СОФИЙСКИ УНИВЕРСИТЕТ „СВ. КЛИМЕНТ ОХРИДСКИ”
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

Семантично-ориентирани архитектури и използване на онтология за организиране на адаптивно търсене

Semantic-oriented architectures and using ontology for organizing adaptive search

ДИСЕРТАЦИОНЕН ТРУД

за присъждане на образователна и научна степен "Доктор"
в професионално направление 4.6 Информатика и компютърни науки
докторска програма "Информационни технологии“

Докторант: Албена Петрова Търнина
Научен ръководител: Проф. д-р Красен Стефанов

София, 2016

СЪДЪРЖАНИЕ

СПИСЪК НА ТАБЛИЦИТЕ.....5

СПИСЪК НА ФИГУРИТЕ.....7

СПИСЪК НА СЪКРАЩЕНИЯТА.....9

УВОД.....10

Глава 1. ОБЩА ПОСТАНОВКА НА ЗАДАЧАТА.....16

1.1 Обект и предмет на дисертационния труд.....19

1.2 Цели на дисертационния труд.....19

1.3 Задачи на дисертационния труд.....20

1.4 Описание на съдържанието на главите.....23

Глава 2. АДАПТИВНОСТ И ПЕРСОНАЛИЗАЦИЯ. МЕТОДИ, ПОДХОДИ И СРЕДСТВА ЗА РЕАЛИЗИРАНЕ НА АДАПТИВНО ТЪРСЕНЕ.....26

2.1 Технологии на Семантичния уеб – езици, протоколи и стандарти.....26

2.2 Адаптивност и персонализация – понятия и дефиниции. Адаптивни системи.....29

2.3 Модел на потребител – видове и характеристики.....33

2.3.1 Характеристики на модела на потребител.....34

2.3.2 Моделиране на потребител – методи и подходи.....37

2.3.3 Архитектура на модел на потребител в САПТ.....38

2.3.3.1 Модел - вектор от термини.....38

2.3.3.2 Модел на потребител, базиран на семантична мрежа.....42

2.3.3.3 Онтологичен модел на потребител.....43

2.4 Използване на стереотипи за реализиране на адаптивност и персонализация.....45

2.5 Адаптивно търсене – дефиниции и характеристики……..……...54

Глава 3. БЕЙСОВИ (ВЕРОЯТНОСТНИ) МРЕЖИ. МЕТОДИ, ПОДХОДИ И СТРАТЕГИИ ЗА ИЗГРАЖДАНЕ НА БЕЙСОВИ МОДЕЛИ.....62

3.1 Модели за представяне на причинна независимост……..…..…..68

3.1.1 Взаимодействието NoisyOR.....69

3.1.2 Взаимодействието NoisyAdder.....71

3.2 Бейсови модели – структура и параметризация. Сравнителен анализ на модели с канонично взаимодействие.....72

3.3 Използване на Бейсови модели в САПТ.....76

Глава 4. АРХИТЕКТУРА НА ФУНКЦИОНАЛЕН МОДУЛ ЗА АДАПТИВНО ТЪРСЕНЕ В ДИГИТАЛНА БИБЛИОТЕКА.....80

4.1 Базова схема на адаптивна и персонализираща логика за реализиране на адаптивно търсене.....80

4.2 Компоненти в архитектурата на Функционален модул за адаптивно търсене в дигитална библиотека.....82

4.2.1 Модул за изпълнение на адаптивна и персонализираща логика.....84

4.2.2 Модул за представяне и управление на знанието за потребител.....93

4.2.3 Модул за прилагане на адаптивност при търсене.....104

Глава 5. РЕАЛИЗАЦИЯ НА МОДУЛ ЗА ИЗПЪЛНЕНИЕ НА АДАПТИВНА И ПЕРСОНАЛИЗИРАЩА ЛОГИКА ПОСРЕДСТВОМ ИЗВОД В БЕЙСОВА МРЕЖА…….....111

5.1 Системата Share.TEC.....112

5.2 Реализация на Модул за изпълнение на адаптивна и персонализираща логика...…..114

Глава 6. ИЗСЛЕДВАНИЯ И ЕКСПЕРИМЕНТИ.....133

ЗАКЛЮЧЕНИЕ.....154

БИБЛИОГРАФИЯ.....159

АВТОРСКА СПРАВКА – ПУБЛИКАЦИИ И УЧАСТИЯ В НАУЧНИ ФОРУМИ.....169

ПРИЛОЖЕНИЯ.....171

ПРИЛОЖЕНИЕ 1.....171

ПРИЛОЖЕНИЕ 2.....219

ПРИЛОЖЕНИЕ 3.....224

ПРИЛОЖЕНИЕ 4.....231

ПРИЛОЖЕНИЕ 5.....245

СПИСЪК НА ТАБЛИЦИТЕ

Таблица №	Описание	Страница
Таблица 1	Сравнение на модели – вектори от термини, адаптирано от [37] и допълнено	39
Таблица 2	Сравнение на системи, използващи стереотипи, въз основа на техни признаци и характеристики	53
Таблица 3	Сравнение между канонични Бейсови модели	75
Таблица 4	Инстанции на класа *Language*, съставящи множеството от стойности на 1-я елемент на вектор	97
Таблица 5	Инстанции на класа *ActorGoal*, съставящи множеството от стойности на 2-я елемент на вектор	97
Таблица 6	Инстанции на класа *Role,* съставящи множеството от стойности на 3-я елемент на вектор	98
Таблица 7	Инстанции на класа *EducationalInstitution*, попълващи 4-я елемент на вектор	98
Таблица 8	Инстанции на класа *InformationalCulturalAgency*, попълващи 4-я елемент на вектор	99
Таблица 9	Инстанции на класа *KnowledgeAreaElement*, съставящи множеството от стойности на 5-я елемент на вектор	100
Таблица 10	Вектори на стереотипи в системата	103
Таблица 11	Елементите *Language, DigitalContentType* и *KnowledgeAreaElement* в модела за метаданни CMM	109
Таблица 12	Релации между променливите в Модела	121
Таблица 13	Основни методи на класа DSL network	126
Таблица 14	Основни методи на класа DLS sysCoordinates	128
Таблица 15	Основни методи на класа DSL node	129
Таблица 16	Основни методи на класа DSL nodeValue	129
Таблица 17	Основни методи на класа DSL Dmatrix	129
Таблица 18	Представяне на вектор на потребител и стереотипи в модела вектор-пространство	141
Таблица 19	Вероятности на избрани възли от Модела, при задаване на априорни вероятности със стойности 0,2 за *истина* и 0,8 за *лъжа* на възлите от най-горния слой	143

Таблица 20	Вероятности на избрани възли от Модела, при задаване на доказателство *истина* на възел Stereotype1 и *лъжа* на всички останали възли от най-горния слой при априорни вероятности със стойности 0,2 за *истина* и 0,8 за *лъжа*	144
Таблица 21	Вероятности на избрани възли от Модела, при задаване на априорни вероятности със стойности 0,5 за *истина* и 0,5 за *лъжа* на възлите от най-горния слой	145
Таблица 22	Вероятности на избрани възли от Модела, при задаване на доказателство *истина* на възел Stereotype1 и *лъжа* на всички останали възли от най-горния слой при априорни вероятности със стойности 0,5 за *истина* и 0,5 за *лъжа*	145

СПИСЪК НА ФИГУРИТЕ

Фигура №	Описание	Страница
Фиг. 1	Модел на Семантичния уеб, по модела на [31]	27
Фиг. 2	Подходи за адаптивно и персонализирано търсене според мястото на модела на потребител [61]	56
Фиг. 3	Вероятностна мрежа	64
Фиг. 4	Структура на модела QMR-DT [87]	73
Фиг. 5	Структура на модела BN3M [51]	74
Фиг. 6	Структура на модела на Radde et al [76]	75
Фиг. 7	Базова схема на адаптивна и персонализираща логика за реализиране на адаптивно търсене	81
Фиг. 8	Архитектура и компоненти на Функционален модул за адаптивно търсене в дигитална библиотека	83
Фиг. 9	Схема на Бейсов модел за адаптиране и персонализиране	85
Фиг. 10	Първоначална схема за възможна употреба на стереотипи в дигиталната библиотека, според [8]	86
Фиг. 11	Реализацията на Модела в графичната среда за разработка GeNIe 2.0	87
Фиг. 12	Схема на възел от мрежата	92
Фиг. 13	Структура на вектор и връзката му с онтологията ТЕО	97
Фиг. 14	Вектор на стереотип с елемент празно множество	101
Фиг. 15	Процес на задаване на стойност на елемента празно множество при сравнение с вектор на потребител	102
Фиг. 16	Процес на адаптивно търсене в системата	108
Фиг. 17	Схема на алгоритъм за съставяне на Бейсова мрежа	121
Фиг. 18	Схема на алгоритъм за получаване на данни за атрибути за адаптиране на заявката в дигитална библиотека според избран стереотип на потребител	126
Фиг. 19	Формуляр за регистрация на потребител	140
Фиг. 20	Конструиране на вектор на потребителя X от зададени от него данни	140
Фиг. 21	Вероятности на състоянията на възел Stereotype1 след зададено доказателство *истина*	144
Фиг. 22	Резултат от задаване на доказателство *лъжа* на всички възли от най-	147

	горния слой	
Фиг. 23	Моделът при зададени доказателства *истина* на Stereotype16 и доказателства *лъжа* за всички останали стереотипи	148
Фиг. 24	Моделът след извършване на извод за потребителя X, класифициран към Stereotype19 „Български Инструктор от CISCO мрежова академия“	150
Фиг. 25	Адаптиране на заявката с формата за разширено търсене в Share.TEC	151

СПИСЪК НА СЪКРАЩЕНИЯТА

ДБ	*Дигитална библиотека*
БМ	*Бейсова мрежа*
БС	*Бейсово съждение*
СВР	*Съвместно Вероятностно Разпределение*
АПС	*Адаптивни и персонализирани системи*
САПТ	*Системи за адаптивно и персонализирано търсене*
IEEE PAPI или PAPI	*IEEE Public and Private Information Specification*
IMS LIP	*IMS Learning Information Package Specification*
OWL	*Web Ontology Language*
RDF	*Resource Description Framework*
RDFS	*RDF Schema*
W3C	*World Wide Web Consortium*
WWW	*World Wide Web*
LOM	*Learning Object Metadata*
CMM	*Common Metadata Model*
TEO	*Teacher Education Ontology*

УВОД

През последните години Семантичният уеб се появи и наложи като рамка, обхващаща набор от технологии и стандарти, задаващи общ формат на данните и на протоколите за обмяната на тези данни в уеб пространството. Семантичният уеб представлява разширение на стандартния уеб със стандарти, позволяващи данните да бъдат споделяни и използвани многократно между приложения, организации и общности. Технологиите на Семантичния уеб стоят в основата на редица новаторски системи, в които данните се представят във форма, която позволява извършване на извод над тях и автоматично извеждане на нови факти. Тези технологии намират все по-широко приложение в системите за доставяне на адаптивно и персонализирано съдържание и услуги в Интернет. Очаква се в близкото бъдеще значителна част от взаимодействията между приложенията да бъдат извършвани автоматично посредством интелигентни софтуерни агенти. Резултат от навлизането на тези технологии е появата на ново поколение системи, осъществяващи по-лесно и ефективно предаване, търсене, смесване и обработване на данни. Сред предимствата на семантично-базираните архитектури за адаптивно търсене е възможността за използване на унифициран речник за описване на обектите в предметната област, базиран на онтология. Сред обектите, описвани от онтологичния речник, могат да бъдат както метаданните, с които са анотирани информационните ресурси в системата, така и данните за самия потребител. Тогава такъв тип системи ще могат да осигурят точно, надеждно и ефективно търсене, което винаги да осигурява най-подходящите колекции от резултати за потребител посредством съпоставяне на метаданните на ресурсите с данните от модела на потребителя.

В последните години се наблюдава нарастване на информацията, достъпна по мрежата в уеб пространството, и увеличаване на броя потребители на тази информация. В една отворена и огромна по своя обем среда за информация като Интернет, потребителят може да се окаже загубен, объркан, демотивиран и неспособен да открие нужната му информация. Поради това нараства нуждата от специализирани среди, доставящи знание на потребителя, в които информационните ресурси да бъдат категоризирани, описани и подредени, а техните метаданни - класифицирани в общо хранилище [95]. Такива среди са дигиталните мултимедийни библиотеки, които могат да съдържат всякакъв вид „физически материали“, които са били дигитализирани –

книги, аудиовизуални или мултимедийни материали, фотографии, архивни документи и други, както и материали, оригинално създадени в цифров формат. Дигиталните библиотеки са онлайн среди за доставяне на знания, които представляват колекции от дигитални обекти, които се събират, класифицират и управляват в съответствие с утвърдени принципи за развитие на колекциите. Дигиталните библиотеки се определят още като управляеми колекции от дигитална информация, които са достъпни за потребител посредством мрежа [97, 105]. В тях се съхранява разнородна информация (графични изображения, звук, видео и др.) и те осигуряват редица услуги, които позволяват на потребител да извлича и използва тази информация. В рамките на една дигитална библиотека се осъществява дейност по управление на информационни ресурси, индексиране и анотиране, сортиране, класифициране, добавяне на метаданни към обектите и извличане на информация (търсене). Дигиталната библиотека служи като среда, в която се обединяват колекции, услуги и хора, за да се поддържа пълният цикъл от създаване, разпространение, използване и съхранение на данни, информация и знания [99].

