3] BRUSILOVSKY P. Adaptive Hypermedia. *User Modeling and User-Adapted Interaction. The Journal of Personalization Research*, **11** (2001), No 1–2, 87–110.

[4] GeNIe and SMILE. `http://www.bayesfusion.com`, 30 October 2018.

[5] DIEZ F. J. Parameter adjustment in Bayes networks. The generalized NoisyOR–gate. In: Uncertainty in Artificial Intelligence. Proceedings of the 9th Conference, Washington D. C., USA, 1993, 99–105.

[6] IEEE P1484.2.22/D8, PAPI Learner—Learner Relations Information. `http://metadata-standards.org/Document-library/Meeting-reports/SC32WG2/2002-05-Seoul/WG2-SEL-046_SC36N0182_papi_learner_relations_info.pdf`, 30 October 2018.

[7] IMS Global Learning Consortium. `http://www.imsproject.org/aboutims.html`, 30 October 2018.

[8] HECKERMAN D. Causal Independence for Knowledge Acquisition and Inference. In: UAI'93 Proceedings of the Ninth International Conference on Uncertainty in Artificial Intelligence, Washihgton, DC, 1993, 122–127.

[9] HENRION M. Practical issues in constructing a Bayes' belief network. In: UAI'87 Proceedings of the Third Conference on Uncertainty in Artificial Intelligence, Seattle, WA, 1987, 132–139.

[10] HENRION M. Some practical issues in constructing belief networks. In: L. Kanal, T. Levitt, and J. Lemmer (eds). *Uncertainty in Artificial Intelligence*, **3** (1989), 161–173.

[11] Kobsa A., J. Koenemann, W. Pohl. Personalised hypermedia presentation techniques for improving online customer relationships. *The Knowledge Engineering Review*, **16** (2011), No 2, 111–155.

[12] Micarelli A., F. Gasparetti, F. Sciarrone, S. Gauch. Personalized Search on the World Wide Web. In: P. Brusilovsky, A. Kobsa, W. Nejdl (eds). The Adaptive Web: Methods and Strategies of Web Personalization. Springler, Germany, 2007, 195–230.

[13] Oard D. W., A. R. Diekema. Cross-Language Information Retrieval. *Annual Review of Information Science (ARIST)*, **33** (1998), 472–483.

[14] Pearl J. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgen Kaufmann Publishers, San Mateo, CA, 1998.

[15] Rich E. User modeling via stereotypes. *Cognitive Science*, **3** (1979), No 4, 329–354.

[16] Rich E. Users are individuals: individualizing user models. *International Journal of Man-Machine Studies*, **18** (1983), 199–214.

[17] Share.TEC. Deliverable 5.5 Development and pilot deployment of the integrated system. `https://www.cnr.it/en/focus/102-27/results-of-the-share-tec-project`, 30 October 2018.

[18] Smeaton A., J. Callan. Personalisation and Recommender System in Digital Libraries. *International Journal on Digital Libraries*, **5** (2005), No 4, 299–308.

[19] Spertus E., M. Sahami, O. Buyukkokten. Evaluating Similarity Measures: A Large-Scale Study in the Orkut Social Network. In: KDD'2005 Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, Chicago, Illinois, USA, 2005, 678–684.

[20] Stefanov K., R. Nikolov, P. Boytchev, E. Stefanova, A. Georgiev, I. Koychev, N. Nikolova, A. Grigorov. Emerging

Models and e-Infrastructures for Teacher Education. In: 2011 International Conference on Information Technology Based Higher Education and Training, Kusadasi, Izmir, Turkey, 2011, 1–8.

[21] Stefanov K., P. Boytchev, E. Stefanova, A. Georgiev, N. Nikolova, A. Grigorov. Digital Libraries in Teacher Education. In: Proceedings of the 40th Jubilee Spring Conference, 2011, 120–135. https://research.uni-sofia.bg/handle/10506/762, 30 October 2018.

[22] Teacher Education Ontology (TEO): version 1. http://www.share-tec.eu/content/1/c6/04/41/02/D2_1_TEO_v1.pdf.

[23] Dimitrov D., D. Nikovski. Artificial Intelligence. Technical University, Sofia, 1999. (in Bulgarian)

[24] Pavlov R., D. Paneva, L. Pavlova-Draganova, L. Draganov. Digital libraries with multimedia content and applications in Bulgarian cultural heritage, 2006, 1–100. http://mdl.cc.bas.bg/Digital_libraries_with_multimedia_content_and_applications_in_Bulgarian_cultural_heritage.pdf, 30 October 2018. (in Bulgarian)

Albena Turnina
Faculty of Mathematics and Informatics
St. Kliment Ohridski University of Sofia
5, James Bourchier Blvd
1164 Sofia, Bulgaria
e-mail: albena.turnina@fmi.uni-sofia.bg
albenatur@yahoo.com

Received November 14, 2017
Final Accepted March 15, 2018