С нарастване на количеството на информационните ресурси в дигиталните библиотеки потребителят отново е изправен пред проблема за откриване на тези колекции, които от една страна отговарят на неговите нужди и от друга страна са съобразени с неговата индивидуалност и предпочитания. Това води до нарастване на необходимостта от наличието на способи и методи за реализиране на адаптивно търсене в дигитална библиотека. Адаптивното търсене е свързано с изграждане на системи, способни да осигуряват индивидуализирана колекция от информационни ресурси за отделен потребител. В основата на системите за адаптивно търсене стои модел, описващ предпочитанията и характеристиките на потребителя. Адаптивното (персонализирано) търсене е процес на селекция на най-подходящите резултати и тяхното подреждане в зависимост от конкретен потребител, група от потребители или всички потребители на системата. Адаптивното търсене се реализира посредством набор от подходи и методи, като: (1) адаптиране на заявката; (2) преподреждане на резултатите и (3) оценяване на резултатите. Съществуват различни типове системи, в които може да се осъществи адаптивно (персонализирано) търсене. Те се различават според областта на знанието, с която са свързани, вида на услугата, която предоставят и езика/езиците на които функционират. Такива системи са средите доставящи знания,

в които влизат средите за електронно обучение и дигиталните библиотеки, системите за новини, търсещите машини (търсачки) и др.

Персонализацията в една система може да бъде насочена както към отделен потребител, така и към група от потребители или към всички потребители на системата. Според това персонализираните системи могат да бъдат групирани в три категории: системи, доставящи индивидуализирана персонализация, системи, в които персонализацията е насочена към общности или групи с общи интереси и системи, доставящи агрегатна персонализация [38]. В първия случай персонализацията се реализира на базата на индивидуален (персонален) модел, във втория - с помощта на групов модел (стереотип), и в третия - на основата на колективните данни, събрани от всички потребители при използване на системата.

Потребителският модел се явява значима и съществена част на системите, доставящи адаптивно и персонализирано съдържание или услуга [9, 37]. Моделирането на знанието за потребител е централна изследователска задача при изграждането на Адаптивни и персонализирани системи (АПС) [10]. Съществуват различни подходи за събиране на данни за моделиране на потребител – системите за уеб персонализация използват проследяващи алгоритми, за да следват потребителя в неговото сърфиране из мрежата, докато адаптивните хипермедийни системи създават потребителски модел на базата на взаимодействието на потребителя със системата. Моделирането на потребителя обхваща представянето на различни негови характеристики и атрибути като: демографски характеристики, ниво на експертност в дадена област, когнитивни особености и др. Класификацията на потребителските характеристики, представена от Brusilovsky [10], обхваща знанието на потребителя, неговите цел и квалификация, опит, интереси и среда.

Съществува голямо разнообразие и вариативност в моделите за описване на знанието за потребител, както в тяхната структура и представяне, така и по отношение на различни техни характеристики и свойства. Те позволяват моделите да бъдат категоризирани като: персонални или групови, експлицитни или имплицитни, динамични или статични, дългосрочни или краткосрочни и др. От съществено значение при създаването на даден модел е да се определи как ще се представя този модел, какви техники за конструиране ще използва, дали и какъв извод ще се извършва над него и др.

Потребителят в системата може да се представи чрез персонален модел или може да бъде включен в групов модел, представящ характеристики на дадена общност от потребители. Персоналният модел на потребител представя неговите конкретни характеристики и предпочитания, без да отчита неговото сходство с други потребители в системата. Поради това не винаги е възможно и лесно събиране на данни за изграждане на този тип модел. Освен това се оказва, че в ред системи такова детайлно представяне на потребител не е нужно, тъй като едни и същи стратегии за адаптивност могат да се приложат върху група от потребители. В тези случай е по-удачно използването на групов (стереотипен) модел, който да представя знанието за цяла група от потребители в системата. Стереотипното моделиране намира най-широко приложение в експертните системи и в обучителните системи [78].

Стереотипът се смята за агрегиран и балансиран модел на група от потребители. Това означава, че ако два и повече потребителя споделят един и същи стереотип, то те принадлежат към обща група, имат сходно поведение в системата и подобни интереси. Стереотипите представят група от характеристики, за които се приема, че са верни за всички членове на групата. Стереотипите предоставят частично описание на група от потребители и поради това, ако дадена характеристика за конкретен потребител липсва, тя може да бъде изведена от стереотипа към който потребителят принадлежи. Стереотипите в адаптивните системи в най-общия случай служат за групиране на потребителите и за прилагане на общи правила за адаптивност върху тях. Класифицирането на потребител към стереотип не винаги е лесна задача. Това е така, защото всеки потребител се представя в системата чрез широк набор от изрични и косвени характеристики, мнозинството от които не са числови. Едно от възможните решения за класифициране на потребител към даден стереотип е клъстеризиране на потребителите на базата на всичките им характеристики. Този подход е свързан с наличието на трудности за правилното клъстеризиране на не-числови множества от данни, като за това са нужни и голямо количество ресурси [7, 93, 94]. Поради това съществуват подходи, които използват само част от характеристиките на потребителя за неговото класифициране към подходящ стереотип.

Информацията за потребителя в системата може да бъде несигурна или неточна, което налага търсенето на методи и способи за справяне с тази несигурност. Основни технологии, които се използват за извод при несигурност, са Бейсови мрежи

(БМ) и Размита логика. Мрежите на Бейс са вероятностни графични модели, в който всеки възел представлява случайна променлива и всяка релация изразява причинно-следствено влияние. Те са смятани за една от най-добрите техники за решаване на диагностични и класификационни проблеми [12]. Използването на модела на БМ позволява да се разсъждава както в диагностична, така и в прогностична посока. Тази способност на БМ за извод в двете посоки е много полезна при моделиране на знанието за потребител.

Изводите над знанието за потребител чрез Бейсови модели е най-разпространено в адаптивните обучителни системи. Това е така, защото Бейсовите модели дават възможност за точно диагностициране на актуалното познавателно ниво на потребител и позволяват да се направи информирана препоръка за най-подходящата следващата дейност (например упражнения, тестове), според наличните доказателства за потребителя (предишни дейности със системата, любима стратегия за обучение и т.н.).

Съществуват различни подходи за изграждане на Бейсов модел, като някои от тях се опират изцяло на знанията на експерти в областта, докато други използват автоматизирани методи за събиране на данни или комбинират двата подхода.

Съществуват три главни групи от подходи за адаптивно търсене в зависимост от мястото на модела на потребителя: (1) подходи при които моделът на потребителя участва в процеса на извличане на информация, т.нар. оценяване на резултати; (2) подходи, при които моделът участва във вторична обработка на извлечените резултати, водеща до преподреждане на резултатите от търсенето и (3) подходи, при които моделът се използва за адаптиране на заявката за търсене [61]. Адаптирането на текущата заявка води до създаването на нова заявка, за която се приема, че представя по-точно нуждите и интересите на потребителя. Главните методи за адаптиране на заявката са: модифициране на заявката, разширяване на заявката с нови термини и задаване на тежести на термините от заявката.

Адаптирането на заявката решава редица проблеми, свързани с недостатъчно познаване на предметната област от потребител. Простите заявки по ключови думи (свободно търсене) са полезни тогава, когато потребителят има ясна представа какво търси и знае какви ключови думи да използват при търсенето. В средите, доставящи знания, често нивото на познаване на предметната област е ниско и

се появява необходимост от трансформиране на оригиналната заявка на потребителя в заявка, която отчита и използва установената семантика в областта. Към момента най-популярните методи за търсене са базирани на въведените от потребителя ключови думи. Тяхна слабост са липсата на точност и прецизност, невъзможността за контекстуализиране на заявката, както и неспособността за изразяване на синонимия. При свободното търсене, е невъзможно да се обхванат по-сложните отношения между обектите на по-дълбоко семантично ниво. Слабостите на този вид търсене, могат да бъдат избегнати посредством употребата на онтология, която задава семантичните връзки между обектите на по-дълбоко ниво.

Популярен метод, който позволява на потребител да търси и навигира в комплексни информационни пространства, е фасетното търсене. Системите за фасетно търсене помагат на потребител да открие нужната информация, като дават възможност за търсене не само чрез ключови думи, но и посредством метаданни, с които са анотирани информационните обекти. Този метод е особено ефективен, когато метаданните са свързани с предметна онтология. Подобен метод е реализиран в системата Share.TEC чрез функционалността за разширено търсене [7, 94]. Недостатък на тази функционалност е допълнителнителното усилие на потребител на системата да зададе набор от атрибути за търсене. В този случай процесът на търсене може да бъде усложнен и от неспособността на потребителя да подбере най-точните и подходящи ключови думи за търсене или да зададе тези атрибути на метаданните, които да го отведат до желаните резултати.

Тези проблеми могат да намерят своето решение чрез изграждане на такава функционалност за адаптивно търсене, която автоматично и прозрачно за потребител да го подпомага в неговото търсене.

Искам да изкажа специалните си благодарности на членовете на катедра „Информационни технологии“, чиито съвети, насоки и напътствия бяха безценни за мен.

Изключително дълбока благодарност и признателност дължа на научния си ръководител Проф. д-р Красен Стефанов за неговия всеотдаен професионализъм, строг научен подход и поддръжка по време на цялата ми докторантура и при написването на тази дисертация.

Глава 1. ОБЩА ПОСТАНОВКА НА ЗАДАЧАТА

В една информационна система интерфейсът за търсене осигурява начална точка за потребител, откъдето да се стартира процеса на търсене в системата. Празното поле за търсене не задава ключове, които да помогнат на потребителя в неговото търсене. Обикновено потребителят не започва търсенето със създаване на дълга и детайлна заявка. Изследванията показват, че типичният потребител стартира търсенето с много къси заявки, инспектира резултатите и след това модифицира тези заявки в постепенен цикъл на прецизиране [4]. В този процес първоначалната заявка служи за „опипване на почвата“ или за тестване на това, какви резултати ще върне системата. При липса на задоволителни резултати, потребителят променя заявката си. Поради това, основна задача при реализиране на адаптивно търсене е да се идентифицират методите, подходите и средствата за адаптивност, които да помогнат на потребителя да получи подходящи ресурси с най-малко на брой промени на заявката от негова страна.

В системата Share.TEC са реализирани две отделни възможности за търсене – свободно търсене и разширено търсене [7, 94]. Свободното търсене осигурява прост и лесен за употреба интерфейс, подобен на интерфейса в търсачката Гугъл. Процесът на търсене при използване на този интерфейс включва въвеждане на свободен текст (ключови думи) от потребител и извеждане на получените резултати. Разширеното търсене, от друга страна, дава възможност освен за въвеждане на свободен текст, така и за избиране на стойности от списъци и падащи менюта. Тези стойности са зададени в съответствие с използваните метаданни в системата. Очевидно е, че използването на разширеното търсене в системата дава по-точни и персонализирани към индивидуалните потребности на потребителя резултати. Главния недостатък на този тип търсене е, че той добавя допълнителен слой на трудност и комплексност за потребителя в процеса на търсене. Многократно е доказвано, че свободното търсене по ключови думи е предпочитано от потребител на дадена система, защото то не изисква допълнителни усилия [4].

Взимайки предвид всички тези предпоставки, този дисертационен труд се цели в намирането на решение, което да комбинира предимствата на двата метода за търсене в системата, като преобразува автоматично свободното търсене в разширено търсене.

Нашият подход е насочен към потребител предпочитащ и използващ свободното търсене по ключови думи. В рамките на настоящия труд ние предлагаме решение, което да компенсира недостатъците на свободното търсене, изградено на базата на прилагане на адаптивна функционалност (адаптивно търсене). Предложеното адаптивно търсене се реализира посредством разширяване на оригиналната заявка на потребител с термини, получени от *Бейсов модел за адаптиране и персонализиране*. Предложеният метод за адаптивност е базиран на изполване на стереотипи за моделиране на потребител. Този тип адаптивност се нарича „групова адаптивност“ и отразява взаимодействието на еденичен потребител със системата, но е повлиян от характеристиките на група от потребители на системата.

В системата Share.TEC се използва групова адаптивност, реализирана чрез имплицитни (косвени) стереотипи, базирани на т.нар. кръгове на сходство. Групите при този подход са виртуални и се изчисляват „при поискване“. За разлика от тях „физическите“ или изрични стереотипи са създадени и се управляват като изрични единици в системата и могат да осигурят постоянно групиране на потребителите. Кръгът на сходство за даден потребител се определя като множеството от профилите на всички потребители, които са близо до неговия профил по отношение на предварително определена метрика. Съществуват два параметъра, които могат да се използват за контролиране на кръга: (1) капацитет на кръга; и (2) радиус на кръга. Капацитетът на кръга определя броя на членовете, които може да притежава кръга. Радиусът на кръга задава нивото на сходство, с оглед на използваната метрика, което се смята за приемливо. Чрез задаване на стойности на тези параметри, потребителят може да оказва контрол върху своя кръг, например като намалява или увеличава капацитета на кръга, той влияе на неговия размер. Сред предимствата на този подход са следните: потрбителите могат да контролират кръга, всеки потребител има свой собствен кръг от подобни потребители, кръговете са лесни за изчисление и не изискват значителни изчислителни ресурси и др.

Въпреки тези предимства, този подход крие и редица недостатъци и слабости, идентифицирани в Share.TEC Deliverable 5.5 Development and pilot deployment of the integrated system [86]. Сред недостатъците на подхода е, че той може да доведе до групиране на потребители, които не са реално сходни. Например, ако даден потребител е уникален в някои отношения, тогава, когато системата търси определен брой сходни профили на неговия профил, тя може да върне няколко

несходни, просто защото няма други. Решение на този проблем е задаване на гранична стойност, отвъд която сходството да се приема за реално съществуващо сходство. Независимо кой параметър (капацит, радиус или и двете заедно) се използва за настройване на кръга, за постигане на добри резултати е желателно тяхното ръчно контролиране от страна на потребителя. Слабост на този подход е и това, че той не осигурява стабилно и постоянно групиране на потребителите. Групите при този подход са виртуални и се изчисляват „при поискване“, при това те са трудни за анализиране. Не е възможно и задаване на стойности по подразбиране на потребителите, поради това, че кръговете не са реални групи. Адаптивността при кръговете на сходство се постига, като предпочитанията на всички потребители от даден кръг се агрегират в предпочитания на кръга, които могат да повлияят на представянето на индивидуалните предпочитания на потребителя, чиито профил е послужил за формиране на кръга [7].

Общ недостатък на имплицитните стереотипи е че, те не осигуряват стабилно и постоянно групиране на потребителите. Тъй като групите при тях са виртуални, те са трудни за анализиране и за поддържане на социални функции от система за препоръчване. Предимство на експлицитните стереотипи е че, те осигурят постоянно групиране на потребителите. Поради това в Share.TEC Deliverable 5.5 е предвидено тяхно възможно създаване и интегриране в системата Share.TEC в бъдеще [86]. На базата на предложените в настоящата работа стереотипни групи е създаден модел за извод над знанието за потребител, стоящ в основата на адаптивно търсене. Детайлен преглед на системата Share.TEC и на реализираната адаптивност в системата е представен в секция „Системата Share.TEC“ към Глава 5 от настоящия труд.

Настоящето изследване е мотивирано от желанието да се разработи новаторски модел за извод при несигурност над знанието за потребител, който да служи за реализиране на адаптивно търсене в дигитална библиотека. Задачата за откриването на новаторски методи, способи и техники за реализиране на адаптивно и песонализирано търсене е значима и приоритетна цел и представлява значителна част от изследванията, свързани с Интернета на бъдещето.

Част от резултатите на тази дисертация са постигнати по време на специализация в университета Ла Сапиенца, по проект Unite, реализиран по 7 рамкова програма, чиито приоритет са откриване на нови решения за Интернета на Бъдещето (Future Internet).

1.1 Обект и предмет на дисертационния труд

Обект на изследването са средите, доставящи знания, и по-конкретно дигиталните библиотеки, в които да се реализират услуги за адаптивно търсене, като от една страна се използват водещи технологии за моделиране на знанието за потребител, а от друга се прилагат стратегии за адаптиране, базирани на извод над знанието за даден потребител.

Предмет на изследването е идентифициране на нови подходи за реализиране на адаптивно търсене в дигитална библиотека, използвайки Технологиите на сематичния уеб за моделиране на знанието в предметната област и вероятностни модели за извод над това знание.

1.2 Цели на дисертационния труд

Цел на Изследването е разработването на иновативна архитектура на Функционален модул за адаптивно търсене в дигитална библиотека и представяне на създадената архитектура, нейните компоненти, базова схема на адаптивна и персонализираща логика за реализиране на адаптивно търсене, както и разработените модели, схеми и алгоритми.

Целта на изследването е свързана с подпомагане на потребителя в неговото търсене на информационни ресурси в дигитална библиотека, в резултат на което потребителят да получи резултати от търсенето, които са съобразени с неговите характеристики и нужди.

Постигането на целта на изследването се осъществява с изпълнението на следните подцели:

Подцели

- Идентифициране на новаторски метод за моделиране на групи от потребители и прогнозиране на техните интереси чрез Бейсов модел.

- Създаване на „физически“ експлицитни стереотипи за системата Share.TEC.

- Идентифициране на нуждите на потребител в системата Share.TEC, за целите на адаптивно търсене.

- Създаване на адаптивна и персонализираща логика за реализиране на адаптивно търсене.

- Разработка на архитектура, модули и компоненти на предложеното решение.

- Софтуерна реализация на избртани модули от архитектурата на и тяхната интеграция в системата Share.TEC.

- Създаване на методи и стратегии за оценяване на ефективността на предложеното решение.

- Идентифициране на възможностите за бъдещо развитие на предложеното решение и разширяване на неговите бъзможности.

Реализиране на целите и подцелите на изследването се постига чрез решаване на Задачите на дисертационния труд, представени в следващата секция.

1.3 Задачи на дисертационния труд

В рамките на настоящата работа ще предложим решение, което дава отговор на следните поставените задачи:

Задача 1: Запознаване с проблема и водещите теории в областта. Проучване и анализ на събраните данни, което намира израз в изготвяне на научен обзор. Изпълнението на тази Задача е свързано с изпълнението на следните подзадачи:

- Запознаване с водещите научни теории и научно-приложни разработки в областта на адаптивното търсене. Проучване на съществуващи системи и техните персонализационни подходи и адаптивни стратегии. Идентифициране

на методите, чрез които може да се реализира адаптивно търсене и анализ на характерните им особености.

- Запознаване със съществуващите методи, подходи и средства за реализиране на групова адаптивност. Проучване на системи, които използват стереотипи за моделиране на потребителите в системата.

- Анализ на методите и средствата за представяне на знанието за потребител в системата. Проучване на типовете и видовете потребителски модели.

- Проучване на характерните особености на Бейсовите мрежи и вероятностния извод. Аналитично изследване на съществуващите Бейсови модели и тяхното поле на приложение. Идентифициране на плюсовете и минусите на съществуващите модели с цел създаване на модел, който представлява оптимално решение. Изследване на адаптивни обучителни системи, които базират адаптивната си стратегия на Бейсови мрежи.

- Аналитично изследване на Технологиите на Семантичния уеб и в частност на онтологиите, като средство за задаване на споделена семантика в предметната област. Запознаване с езиците, стандартите и протоколите на Семантичния уеб.

- Изследване на дигиталните библиотеки и съвременните подходи за реализация на адаптивност в тях.

Задача 2: Разработване на архитектура на Функционален модул за адаптивно търсене в дигитална библиотека. Реализирането на тази задача е свързано с изпълнението на следните подзадачи:

- Разработването на базова схема на адаптивна и персонализираща логика за реализиране на адаптивно търсене.

- Разработване на новаторски модел, базиран на Бейсова мрежа за извършване на извод над знанието за потребител.

- Разработване на „физически” стереотипи, които да групират потребителите на системата, въз основа на техни съществени сходства.

- Идентифициране на метод за класифициране на потребител към стереотип/и.

- Разработване на стратегия за класифициране на потребител към представените стереотипи.

- Разработване на модели за представяне на потребител и стереотип, които да позволят оценяване на тяхното сходство.

- Разработване на избрани модули от архитектурата на предложения модул за адаптивно търсене в дигитална библиотека.

Задача 3: Реализация и експериментално внедряване на компоненти от предложената архитектура в системата Share.TEC. При изпълнението на тази задача трябва да решим следните подзадачи:

- Реализиране на алгоритъм за съставяне на Бейсова мрежа.

- Избор на алгоритъм за обновяване на Бейсовата мрежа.

- Избор на среда за разработка, която да отговаря на нуждите за създаване на софтуерна реализация.

- Софтуерна реализация на избрани модули от архитектурата.

- Внедряване на реализираните модули в системата Share.TEC.

Задача 4: Анализиране на предложеното решение. Представяне на подход за оценяване на резултатите от предложената стратегия за реализиране на адаптивност. Очертаване на насоките за бъдещо развитие на предложеното решение. При изпълнението на тази задача трябва да решим следните подзадачи:

- Създаване на стратегия за оценяване на ефективността на предложеното решение.

- Очертаване на насоки за бъдещо развитие на предложеното решение, включващи възможности за разширяване на неговата употреба.
- Оценяване на възможността за интегриране на Функционалния модул в други системи.

1.4 Описание на съдържанието на главите

Трудът обхваща Увод, шест Глави, Заключение и пет Приложения, списък на използваната литература, списък на авторските публикации и списък на участия в научни форуми и събития.

В **Глава 1 „Обща постановка на задачата”** е представен проблемът за реализиране на адаптивно търсене на базата на стереотипи в дигитална библиотека. Представено е предложеното решение на проблема и са очертани неговите предимства в сравнение с други подобни решения. Формулирани са обект и предмет на дисертацията, както и цели и задачи на изследването.

В **Глава 2 „Адаптивност и персонализация. Методи, подходи и средства за реализиране на адаптивно търсене“** е направен преглед на езиците, стандартите и протоколите на Семантичния уеб. Проследено е актуалното състояние и са очертани развойните тенденции. Главата продължава с аналитично изследване на методите, подходите и стратегиите за реализиране на персонализация и адаптивност, като особено внимание е отделено на адаптивното търсене, при което са проследени средства, стандарти и добри практики. Разгледани са примерни системи, чиито опит и разработки са от съществено значение за настоящия труд. Направено е аналитично изследване на модел на потребител, като са разгледани главните типове модели и са идентифицирани техните характерни особености. Изследвани са съществени характеристки и на стереотипните модели, като са разгледани: подходите за определяне на стереотипните групи; методите за класифициране на потребител към стереотип; структурата на стереотипите в системата и др. Изследвани са съществуващи системи, в които адаптивността е базирана на стереотипи. Направен е сравнителен анализ на шест такива системи въз основа на показателите: тип на системата; дали

стереотипното моделиране е единствен или допълващ метод в системата; определяне на стереотипните групи; класифициране на потребител към стереотип; структурата на стереотипите; представяне на стереотипите в системата; реализиран метод или стратегия на адаптивност. Разгледани са добри практики и новаторси насоки в системите за извличане на информация и в дигиталните библиотеки. Получените резултати са анализирани и обобщени.

В **Глава 3 „Бейсови (вероятностни) мрежи. Методи, подходи и стратегии за изграждане на Бейсови модели"** последователно и методично са разгледани базови научни теории и постановки, аналитично е изследвано състоянието и новите тенденции при изграждане и използване на вероятностни модели в адаптивни системи. Изследвани и анализирани са различни типове Бейсови модели, като са проучени тяхната структура и параметри. Детайлно са изследвани три канонични модела, при което са изведени техни съществени характеристики и е направен сравнителен анализ между тях. Изследвано е приложението на Бейсови модели в редица системи, като фокусът е върху адаптивните обучителни системи. Изследвани и анализирани са използваните в тях методи, стратегии и подходи за реализиране на адаптивност на базата на Бейсов модел.

Глава 4 „Архитектура на Функционален Модул за адаптивно търсене в дигитална библиотека" представя детайлно описание на предложеното решение. Представена е базова схема на адаптивна и персонализираща логика за реализиране на адаптивно търсене, в която са проследени основните дейности, осъществявани от потребител и техните отговори от системата. Описанието на предложения Функционален модул за адаптивно търсене е придружено със схеми и диаграми. Детайлно са представени отделните компоненти от архитектурата, като са изложени основните функции, които се осъществяват от всеки от съставните под-модули на Функционалния модул. Представен е предложеният Бейсов модел за извод над знанието за потребител, като са разгледани неговата структура и параметри. Представени са разработените „физически" стереотипи за системата Share.TEC.

В **Глава 5 „Реализация на модул за изпълнение на адаптивна и персонализираща логика, посредством извод в Бейсова мрежа"** е разгледана

накратко системата Share.TEC, като са очертани нейни съществени характеристики. Представената информация е анализирана и обобщена. Главата продължава с описание на реализацията на под-модули от предложеното решение, като специален фокус е поставен върху *Модул за изпълнение на адаптивна и персонализираща логика*. Детайлно са представени използваните алгоритми, средата за разработка и софтуерната реализация. Показани са основните класове от средата за разработка, използвани за реализацията на разработените алгоритми. Информацията е придружена със схеми, таблици и листинги със софтуерен код.

В **Глава 6 „Изследвания и експерименти“** са проследени ключови моменти в процеса на създаване на системата. Представени са серии от тестове и експерименти от решаващо значение, които доказват ефективността на предложеното решение и неговата способност да удовлетвори поставените изисквания.

В Заключението са представени основните приноси на дисертационния труд, които са в две насоки: научни приноси и научно-приложни приноси. Представени са и насоките за бъдещо развитие. Дисертацията завършва със списък на използвана литература и списък на публикациите, свързани с дисертационния труд.

Представени са пет Приложения: Софтуерен код на Бейсов модел за адаптиране и персонализиране; Списък на променливите в БМ; Вектори на стереотипите; Софтуерен код, представящ реализация на избрани операции над Бейсов модел за адаптиране и персонализиране и Релации между променливите и техните предшественици в Модела.

Глава 2. АДАПТИВНОСТ И ПЕРСОНАЛИЗАЦИЯ. МЕТОДИ, ПОДХОДИ И СРЕДСТВА ЗА РЕАЛИЗИРАНЕ НА АДАПТИВНО ТЪРСЕНЕ

Нуждата от специализирани среди, доставящи знание на потребителите по всяко време и от всяко място нараства с течение на времето. Такива среди са дигиталните мултимедийни библиотеки, в които се съхранява разнородна по своята форма и обем дигитална информация. Наред с тяхното широко използване се появи необходимостта от наличието на способи и подходи за адаптивно търсене в дигитална библиотека, при което доставените резултати да са съобразени с индивидуалните предпочитания на потребителя. Появата на нови методи и стратегии за адаптивно търсене са свързани с използването на такива техногии, които да могат да отговорят на повишените изисквания на системите от ново поколение. Използването на технологиите на Семантичния уеб за реализиране на адаптивно търсене е свързано със способността им да създават контекстно зависими, машинно разбираеми и формулирани на естествен език потребителски заявки. Нещо повече, възможно е създаването на динамичен потребителски модел от ново поколение, който точно, бързо и ефективно да моделира интересите на потребител. В последните години все по-широко се използват онтологиите, с помощта на които е възможно задаване на споделена семантика за предметната област.

2.1 Технологии на Семантичния уеб – езици, протоколи и стандарти

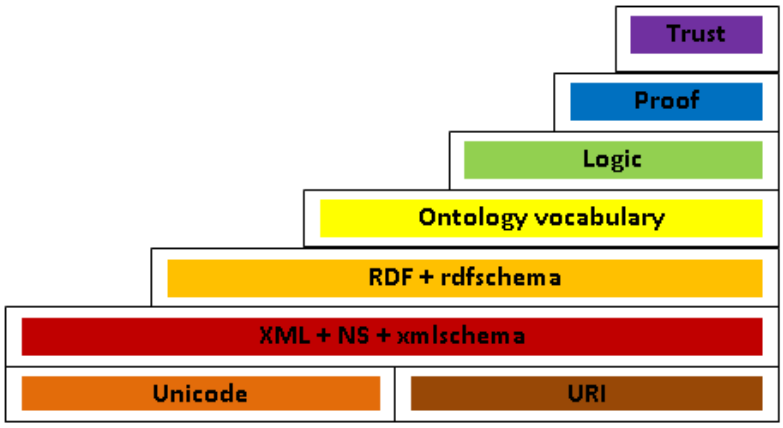
Семантичният уеб съдържа набор от технологии и стандарти, позволяващи добавяне на машинно разбираемо значение към уеб услугите и приложенията. Технологиите на Семантичния уеб стоят в основата на редица системи, които доставят персонализирано и адаптивно съдържание. Сред предимствата на тези технологии е възможността за ефективно предаване, търсене, смесване и обработване на данни. Технологиите на Семантичния уеб уверено навлизат в областта на електронното обучение и дигиталните библиотеки, за които приоритет е възможността за бързо и ефективно доставяне на подходящо съдържание. Технологиите на Семантичния уеб намират приложение за описание на информационни ресурси в дигиталните

библиотеки по систематичен и универсален начин и задават средства за обработване и обобщаване на големи количества от разнородна мултимедийна информация. На базата на тези технологии е възможно изграждането на потребителски модел от ново поколение, който точно, бързо и ефективно да моделира интересите на потребител. Такива модели стоят в основата на различни персонализационни услуги, сред които и адаптивно търсене [31].

Техниките и методите на Семантичния уеб намират приложение при анотиране на информационни ресурси, представяне на факти по формален начин и извършване на извод над тях, в семантичен анализ, разпознаване на обекти и тяхното индексиране, при взаимодействието човек-компютър, описание на дадена област от знание чрез онтология, създаване на модел на потребител и др.

Понятието Семантичен уеб е въведено за първи път от Тим Бърнърс-Лий, директор на World Wide Web Consortium (W3C), като с него се означава следващото поколение на WWW, при което се добавят машинно-обработваеми семантики към ресурсите. Посредством Семантичния уеб се цели създаването на общ фреймуърк, който да позволява данните да бъдат споделени и използвани повторно в различни приложения и от разнородни общности. Тим Бърнърс-Лий разработва модел на Семантичния уеб, който обхваща набор от езици и стандарти.

Този модел представлява седемслойна струкура от протоколи и стандарти. Фиг.1 представя модел на Семантичния уеб, по модела на [31].



Фиг.1 Модел на Семантичния уеб, по модела на [31]

На най-долния слой на седемслойния модел се намират наборите от символи Unicode и Uniform Resource Identifiers (URIs). Символите Unicode правят възможно използването на всички световни азбуки, а URIs осигуряват уникалното представяне на елементите чрез низове от символи, с които се идентифицират ресурсите в WWW. Над тях се разполагат eXtensible Markup Language (XML) и XML Schema. Езикът eXtensible Markup Language осигурява елементарен синтаксис за структуриране на съдържанието, но не дава възможност за описание на смисъла на това съдържание. Съществуват алтернативни езици за описание на структурата на ресурс, като например предшественика на XML - SGML и езикът Turtle, които са с по-ограничено приложение. XML Schema е език за ограничаване на структурата и съдържанието на елементите, съдържащи се в рамките на XML документите. На третия слой отдолу нагоре се намира езикът Resource Description Framework (RDF), посредством който се добавя семантика за даден ресурс, чрез тройките „субект, обект, предикат". Изразителната сила на езика RDF е ограничена и не дава възможност за моделиране на по-сложни отношения. На същия слой се намира езикът RDF Schema (RDFS), посредством който могат да се дефинират структури като класове и подкласове и да се изразяват йерархии от класове.

На четвъртото ниво в седемслойния модел на Семантичния уеб се намира онтологичния слой. На този слой се задават Онтологиите, чрез които е възможно изразяването на по-сложни семантични структури в сравнение с RDFS. Най-широко разпространеният онтологичен език за момента е Web Ontology Language (OWL), който заменя по-стария DAML+OIL. Последният е получен от обединението на Ontology Inference Layer (OIL) и DARPA Agent Markup Language (DAML), разработвани независимо в Европа и САЩ. Онтологичният език OWL има голяма изразна сила и позволява автоматично генериране на изводи. OWL притежава богат речник за описание на свойства и класове, като позволява описването на отношения между класовете като кардиналност и еквивалентност. Съществуват 3 подезика на OWL, които са с различна изразна сила: OWL Lite, OWL DL и OWL Full. Задаването на заявки в семантични източници на данни е възможно чрез езика SPARQL. На онтологичния слой действат и форматите RIF и SWRL, чрез които могат да се задават правила, които да се изпълняват над колекцията от факти.

Най-горните три слоя на седемслойния модел на Семантичния уеб са по-слабо разработени в сравнение с останалите четири. Те са логически слой,

доказателствен слой и слой на доверието. Логическият слой осигурява изпълнение на твърденията, дефинирани на онтологичния слой. На доказателствения слой се реализират доказателства, посредством правила за истинност и доверие.

2.2 Адаптивност и персонализация – понятия и дефиниции. Адаптивни системи

Адаптивността е свързана с цялостната способност на системата да се адаптира към потребител. Тя улеснява намирането на най-подходящото съдържание за даден потребител и помага при навигиране из системата. Адаптивността може да бъде контролирана явно от потребител, или да бъде осигурявана имплицитно от системата, чрез анализиране на поведението на потребител и неговото взаимодействие със системата [86].

Barra et al [6] дефинират адаптивността като „способността да се разбира потребителското поведение, така че да се взима предвид неговото ниво на познание с цел да му се предоставят точните документи“. Адаптивността е постоянен и итеративен процес, който продължава през целия жизнен цикъл на системата.

Адаптивността се реализира чрез набор от методи и подходи, свързани със събиране и анализиране на знание за потребителя и прилагане на адаптивни стратегии. Това се осъществява чрез: (1) събиране на статистика за потребителя, (2) изграждане на потребителски модел, (3) анализиране на знанието за потребителя от модела, (4) определяне на това, как системата да се адаптира максимално добре към потребителя, (5) самия процес на адаптивност.

Груповата адаптивност е адаптивност, която засяга взаимодействието с определен потребител, но е повлияна от взаимодействията на много потребители. Груповата адаптивност е свързана с разделяне на потребителите на групи, въз основа на една или няколко характеристики. Груповата адаптивност се реализира посредством използването на изрични (експлицитни) или косвени (имплицитни) стереотипни групи.

Двата главни подхода за реализиране на персонализация са базирани на измерване на сходството между потребител и съдържание (content-based) или между отделните потребители в системата (collaborative-based). Първият подход разглежда всеки потребител изолирано от останалите потребители в системата, при което се

оценява степента на сходство между интересите на потребителя и наличните ресурси в колекцията. Вторият подход оценява сходството между потребителите, при което може да ги класифицира в групи според техните интереси. Този подход се базира на предположението, че потребителите със сходни интереси е вероятно да харесат едни и същи информационни ресурси. При откриване на такова сходство на новия потребител се предлагат ресурси, харесани от предишни потребители в системата. Съществуват и други подходи, които са по-рядко срещани, като такива базирани на правила или на демографските характеристики на потребителите, както и хибридни подходи, които комбинират няколко подхода. Подходите за персонализация могат да се обобщят в следните групи:

- *Подходи базирани на съдържание (content-based)* – оценява се сходството между интересите на потребител и наличните ресурси в системата. При този подход се препоръчват ресурси, които са подобни на ресурсите, харесани преди от същия потребител. Този подход притежава някои ограничения, като например все по-тясното фокусиране в интересите, които потребителят е изявил в системата. Този недостатък е свързан с това, че на потребителя се показват все едни и същи теми и не се експлоатират негови потенциални интереси [72].

- *Колаборативен подход (collaborative-based)* – този подход търси да установи степентта на сходство на нов потребител със съществуващите потребители или групи от потребители в системата. Негово предимството е това, че се използва обобщения опит на голяма група от предишни и настоящи потребители с цел да се идентифицират ресурси, подходящи за нов потребител [43].

- *Подход базиран на демографски характеристики* – този подход обхваща системи, които използват демографските характеристики (пол, възраст и др.), като база за определяне на сходството между отделните потребители. Обикновено този подход не е достатъчен сам по себе си и се комбинира с други подходи [71].

- *Подходи базирани на правила* – при тези подходи адаптивността се реализира на базата на правила, с чиято помощ се извеждат заключения за потребител на базата на наличната информация за него. Обикновено в този тип

системи потребителите са насърчавани да отговарят на набор от предварително формулирани въпроси, чрез които се събират данни за тях [100].

- *Смесени/хибридни подходи* – това са подходи, които използват комбинация от два или повече различни подхода за персонализация, като по този начин избягват ограниченията на отделните подходи [5].

В общия случай адптивността се постига, като се прилагат адаптивни методи и стратегии над модел на потребител. Съществуват различни подходи за моделиране на потребител, главните от които са: (1) класифициране на потребител към стереотипни групи (стереотипи); (2) използване на техники за научаване на модела на потребителя от неговото поведение в системата; (3) изследване на приликите между нов потребител и съществуващи потребители в системата и др. Адаптивността намира приложение в различни типове системи, като например в системите за интелигентно обучение, в системите за достъп до информационно съдържание, в електронните каталози, в системите за здравна помощ, препоръчващите системи и др. В хипермедийните системи адаптивността се изразява в адаптиране на съдържанието, адаптиране на представянето на резултатите или адаптиране на структурата [52]. Процесът на адаптивност и персонализация в средите доставящи знания цели откриването и представянето пред потребителя на тези информационни ресурси, които са най-подходящи за него [96].

В средите за електронно обучение адаптивността и персонализизацията се характеризират с редица особености, които произтичат от спецификите на процеса на обучение като дейност. В процеса на обучение от съществено значение е откриването на най-подходящите за даден обучаем учебни ресурси. Поради това персонализацията насочена към идентифицирането на такива учебни ресурси е от ключово значение. В тези среди персонализацията се разбира главно като персонализиран достъп до учебни единици. В този процес се взимат предвид както характеристиките на обучаемия, така и неговото поведение в системата. Целта на средите за електронно обучение е да улеснят процеса на придобиване на знания от обучаемия и да направят обучението максимално ефективно. Поради това адаптивността и персонализацията в тези среди е насочена към модифициране на предлаганото учебно съдържание в съответствие с обучаемия с цел улеснявяне на процеса на усвояване на знания. Обучаемите се различават един от друг въз основа на различни характеристики, като стил на учене,

когнитивни особености, ниво на познание в дадената област и др. и тези характеристики са в основата на изграждане на модел на обучаемия.

Сред системите, за които адаптивността и персонализацията са от съществено значение, са дигиталните библиотеки. Дигиталните библиотеки са онлайн среди за доставяне на знания, в които информационните ресурси се събират, класифицират и управляват в съответствие с утвърдени принципи за развитие на колекциите. В рамките на една дигитална библиотека се осъществява дейност по управление на информационни ресурси, индексиране и анотиране, сортиране, класифициране, добавяне на метаданни към обектите и извличане на информация (търсене). Целта на дигиталната библиотека е да прави възможен достъпа до дигитални информационните ресурси, по структуриран и надежден начин, обединявайки по този начин информационните технологии, образованието и културата в едно цяло [99].

Адаптивостта и персонализацията в дигиталните библиотеки се осъществява в следните направления:

- *Адаптивно търсене* на информационни ресурси, на базата на индивидуалността на потребителя.
- *Персонализирано представяне* на съдържанието и на графичния интерфейс на средата.
- *Препоръчване* на съдържание според характеристиките на потребителя.
- *Персонализиране и адаптиране на представеното съдържание*, както и на услугите, предлагани от системата, в съответствие с характеристиките на потребителя.
- *Персонализирано групиране и обобщаване* на информационните ресурси, като това групиране може да се осъществи въз основа на различни признаци [88].
- *Адаптиране* според използваните от потребителя устройства – настолен компютър, мобилни устройства и т.н.
- *Адаптиране* според местоположението на потребителя и близостта до обекти с по-особено значение, като паметници на изкуството, архитектурата и

т.н. С това се цели добавяне на допълнителен контекст към предлаганото съдържание.

- *Адаптиране* според характеристики на околната среда, като климат, час от деня и др.

- *Адаптиране* въз основа на създадената или разгледаната от потребителя информация.

Към момента последните четири направления представляват по-скоро насоки за бъдещо развитие, отколкото реализирана функционалност в дигиталните библиотеки. Обект на дискусия и изследване е и възможността за представяне на съдържание, което се получава от различини източници и се обединява преди да се представи на потребителя.

2.3 Модел на потребител – видове и характеристики

Потребителският модел се явява значима и съществена част в системите, доставящи адаптивно и персонализирано съдържание или услуга [9, 37].

Моделирането на знанието за потребител е централна изследователска задача при изграждането на АПС [11]. Моделът на потребител се използва от системите за адаптивно търсене по три различни начина: (1) когато служи за адаптиране на заявката за търсене, (2) когато се използва за преподреждане на резултатите, (3) когато действа като съставна част в процеса на извличане на информация [61]. Различните видове персонализирани и адаптивни системи прилагат различен похват при моделирането на потребител – системите за уеб персонализация използват проследяващи алгоритми, за да следват потребителя в неговото сърфиране из мрежата, докато адаптивните хипермедийни системи създават потребителски модел на базата на взаимодействието на потребителя със системата. на потребител може да включва различни негови характеристики и атрибутикато: демографски характеристики, ниво на експертност в дадена област, когнитивни особености и др. Класификацията на потребителските характеристики, представена от Brusilovsky [10] обхваща: познанието на потребител, неговите цел и квалификация, опит с хипертекст, интереси и среда. Gasparini et al [36] предлагат подход за моделиране на потребител в

среди за електронно обучение, който взима в предвид различни контекстуални аспекти като технологични, образователни, персонални и културни, с цел подобряване на персонализационните възможности на системата. Подобно изследване, за влиянието на културните аспекти при изграждане на дизайна на потребителския интерфейс е проведено от Callahan [13].

2.3.1 Характеристики на модела на потребител

Потребителските модели могат да се групират въз основа на различни критерии, като: вид на модела, начин на представяне, структура на модела, методи за конструиране на модела, целева група на модела, изменяемост на модела във времето, времеви обхват на модела, начин на събиране на данни за попълване на модела и др.

Съществува голямо разнообразие в моделите за описване на знанието за потребител, както в тяхната структура и представяне, така и по отношение на различни техни характеристики и параметри. Това позволява моделите да бъдат категоризирани като: персонални или групови, изрични (експлицитни) или косвени (имплицитни), динамични или статични, дългосрочни или краткосрочни и др. От съществено значение при създаване на модел е да се определи как ще се представя този модел, какви техники за конструиране ще използва, дали и какъв извод ще се извършва над него и др. Следва представяне на характеристики и атрибутина модела на потребител, които позволяват моделите да бъдат категоризирани в различни групи.

Целева група на модела - персонални и групови модели. Моделирането на потребител може да се реализира на базата на персонален модел или на групов модел. Персоналният модел представя знанието за конкретен потребител, без да отчита неговото сходство с други потребители в системата. Това знание включва конкретни факти за даден потребител, като демографски данни, индивидуални предпочитания, образователен статус и др. Този тип модел обикновено описва много точно и прецизно знанието за даден потребител. Трудностите, възникващи при употребата на този тип модел, са свързани с първоначалното инициализиране на модела, когато данните за потребителя са малко или въобще липсват. Освен това се оказва, че в ред системи такова детайлно представяне на потребителя не е нужно. В тези системи е по-удачно

използването на групов модел (стереотипи), който да представи характеристиките на дадена група от потребители.

Стереотипът се смята за агрегиран модел на група от потребители. Това означава, че ако два потребителя принадлежат към един и същ стереотип, то те принадлежат към обща група, държат се сходно и имат подобни интереси. Стереотипите представят група от характеристики, за които се приема, че дадена група притежава. Стереотипите могат да бъдат дефинирани в системата изрично с т.нар експлицитни стереотипи или косвено – имплицитни стереотипи. Експлицитните стереотипи са дефинирани „ръчно“ от експерти в областта, при което те съществуват и се управляват в системата като реално съществувщи единици, имат постоянно представяне и структура. Имплицитните стереотипи се наричат още „виртуални“, тъй като се променят динамично и представляват временно групиране на потребители въз основа на някакво сходство между тях. Стереотипите задават частично описание на набор от потребители и по този начин, когато определена характеристика на даден потребител липсва, тя може да бъде изведена от стереотипа към който потребителят принадлежи. В общия случай стереотипите обхващат набор от характеристики, които се приема, че са винаги верни за членовете на стереотипа. Основна техника, която се използва, за да се изгради модел на потребител, когато липсват данни за него, е посредством употребата на стереотипи. Типично за системите, използващи стереотипи е общото представяне на стереотипите и на потребителите чрез двойките характеристика-стойност. По този начин ако стойността на някоя характеристика за потребител липсва, тя може да бъде изведена от стереотипа, към който потребителят е класифициран. Стереотипите в най-общия случай служат за групиране на потребители с цел прилагане на някакви общи правила за адаптивност върху цялата група.

Методи за събиране на данни за потребител – експлицитни и имплицитни. Данни за потребителя в една система могат да се получат по два начина – директно, когато потребителя предоставя сам тези данни, и индиректно, когато тези данни се събират автоматично от системата. Методите за директно събиране на данни включват попълване на регистрационни форми и въпросници от потребителя или неговото участие в диалог човек-машина в системата.

Индиректното събиране на данни включва употребата на софтуерни агенти и проследяващи алгоритми, както и методи, при които се събират данни при взаимодействието на потребителя със системата. За индиректно събиране на данни,

както и за тяхното обработване се използват методи и техники, взети от областта машинно обучение. Сред по-широко използваните алгоритми са алгоритъмът LLSF, алгоритъмът за k-най-близките съседи (kNN), алгоритъмът на Рочио (Rocchio), алгоритъмът за неявно семантично индексиране (LSI) и др. Съществуват и по-нестандартни подходи, базирани на генетични алгоритми или на невронни мрежи.

Предимство на индиректното събиране на данни е, че чрез него могат да се проследят промените в интересите на потребител в реално време. Друго тяхно предимство е това, че потребителят не се ангажира пряко с предоставяне на данни за себе си. Недостатък на индиректното събиране на данни е, че този процес отнема от времето и ресурсите на системата, както и че е необходимо повече време за събиране на нужните за изграждане на модела данни.

Попълването на даден модел с данни може да се реализира като с данните, директно предоставени от потребителя, така и с индиректно събраните от системата данни или и чрез двете едновременно. В първия случай моделът на потребителя се нарича изричен (експлицитен), а във втория – косвен (имплицитен).

Изменяемост на модела във времето - статичен и динамичен модел. Друга характеристика на модела на потребител е свързана с изменяемостта на модела с течение на времето. Когато моделът е способен да се изменя във времето, той се счита за динамичен. Моделите, които не се променят във времето и пазят едни и същи данни се смятат за статични. Моделът на потребител в Системите за адаптивно и персонализирано търсене (САПТ) може да бъде както статичен, така и динамичен. Предимство на динамичния модел е способността му да отразява промените в интересите на потребителя с течение на времето. В системите от по-ново поколение моделирането на потребител е дълъг и продължителен процес, който продължава докато потребителят използва системата. Идеалната система е тази, която е способна да отчете автоматично промените в интересите и предпочитанията на потребителя и да ги отрази в неговия модел.

Времеви обхват на модела – дългосрочен и краткосрочен модел. Друга характеристика на модела на потребител е свързана с това, дали той представя устойчиви и дългосрочни интереси на потребителя или моментни (краткосрочни) интереси. Въз основа на този критерии моделите се делят на дългосрочни и краткосрочни. Тъй като моментните интереси се променят в кратки срокове и не

винаги са налични достатъчно данни за тяхното определяне, краткосрочните модели се смятат за по-трудни за създаване и управление, отколкото дългосрочните.

2.3.2 Моделиране на потребител – методи и подходи

Процесът на моделиране на потребител е свързан с избирането на съществени негови характеристики и атрибути, които да го представят в модела. Към тях спадат: демографски данни, като възраст и пол, ниво на познание в определена област, когнитивни и мисловни особености, начина на възприемане и учене и др. Съществуват различни подходи за подбор на тези характеристики на потребител, които да го представят в неговия модел, като широко известна е класификацията на Brusilovsky [10]. В системите за адаптивно електронно обучение и интелигентните обучаващи системи за съществени се приемат характеристиките на потребител, свързани с лекотата на усвояване на учебно съдържание, начина на учене, когнитивните особености и др. Това е така, защото целта на този тип системи е улесняване и подобряване на учебния процес и съпътстващите го активности. Моделът на обучаем в адаптивните обучителни системи се използва за доставяне на персонализирано учебно съдържание, моделиране на обучаемите в съответствие с техните особености и/или ниво на знание, препоръчване на подходящо учебно съдържание, динамично управление на учебния процес и др. Съществуват два главни стандарта за моделиране на обучаем - IEEE PAPI Learner [28] и IMS LIP [30]. Тези стандарти служат за създаване на унифицирани модели на представяне на обучаем, които да осигуряват съвместимост и многократна използваемост на моделите в различни системи за електронно обучение.

В САПТ в редица случаи при моделиране на потребител се използва историята на неговото търсене. В този случай моделът на потребител се попълва с индиректно събрани данни, състоящи се от въведените от потребителя термини за търсене. В други системи освен тях се анализират и резултатите от търсенето и/или разгледаните от потребителя документи. Съществуват подходи, които използват експлицитни данни, предоставени от потребителя в системата. Възможно е комбинирането на тези два подхода.

2.3.3 Архитектура на модел на потребител в САПТ

Сложността на модела на потребител зависи от целите и функциите на конкретната система. В своята най-проста форма потребителският модел представлява прост въпросник или регистрационен формуляр, съдържащ експлицитна информация, предоставена от самия потребител. В по-сложните случаи моделът се състои от динамични информационни структури, съдържащи както базова информация за потребителя, като образование и интереси, така и данни свързани с промяната в интересите и поведението на потребителя с течение на времето. Най-простата форма на представяне на модел на потребител е n-мерен вектор, състоящ се от термини. По-сложни форми на представяне на модела на потребител включват RDF тройки или онтология, семантична мрежа и др.

Главните видове модели на потребител в АПС са:

- Модел, представен като вектор от ключови думи
- Онтологичен модел
- Модел, представен като семантична мрежа/и [37]

2.3.3.1 Модел - вектор от термини

Най-простото представяне на модел на потребител в САПТ е като n-мерен вектор, съставен от термини. При този подход областта се разделя на n елемента (от които се интересува системата) и всеки се описва от един компонент на вектора. Така потребителят може да се представи чрез вектора:

$X = (x_1, x_2, ..., x_n)$, където стойността на x_i представя потребителската характеристика по отношение на i-я елемент в областта.

Тази стойност може да бъде булева, показваща наличие или отсъствие или може да бъде цяло число, показващо постигнатото ниво на знание относно i-я елемент [23]. Това представяне на потребител е лесно за реализиране и поради това има широка реализация в различен тип системи.

При попълване на този тип модел с данни за потребителя е важно да се определи кои термини и понятия са от съществено значение за представянето на потребителя. Съществуват различни методи за идентифицирането на тези термини, като те могат да бъдат предоставени изрично от потребителя или изведени автоматично от системата въз основа на действията на потребителя в системата (изпратени заявки, разгледани документи и др). В този тип модел в общия случай термините се характеризират с дадена тежест, която указва значимостта на даден термин за описанието и представянето на потребителя.

Векторният модел използва проста структура за представяне на знанието за потребител, в която няма смислови релации между отделните термини от вектора или отношения между тях. Най-простата реализация на този модел използва само един вектор, в който се съдържат всички термини, съществени за описанието потребителя. В по-сложните реализации отделни аспекти, интереси и характеристики на потребителя се представят чрез отделни вектори. При този подход се постига по-детайлно и точно описание на потребителя.

Таблица 1 представя сравнително съпоставяне на модели – вектори от термини, заедно с техни характеристики в няколко различни системи.

Таблица 1 Сравнение на модели – вектори от термини, адаптирано от [37] и допълнено

Система	**Представяне на модела**	**Методи за събиране на данни**	**Техники на конструиране на модела**
Amalthaea [65]	Единичен вектор от термини с тегла	Индиректно	Извличане на ключовите думи с най-голяма тежест
WebMate [16]	Отделен вектор за всеки интерес	Директно	Създаване на вектор от документ Сравняване на векторите на интересите Сливане на най-сходните вектори на интереси
Alipes [102]	Набор от вектори за всеки интерес	Директно	Създаване на вектор от документ Сравняване с векторите на интересите Добавяне към най-близкото съвпадние
MiSearch [89]	Вектор, съставен от класове с	Индиректно	Извличане на термини от заявките за търсене Сравняване с векторите на документите в

	дадена тежест		колекцията

Всички системи, представени в Таблица 1, използват като източник на данни уеб страници и като метод за оценяване на сходството – косинусовото сходство (Cosine similarity).

Предимство на модела вектор от термини, в сравнение с други видове модели, е неговата простота, изчислителна ефективност и доказана резултатност. Представянето на модела на потребител като вектор от термини е тясно свързано с Модела вектор-пространство.

Популярен и често срещан метод за представяне на модел на потребител и на документи (ресурси) е моделът вектор-пространство. В този модел всеки документ се представя като множество от термини или като n-мерен вектор [81].

Тази форма на представяне служи като основа за измерване на сходството между векторите на ресурсите и на модела на потребител. Ако сходството между векторите е в рамките на дадена граница, то се приема, че те са сходни. Моделът вектор-пространство намира широко приложение за измерване на сходството между заявка за търсене и наличните документи в дадена колекция от ресурси.

Един от най-популярните методи за измерване на сходството между два вектора в областта извличане на информация е т.нар косинусово сходство (Cosine similarity) [90]. Косинусово сходство е мярка за сходството между два вектора, което измерва косинуса на ъгъла между тях. Косинус от 0 градуса е 1 и е по-малко от еденица за всеки друг ъгъл. Така векторите с една и съща посока имат Cos = 1, перпендикулярните: Cos = 0, а противоположните: Cos = - 1. За удобство за всеки тъп ъгъл (cos < 0) се приема, че има мярка 0, а не отрицателна. Тогава сходството на вектора на потребител с вектора на стереотип е равно на косинуса на ъгъла между тях.

Ако единият вектор се представя като: $\vec{v} = (v_1, v_{2,...}v_n)$, а другият като: $\vec{u} = (u_1, u_{2,...}u_n)$, то сходството между тях се измерва като косинуса на ъгъла между тях:

$$\text{sim}(\vec{v}, \vec{u}) = \cos\boldsymbol{\vartheta} = \frac{\vec{v} \,.\, \vec{u}}{|\vec{v}||\vec{u}|}$$

Алтернативен метод за измерване на сходството е на базата на скаларното произведение на два вектора. Ако всички вектори са с еднаква дължина, то косинусът

на ъгъла между тях е същят като тяхното скаларно произведение (в Евклидовото пространство). Това сходство може да се измери и като:

$$sim(u,v) = \sum u_i \,.\, v_i$$

Тъй като всеки термин, който не присъства или в единия или в другия вектор има стойност за u_i или v_i равна на 0, сумирането се извършва само за термини, които са общи и за двата вектора.

Моделът на потребителя в системата Amalthaea [65] се представя като вектор с тегла, съставен от термини. Моделът се попълва с индиректно събрани от системата данни, но съществува и възможността потребителят да зададе директно термините за своя модел, в който случай те получават по-голяма тежест от индиректно събраните от системата. Изведените документи от търсенето на потребителя се конвертират във вектори, подобни на вектора на потребителя. Тежестта на отделните термини в тези вектори се изчислява с помоща на метода tf–idf, широко използван в областта извличане на информация [58]. Векторите на документите се сравняват с вектора на потребителя, посредством метода на косинусовото сходство и въз основа на това сравнение на потребителя се представят само тези документи, чиито вектори са сходни с неговия модел – вектор.

Подобен подход се използва в системата WebMate [16], където моделът на потребителя представлява набор от вектори от термини, по един вектор за всеки отделен интерес на потребителя. В тази система тежестта на термините се определя също с метода tf-idf, а сходството между два вектора се измерва с помоща на метода косинусово сходство [58]. В системата Alipes [102] всеки отделен интерес на потребителя е представен чрез три вектора - дългосрочен, краткосрочен (позитивен) и краткосрочен (негативен). Векторите на документите се сравняват с всеки от трите вектора с помоща на метода косинусово сходство.

В системата MiSearch [89] моделът на потребителя е вектор, съставен от класове с дадена тежест, при което този модел се попълва с термини от заявките за търсене. Документите в колекцията се представят също като подобни вектори. Системата MiSearch използва външна, референтна концептна йерархия - Open Directory Project (ODP), която служи за категоризиране на класовете с помоща на текстови класификатор. В тази система се използва моделът вектор-пространство, като за

изчисляване на сходството между модел на потребител и вектор/и на документ се използва формулата:

$$\mathrm{sim(user,doc)} = \sum_{k=1}^{N} wp_{i,k} \,.\, wd_{j,k}$$

където:

$wp_{i,k}$ е тежестта на понятието k в i-я потребителски модел

$wd_{j,k}$ е тежестта на понятието k в j-я документ [89].

Съществуват и други подходи за изчисляване на сходство между документ (ресурс) и модел на потребител, например чрез използване на Бейсов вероятностен класификатор и др.

Популярността на модела вектор-пространство се дължи на неговата простота и доказана ефективност. Резултатността от прилагането на този метод е пряко свързана с правилния подбор на термини за попълване на модела вектор на потребител.

2.3.3.2 Модел на потребител, базиран на семантична мрежа

Семантичната мрежа е абстрактна структура, която изразява семантични релации между понятия. Тя се реализира посредством възли и дъги, свързващи възлите. Възможно е задаването на тежести на възлите и на дъгите за числово изразяване на тяхната значимост в модела. Посредством семантична мрежа може да се моделира отношението, както между конкретни термини, така и между конкретен термин и абстрактен клас.

Най-простата реализация на модел на потребител, базиран на семантична мрежа е посредством единична семантична мрежа, съдържаща възли, представящи конкретни термини. В по-сложните реализации във възлите на мрежата могат да стоят класове и към тях да бъдат свързани термини. Съществуват системи, в които всеки отделен интерес на потребител се представя с отделна семантична мрежа.

Съществува известно сходство в структурата на онтологичния модел и модела, базиран на семантична мрежа. Структурата и на двата вида модела е съставена от възли и дъги, но при концептния модел във възлите стоят абстрактни понятия, докато при семантичната мрежа там могат да стоят и конкретни термини.

Моделът на потребител в системата WIFS [60] е частично базиран на семантични мрежи. В тази система потребителят директно задава своите интереси в процеса на регистрация в системата. Броят на интересите на потребителя може да нараства с течение на времето, като системата е способна да идентифицира индиректно новите му интереси. Тези интереси служат за категоризиране на потребител към даден стереотип. Стереотипите в системата са дефинирани от експерти, познаващи предметната област. Интересите на потребителя в системата WIFS се представят не чрез една, а посредством набор от семантични мрежи.

2.3.3.3 Онтологичен модел на потребител

Технологиите на Семантичния уеб дават рамка от протоколи и стандарти, които правят възможно унифицираното представяне на потребител в различни системи. Съществена технология за достигане до семантика в рамките на Семантичния уеб са онтологиите. Според известната дефиниция на Грубер [31] онтологията е „формална експлицитна спецификация на споделена концептуализация”. Онтологията се определя още като концептуален модел, който описва „нещата, които съществуват” в рамките на една дефиниционна област по съгласуван и формален начин. Онтологията задава стандартизирано представяне, което дава възможност за разбирането и споделянето на фактите от дадена област между хора и компютърни системи.

Съществена част от системите, доставящи адаптивност и персонализация, са базирани на някаква референтна понятийна йерархия или таксономия, която задава понятията и взаимовръзките в съответната предметна област. Таксономията на понятията определя главните „суперклас-подклас” отношения в предметната област.

Онтологичните модели използват онтология за представяне на знанието за областта, както и за потребителя. Онтологичните модели са способни да изразяват сложни отношения и да позволяват извършване на извод над съществуващите факти. Онтологичните модели, подобно на моделите тип семантична мрежа позволяват извеждането на нови факти от колекцията налични факти [26]. Тяхното предимство е това, че тъй като се основават на предварително дефинирани понятия и отношения те са устойчиви на терминологични вариации. Често срещан подход за моделиране на потребител чрез онтология е чрез използването на модел на наслагването (overlay

model), който представя знанието за потребител, като част от знанието за областта. Възможно е задаването на тежести към класовете в модела, чрез които да се изразява степента на интерес на потребител към дадено понятие. Sieg et al. [83] представят подход за създаване на онтологичен модел на потребител, на базата на онтология на областта, който използва тежести за изразяване на интереса на потребителя към дадено понятие. Съществуват подходи за създаване на модел на потребител от предметна онтология, с помощта на статистически методи и алгоритъма на разпространяващата се активация [48]. Този тип модели могат да се използват многократно, както и да се пренасят и интегрират в различни системи.

Онтологичните модели могат да се представят като вектор, чиито елементи са класове от дадена йерархия. Liu и Meng [56] представят система, в която моделът на потребител е вектор, съставен от класове. За идентифициране на тези класове се използва базата на Google Directory, която на свой ред използва ODP.

Онтологичен модел за представяне на обучаем е представен в [25, 26]. Този модел е базиран на пет категории, които са: идентификация, други потребителски характеристики (предпочитания, задачи, интереси), представяне в учебния процес (резултати, сертификати), информация за обучаващата организация и календар. Paneva [69] представя онтологично базиран модел на обучаем, разработен на базата на стандартите за моделиране на обучаем IEEE PAPI Learner [28] и IMS LIP [69].

Процесът на създаване на онтологичен модел, както и на модел, базиран на семантична мрежа, е сходен с този при модела вектор от термини. Първата стъпка е идентифициране на подходящите термини и понятия, описващи потребител. Този процес може да притежава допълнителен слой на сложност, свързан с класифициране на термините към класове от онтология или семантична мрежа. За тази цел обикновено се използва помощта на референтен речник или знанието на експерти в областта.

Методи за идентификация на потребител в системата

В общия случай персонализация се предшества от процес на идентификация на потребител. Идентификацията задейства процес на зареждане на модела на потребител и прилагане на съответните адаптивни стратегии. Когато потребителят е нов в системата се стартира процес на инициализация на неговия модел, както и

протичат активности, свързани с категоризация на потребител към дадена стереотипна група или клъстер от потребители. Обща практика в много системи е предлагането на интерфейс, чрез който потребителят сам да въведе данни за себе си и своите предпочитания. Тези данни се използват за попълване на потребителския му модел. Идентификацията на потребител в система се осъществява посредством набор от методи и техники, като: (1) употреба на софтуерни агенти; (2) регистриране (логване) в системата; (3) използване на прокси сървър; (4) чрез бисквитки, инсталирани на клиентската машина; (5) посредством идентификатори на сесия и др. Много добра точност и проследяване между отделните сесии се постига чрез регистриране на потребителя в системата. Предимството на този метод е, че той позволява проследяване и идентифициране на потребител не само между отделни сесии, но и при използване на различни устройства за достъп.

2.4 Използване на стереотипи за реализиране на адаптивност и персонализация

Стереотипите са въведени и развити първоначално от Rich [78, 79] и впоследствие широко използвани за моделиране на потребител в разнородни типове системи. Стереотипът може да се разглежда като колекция от характеристики, създадена за да представи описание на често възникващи ситуации. Rich [78] уподобява стереотипите на скриптовете на Schank et al [82] и фреймовете на Minsky [62]. Стереотипите намират приложение при моделиране на потребител в диалогови системи [70], за прилагане на психологическа перспектива [46] и др. Използването на стереотипи е често срещано в системите за информационно филтриране, като метод за разделяне на потребителите на групи, въз основа на техни общи характеристики. Ardissino и Sestero [2] използват стереотипи за моделиране на потребители, а Henze и Nejdl [45] ги използват за подобряване на процеса на учене в онлайн обучителна система. В мнозинството от случаите моделирането на потребител с помощта на стереотипи се използва, когато в системата няма достатъчно данни за потребителя, например когато той е нов в системата. В други случаи стереотипите се използват за разделяне на потребителите на групи, върху които се прилагат общи правила за адаптивност. Въпреки, че стереотипите са доказали своята ефективност, мнозинството от системите използват индивидуални потребителски модели. Тези модели се попълват с данни, директно предоставени от потребителя или индиректно събрани от системата.

Не винаги събирането на тези данни е възможно или лесно осъществимо. В тези случай стереотипите могат да допълнят липсващите данни за потребителския модел. Поради това индивидуалното моделиране и стереотипното моделиране не са конкурентни методи, а взаимнодопълващи се. Възможни са два подхода при използването на стереотипи в дадена система – като единствен и като допълващ метод. От съществено значение при използването на стереотипи е тяхното правилно определяне. Съществуват два базови подхода за определяне на стереотипите и техните характеристики – *поведенчески* и *математически*. При поведенческия подход стереотипите се определят от експерти, запознати с областта и нейните особености. Поведенческият подход се използва широко в системите за обучение, които адаптират нивото на взаимодействие човек-система към нивото на знанията на потребител [15, 17, 32]. В този тип системи се дефинират няколко нива на опит, знания и квалификация на потребителите, като експерти определят кое ниво на работа е подходящо за всяко ниво потребител. При математическия подход определянето на стереотипните групи се извършва посредством някаква форма на изчисление, като клъстеризиране. Въпреки, че този подход може да дава точни резултати, той изисква дълго време за събиране, анализиране и агрегиране на данните в системата. Стереотипните групи, които се формират чрез този подход, са изкуствени и не винаги отразяват реалното разпределение на групите от потребители. Поради това, мнозинството от системите, които използват стереотипи отхвърлят математическия подход в полза на поведенческия [64, 80]. Друг важен момент е класифицирането на потребител към стереотип, което се осъществява посредством един от двата метода: (1) *метод на тригерите* и (2) *метод на пълното сходство*. При метода на тригерите наличието на определена характеристика в модела на потребител се явяват задействащ механизъм (тригер) за неговото класифициране към стереотип. При метода на пълното сходство най-сходният стереотип за даден потребител се определя въз основа на всички характеристики в модела на потребител, без да се отдава предпочитание на конкретна характеристика.

Системата GRUNDY [78, 79, 80] е пионер в моделирането на потребител с помощта на стереотипи. GRUNDY е експертна система, която изпълнява функциите на библиотекар, като препоръчва книги на читателите. Моделът на потребител в системата е комбинация от три елемента: данни, предоставени директно от самия потребител; изводи на базата на предишни търсения на потребител; и факти за

потребител, получени от стереотипите, към който той е класифициран. В тази система стереотипите се използват за получаване на данни за потребител, когато информацията за него липсва или е непълна. Колкото „по-нов" е потребителят в системата, толкова по-голяма е частта от данни за него, получена от стереотипите. Ако потребителят е нов, в системата се активират тези стереотипи, които най-добре съответстват на отговорите, предоставени от него при регистрацията му. Ако потребителят не е нов, тогава се зарежда неговия модел. Системата препоръчва книги, които се смятат за подходящи за даден потребител, въз основа на неговия модел. Резултатите от използването на стереотипи в системата GRUNDY показват категорично тяхната ефективност. Стереотипите в системата са колекция от комбинации атрибут-стойност-рейтинг, които описват групи от потребители на системата. Чрез рейтинга се изразява степентта на сигурност, че всеки потребител, който се вписва в стереотипа, ще притежава дадена характеристика. Стереотипите в системата са подредени йерархично в насочен ацикличен граф (DAG), при което стереотипите от по-долните нива са специализация на тези от по-горните нива. В системата GRUNDY всеки стереотип притежава тригери, които се използват за класифициране на потребител към този стереотип. Когато в модел на потребител се открие параметър, който е „тригер“ за даден стереотип, и стойността на параметъра в модела на потребител е същата като тази в стереотипа, то тогава потребителят се класифицира към този стереотип. GRUNDY предлага подходящи книги, като използва следният подход: първоначално се извежда набор от характеристики от модела на потребител, притежаващи висок рейтинг; избира се случайна характеристика от тях и се извежда набор от книги, които притежават съответстващ параметър; сравняват се параметрите на потенциално подходящите книги с характеристиките в модела на потребител за откриване на най-доброто съвпадение.

Системата GUMS [32] представлява модул, който изгражда модел на потребител за интерактивни експертни системи в различни области. Този модел първоначално се инициализира чрез стереотипи. Системата е реализарана на Пролог и използва проста предикатна логика. Стереотипите се състоят от факти и правила, описващи съответен клас потребители. Системата позволява правилата и фактите в стереотип да бъдат или абсолютно верни или верни по подразбиране (ако няма информация за обратното). Базата от знания на системата, се състои от две части: колекция от вградени стереотипи и колекция от индивидуални модели на потребители.

Стереотипите са организирани йерархично, при което стереотипите от по-долен слой наследяват параметрите на стереотипите от по-горен слой. Характеристиките, с които потребител се класифицира към даден стереотип, се наричат "абсолютни факти". Потребителят може да бъде класифициран към даден стереотип на базата дори на един единствен "абсолютен факт", който изпълнява ролята на тригер. Адаптивността се реализира, като се използват правила, които извеждат нови факти за потребителя от съществуващите факти в базата от знания.

UM-tool [9] е експертна система в областта на компютърните науки. Системата персонализира взаимодействието човек-система на базата на стереотипи. В системата са разработени 24 стереотипа, всеки от които съответства на определено взаимодействие. Моделирането на потребител се свежда до идентифициране на подходящите за него стереотипи. По този начин индивидуалността на всеки потребител се представя чрез индивидуален набор от стереотипи. При влизане на потребител в системата се активира модул, който проверява дали има съществуващата информация за него. Ако такава липсва му се предоставя въпросник, чрез който се получат данни за него. Данните от въпросника се използва за класифициране на потребител към един или няколко стереотипа. В системата UM-tool стереотипите се описват чрез пет параметъра: професионален статус, квалификация и ниво на опитност и знания в три области, свързани с дадена тема. За всеки параметър са зададени набор от възможни стойности (например, стойностите за параметъра „ниво на опитност“ са: начинаещ, напреднал и експерт). Дефинираните в системата стереотипи имат един и същ обхват от стойности за различните параметри. Типът атрибути и възможните им стойности са определени от експерти в съответната област.

В разгледаните дотук системи: UM-tool, GUMS и GRUNDY стереотипите са подредени в йерархична структура, при която на най-високото ниво е дефиниран базов стереотип, със зададени стойности за базовите параметри. Стереотипите, които се намират по-долу в йерархията, наследяват стойностите от базовия стереотип. При този тип организация на стереотипите, колкото по-долу в йерархията е един стереотип, толкова е по-конкретен. Потребител, който принадлежи към определен стереотип в йерархията е свързан с "предшестващия" стереотип от по-високо ниво и наследява всички негови параметри. По този начин стереотипите се представят като фреймове [62].

Системата KNOME [17] е модул за моделиране на потребител в система за консултиране за операционната система UNIX. Чрез интерактивна сесия с потребител KNOME определя нивото му на експертност и създава модел на знанието на този потребител в областта UNIX. Това ниво се определя на базата на доказателства, получени по време на диалога човек-система. Въз основа на различните нива на експертност на потребител, системата решава какви отговори ще получи той при запитване в областта на UNIX. Всяко ниво на експертност съответства на стереотип в системата, като стереотипите са: новак, начинаещ, средно ниво и експерт. За разлика от стереотипите в системата GRUNDY, които се състоят от колекции от параметри, стереотипите в KNOME са реализирани като категории в рамките на системата за представяне на знания KODIAK. Информацията за UNIX в системата също е разделена на четири нива на трудност, съответстващи на четирите стереотипа. Реализирания в системата метод за разделяне на информацията, паралелно със стереотипно разделяне на потребителите се нарича двойно стереотипизиране. Когато потребител влезе в системата, първо се проверява дали вече съществува модел за него и в случай, че не съществува системата го разпитва, за да открие нивото му на знание за UNIX. В системата KNOME класифицирането на потребител към стереотип се реализира, като се измерва степентта на сходство на индивидуалния модел на потребителя с отделните стереотипи. Използват се различни методи за определяне на това сходство, сред които са: елиминиране - изключване на очевидно неподходящите стереотипи, дедукция - измерване на всички параметри на стереотипите и на потребителя и поставяне на положителна стойност за всички общи параметри, и отрицателна на всички останали и др. Стереотипът получил най-висок резултат се приема за най-сходен с даден потребител. Chang et al [15] разработват система, в която класифицирането на потребител към стереотип се осъществява само ако има пълно съответствие между всички парметри в индивидуалния модел на потребителя и тези на стереотипа. Техният подход за класифициране на потребител се базира на търсене на пълно сходство между потребител и стереотип. Stadnyk и Kass [92] разработват модел на информационна филтрираща система, в която потребителят се описва с индивидуален модел. Тяхната система цели автоматичното идентифициране с помощта на математически средства на групи от хора с общи характеристики, които могат да бъдат определени като стереотипна група.

В дигиталната библиотека Share.TEC е използван математическия подход Nearest Neighbourhood за динамично създаване на виртуални стереотипи или т.нар. кръгове на сходство. Системата групира индивидуалните потребителски модели на базата на сходство между тях и по този начин формира групи от потребители, подобни на стереотипните групи. Кръгът на сходство за даден потребител се определя като множеството от всички потребителски модели, които са близо до неговия модел по отношение на предварително определена метрика. Съществуват два параметъра, които могат да се използват за контролиране на кръга: (1) капацитет на кръга; и (2) радиус на кръга. Капацитетът на кръга определя броя на членовете, които може да притежава кръга. Радиусът на кръга задава нивото на сходство, с оглед на използваната метрика, което се смята за приемливо. Чрез задаване на стойности на тези параметри, потребителят може да оказва контрол върху своя кръг, например като намалява или увеличава капацитетът на кръга, той влияе на неговия размер. Недостатък на този метод е, че той може да доведе до групиране на потребители, които не са реално сходни. Например, ако даден потребител е уникален в някои отношения, тогава, когато системата търси определен брой сходни на неговия модели, тя може да върне няколко не-сходни просто защото няма други. Това е така, защото може да съществува разминаване между психологическо и математическо сходство, породено от различните средства за тяхното измерване. Решение на този проблем е задаване на гранична стойност, отвъд която математическото сходство да се приеме за психологическо сходство. Независимо кой параметър (капацитет, радиус или и двете заедно) се използва за настройване на кръга, за постигане на добри резултати е желателно „ръчното“ им контролиране от страна на потребител. Групите при този подход са динамични и се адаптират бързо при промяна на поведението/интересите на потребител. Слабост на този подход е, че той не осигурява стабилно и постоянно групиране на потребителите. Групите при този подход са виртуални и се изчисляват „при поискване“, при това те са трудни за анализиране. Не е възможно и задаване на стойности по подразбиране на потребителите, поради това, че кръговете не са реални групи. Адаптивността при използвания подход се реализира, като предпочитанията на всички потребители от кръга се агрегират в предпочитания на кръга, които могат да повлияят на индивидуалните предпочитания на потребителя, чийто модел е послужил за формиране на кръга [7].

Стереотипите в разгледаните до момента системи имат йерархична структура или нямат постоянна структура. Възможни са и други представяния на структурата на стереотипите, като: таблици, хипервръзки и правила. В структурата тип таблица стереотипите се представят чрез набор от параметри или полета, при което всеки стереотип може да се разглежда като колекция от параметри със зададени стойности. Името „таблица" идва от това, че всеки стереотип може да се представи като ред в таблица, в която параметрите са колоните [84].

Radde et al [76] предлагат система за моделиране на потребител, която може да бъде интегрирана в системи за препоръчване. Целта на тяхната система е да идентифицира „нуждите" на потребител и въз основа на тях да препоръча подходящ продукт. Този процес се реализира под формата на диалог между потребител и системата за препоръчване. Чрез този диалог се цели изясняване на предпочитанията на потребителя. Тяхната система имитира процеса на продажба, който се осъществява от човек консултант. Този процес е свързан с диалог, посредством който консултантът цели да категоризира (стереотипизира) клиента с оглед препоръчване на подходящ продукт. Сред критериите за оценка са: бюджета на клиента, финансовото му положение, възрастта му и др. Въз основа на характеристиките на клиента, той бива класифициран към даден стереотип и му се препоръчва подходящ продукт. Подходящият продукт е този, който притежава търсените от клиента атрибути и характеристики. За реализиране на тази задача е създаден модел, базиран на Бейсова мрежа, който се състои от три вида случайни променливи – стереотипи, нужди и атрибути. Бейсовата мрежа е трислойна с отношения между възлите от най-горния слой към средния слой и от средния слой към най-долния слой. Представеният модел позволява на системата за препоръчване да класифицира потребителите към стереотипи, да оцени техните нужди и накрая да изведе най-подходящите за тях атрибути на продукти. Стереотипните групи са разработени от експерти в областта. Класифицирането на потребител към един или повече стереотипи се реализира посредством диалог между потребителя и системата за препоръчване. Отношенията между слоевете на Бейсовата мрежа се представят от дъги, наречени „влияния" и „съвпадения", които притежават тежест и тип (положителен или отрицателен). Положителното „влияние" означава, че вероятността потребителят да има дадена нужда нараства, ако той е класифициран към съответния стереотип. И обратно - отрицателното „влияние" с свързано с намаляване на тази вероятност. Системата

работи, като се задават доказателства на случайните променливи, представящи стереотипите в модела и се извеждат вероятностите на атрибутите. Тези вероятности се използват като тежести в модифицирана SQL заявка за извличане на резултати от база данни.

Системата SeAN [1] е уеб базирана, адаптивна система за персонализиран достъп до новини. Тя използва стереотипи за създаване на първоначален модел на нови потребители. Класифицирането на потребител към стереотип/и се извършва въз основа на данните, предоставени при регистрирането му в системата. При първоначално влизане в системата, потребителят попълва регистрационен формуляр, който съдържа набор от въпроси, като: възраст и пол; ниво на образование, област на интерес и т.н. В системата съществува потребител „по подразбиране“ в случай, че потребителят не желае да попълни въпросника или предоставените от него данни са непълни. Стереотипите в системата притежават две групи от слотове, съдържащи параметри - „класификационни“ слотове и „прогнозиращи“ слотове. Дефинирани са четири „фамилии“ от стереотипи, като всяка от тях се използва за представяне на различно „измерение“ от характеристики на потребител. Четирите фамилии стереотипи са: интереси, експертност в областта, когнитивни характеристики и начин на живот. Класифицирането на потребител към стереотип се извършва като се сравняват данните, въведени от потребителя при неговата регистрация и параметрите в класификационните слотове на стереотипа. Потребителят се класифицира отделно във всяка фамилия от стереотипи. Тъй като стереотипът се представя като сечение от всичките си характеристики, сходството между потребител и стереотип се изчислява като произведение от приносите на всички слотове. Прогнозата за даден потребител се базира на стойностите в прогнозиращите слотове и изчисленото сходство между потребител и даден стереотип. Стереотипите от всяка фамилия поотделно предоставят собствена прогноза за потребител, като отделните прогнози накрая се сливат. Адаптивността в системата се реализира чрез правила, които използват стойностите от модела, за да изчислят каква е вероятността потребителят да иска да види дадена новина [1].

В Таблица 2 е представен сравнителен анализ на шест системи, които са съпоставени въз основа на признаците: (1) тип на системата; (2) дали стереотипите са единствен или допълващ метод за описание на потребител; (3) подход за определяне на стереотипите - поведенчески или математически; (4) метод на класифициране на

потребител към стереотип - метод на тригерите или на пълното сходство; (5) структурата на стереотипите; (6) представяне на стереотипите в системата; (7) метод/ стратегия на адаптивност.

Таблица 2 Сравнение на системи, използващи стереотипи, въз основа на техни признаци и характеристики

Система	**Тип на системата**	**Модели в системата**	**Метод за определяне на стереотипните групи**	**Класифициране на потребител към стереотип**	**Структура на стереотипите**	**Представяне на стереотипите и Модел на адаптивност**	**Метод /стратегия/ на адаптивност**
GRUNDY [78 ,79, 80]	Експертна система за препоръчване на книги	Индивидуален модел + стереотипи	Поведенчески	Метод на Тригерите	Йерархична структура	Стереотипите се състоят от набор от тройките (фасет, стойност, рейтинг)	Препоръки на базата на вероятностен извод
GUMS [32]	Модул за моделиране на потребител в експертни системи	Индивидуален модел + стереотипи	Поведенчески	Метод на Тригерите	Йерархична структура	Стереотипите се състоят от набор от факти и правила.	Извеждане на нови факти за потребител чрез правила в база данни. Използва се предикатна логика.
KNOME [17]	Модул за моделиране на потребител в консултираща система за UNIX	Индивидуален модел + стереотипи	Поведенчески	Метод на пълното сходство	Таблична структура	Стереотипите представляват категории в системата за представяне на знания KODIAK	Двойно стериотипизиране. Дедукция на базата на правила.
Share.TEC [7]	Модул за групова адаптивност в дигиталната библиотека Share.TEC	Виртуални стереотипи	Поведенчески	Метод на пълното сходство	Стереотипите не са постоянни и не са структурирани	Клъстери или „кръгове“ от потребители	Създаване на агрегиран модел, който афектира персоналния модел на патрона на кръга
SeAN [1]	Уеб адаптивна система за персонализиран достъп до новини	Индивидуален модел + стереотипи	Поведенчески	Метод на пълното сходство	Четири независими фамилии от стереотипи.	Всеки стереотип се състои от две групи слотове. Всеки слот съдържа вероятностна стойност	Правила, изчисляващи вероятността за интерес към съдържание
Система на Radde et al. [76]	Модул за моделиране на потребител в системи за препоръчване	Стереотипи	Поведенчески	Метод на пълното сходство	Таблично представяне (вектори)	Стереотипите са случайни променливи в трислойна Бейсова мрежа	Модифициране на SQL заявка към база с данни

Vassileva [101] предлага използване на стереотипи в хипертекстова адаптивна система. Системата е реализирана в болница и стереотипните групи представят служителите в тази болница, разделени въз основа на занятие, статус и квалификация. Хипервръзките съответстващи на всеки стереотип са предварително зададени и всеки потребител, принадлежащ към конкретен стереотип, получава достъп само до хипервръзките на своя стереотип. Индивидуалността на потребител се представя чрез хипервръзките, до които той има достъп.

Съществуват системи, в които потребителите се описват с помощта на набор от правила [57]. Stadnyk и Kass [92] предлагат метод за използване на стереотипи, създадени на базата на правила. Те установяват, че има правила, които са общи за много потребители и следователно, чрез тях може да се описват стереотипни групи. Те идентифицират пет категории, чрез които потребителите определят желанията и потребностите си: (1) област на интерес; (2) цели; (3) тип информация, от която се интересуват (като съобщения, статии и др.); (4) параметри на интересуващата ги информация (дължина на темата, източник на темата и др.) и (5) комбинации от горните категории. Те създават правила, в дясната страна на които стои решението на потребителя дали да прочете дадено произведение или не, а в лявата страна се задават параметрите на това произведение.

2.5 Адаптивно търсене – дефиниции и характеристики

Адаптивното търсене е свързано с изграждане на системи, способни да осигуряват индивидуална колекция от резултати за различни потребители. В основата на тези системи стои модел, описващ знанието за потребител. Адаптивното търсене е процес на селекция на най-подходящите резултати и тяхното класиране в зависимост от конкретен потребител, група от потребители или всички потребители на системата. То се реализира посредством набор от подходи, включващи: адаптиране на заявката, преподреждане на резултатите и оценяване на резултатите. Адаптивното търсене се базира на набор от методи, техники и стратегии, като: история на търсенето, потребителски модели, колаборативни подходи, клъстеризиране на резултатите, хипертекстово съдържание, текущ контекст [61].

Съществуват различни типове системи, в които адаптивното търсенето намира приложение, като те се различават в зависимост от областта на приложение, вида на услугата, която предоставят, и езика, на който функционират. Такива системи са средите за електронно обучение, мултимедийните дигитални библиотеки, системите за новини, търсещите машини (търсачки) и др. АПС за извличане на информация могат да бъдат класифицирани въз основа на различни признаци, сред които са: (1) тип на системата; (2) обхват на персонализацията и (3) подходи за осъществяване на персонализацията.

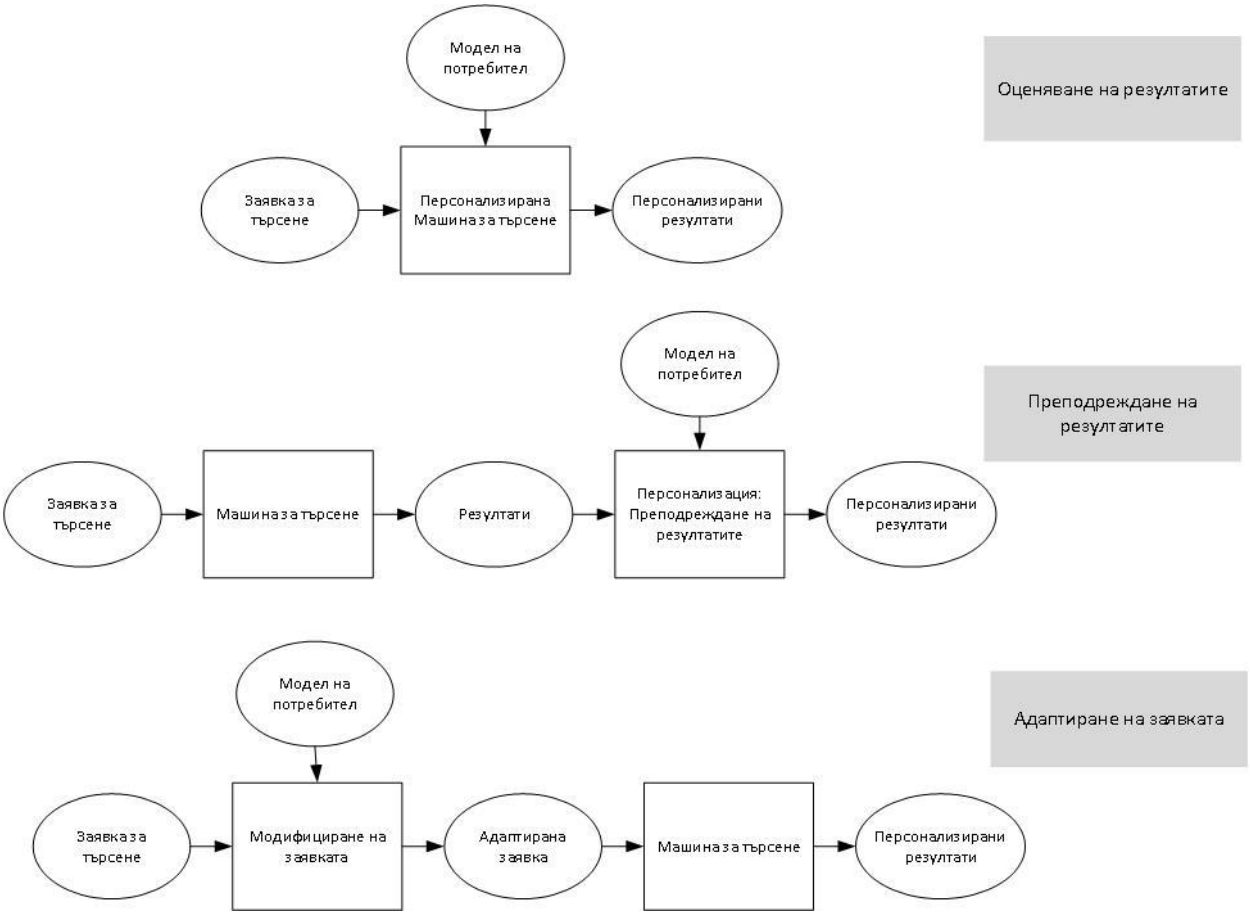
Адаптивността и персонализацията в системата може да има различен обхват и да бъде насочена към отделен потребител, към група от потребители или към всички потребители на системата. В първия случай адаптивността се реализира на базата на индивидуален (персонален) модел, във втория, с помощта на групов модел (стереотип) и в третия на основата на колективните данни, събрани от всички потребители при използване на системата.

Адаптивното търсене, основано на индивидуален модел, прилага индивидуални правила за адаптивност за всеки отделен потребител; базираното на групов модел прилага общи правила за адаптивност към цялата група потребители; а агрегатното – към всички потребители. Kruschwitz et al [54] предлагат метод за реализиране на адаптивно търсене в дигитална библиотека, който се базира на агрегатна персонализация. В тяхната система се събират данни от изпратените заявки от всички потребители и на тяхна база се реализира персонализация.

Micarelli et al [61] различават три подхода за адаптивно (персонализирано) търсене, според мястото на модела на потребителя в тях:

- Когато моделът участва в процеса на извличане на информация, т.нар. оценяване на резултати
- При преподреждане на резултатите, когато моделът участва при вторична обработка на извлечените резултати
- При адаптиране и модифициране на заявката с помощта на модела [61].

Фиг. 2 представя подходи за адаптивно и персонализирано търсене според мястото на модела в тях [61].



Фиг. 2 Подходи за адаптивно и персонализирано търсене според мястото на модела на потребител [61]

Главните методи за реализиране на адаптивно търсене са свързани с (1) адаптиране на заявката; (2) адаптиране на върнатите резултати или (3) с двете заедно. Процесите на адаптивност и персонализация в многоезичните системи са допълнени с превод на заявката и резултата между различни езици [68].

Адаптирането на заявката води до нова заявка, за която се приема, че представя по-пълно нуждите и интересите на потребител. Адаптиране на заявката може да се реализира още на първия етап на търсене или на по-късен етап, например когато потребителят не е удовлетворен от качеството на изведените резултати и зададе критерии за адаптиране. Главното предимство на този подход за персонализация е в това, че при него не се добавя допълнителна обработка на резултатите в сравнение с неперсонализираните подходи. При адаптиране на заявката потребителският модел влияе върху класирането на резултатите непряко, единствено посредством промяната на самата заявка.

Адаптирането на заявката решава редица проблеми, свързани с недостатъчно познаване на предметната област от потребителя. Простите заявки по ключови думи са полезни тогава, когато потребителите имат ясна представа какво търсят и знаят какви ключови думи да използват при търсенето. В средите, доставящи знания, често нивото на познаване на предметната област е ниско и се появява необходимост от трансформиране на оригиналната заявка на потребител в заявка, която отчита и използва установената терминология в областта.

Главните методи за адаптиране на заявката са: (1) модифициране на заявката, (2) разширяване на заявката с нови термини и (3) задаване на тежести на термините от заявката.

Модифицирането на заявката е свързано със замяна на оригиналните термини за търсене с други, за които се приема, че представят по-точно интересите на потребител. Koutrika и Ioannidis [53] предлагат метод, базиран на правила за модифициране на заявката за адаптивно търсене в система препоръчваща филми. В техния подход оригиналната заявка се заменя с множество заявки, като процесът е управляван от правила, базирани на индивидуалните предпочитания на потребител. Тези заявки се свързват дизюнктивно помежду си посредством логически оператор “OR”.

При разширяване на заявката се добавят нови термини към оригиналните термини от заявката. По този начин късите заявки, състоящи се от един термин, могат да бъдат допълнени с нови термини, които конкретизират търсенето.

Адаптиране на заявката може да се реализира както с термини, така и с понятия или категории от областта. Категориите служат за представяне на обобщено понятие. Liu и Meng [56] представят система за адаптивно търсене, в която заявката се адаптира на базата на категории. В тяхната система моделът на потребител представлява вектор от категории, базирани на йерархията ODP. Техния подход е базиран на модифициране на заявката посредством специфициране на категориите, с които оригиналните термини от заявката са свързани.

В най-общия случай термините и категориите за модифициране или разширяване на заявката се взимат от потребителски модел. В други случаи термините се взимат от референтни речници, таксономии или онтологии на областта. Заявката може да се разширява и посредством методите на „псевдо-уместна обратна връзка“ и

„уместна обратна връзка“ или като се използват данни, получени при интерактивна сесия с потребител [38]. Друг подход се базира на идентифициране на локалния контекст в заявката, което се реализира като се анализират документите, класирани най-високо в списъка с резултати. Тези документи се подлагат на обработка, в резултат на която се извеждат ключовите категории в тях. Изведените категории се използват за разширяване на заявката, с която се извършва повторно търсене [103].

При адаптиране на заявката чрез разширяване с нови термини, броят на тези термини се различава при отделните системи. Съществува подход, наречен „селективно разширяване на заявката“, при който броят термини, с които се разширява оригиналната заявка, се определя динамично в зависимост от самата заявка, като сред факторите са: дължина на заявката, еднозначност на заявката, обхват на заявката и др. [18].

Адаптиране на върнатите резултати от търсенето е процес на селекция на най-подходящите за потребител резултати и тяхното преподреждане. При този процес се цели поставянето на резултати, смятани за по-значими за даден потребител, на по-предно място в списъка с резултати от търсенето. По този начин класирането на резултатите е строго индивидуален за даден потребител и следва неговите предпочитания.

Главните методи за адаптиране на върнатите резултати са: (1) преподреждане на резултатите; (2) филтриране на резултатите и (3) оценяване на резултатите [61].

Преподреждането и филтрирането на резултатите е свързано с допълнителен процес на обработка над изведения в резултат на търсенето списък. Редица системи прилагат този подход като персонализация от страна на клиента, при което резултатите от заявката се анализират локално. Този анализ обикновено се прави само за ресурсите, класирани на челните места в списъка с резултати или се взимат предвид само фрагментите от текстовото описание на ресурсите. Този подход се смята за относително бавен, поради времето, нужно за връзка със съответната търсеща машина, която да извлече списък с документи. При филтриране на резултатите от първоначалния списък част от тях се елиминират, тъй като те се приемат за „неинтересни“ за потребителя. За разлика от филтрирането, при преподреждането на резултатите всички първоначално изведени резултати се представят, но редът на

тяхното класиране е променен. Целта на тази промяна е поставянето на „интересните“ резултати на по-предна позиция в списъка. Възможно е комбинирането на тези два метода, при което резултатите първо се филтрират и след това се преподреждат.

В системата Misearch [89] се реализира адаптивно търсене, като резултатите от търсенето се преподреждат. В тази система моделът на потребител представлява йерархия от категории с определена тежест, създаден на базата на референтната йерархия ODP. Моделът на потребител се използва в процеса на адаптиране на резултатите от търсене, като първоначално върнатите резултати се преподреждат според категориите в модела. Преподреждането на резултатите се реализира като се използват предишните заявки и се измери тяхното сходство с текущата заявка. Ако между тях съществува значителна степен на сходство то резултатите, получени от предишни заявки, се поставят на по-предно място в списъка с резултати от настоящата заявка.

Адаптивно и персонализирано търсене, посредством преподреждане на резултатите е реализирано и в проекта на Майкрософт Stuff I’ve Seen (SIS) [27]. Процесът на персонализация се извършва чрез преподреждане на резултатите, върнати от търсеща машина на базата на прост потребителски модел, съставен от термините в разгледаните от потребител документи. Моделът на потребител се обновява автоматично, в съответствие с разгледаните от потребителя документи.

Друг метод за адаптиране е т.нар. оценяване на резултатите, при който процесът на персонализация се осъществява едновременно с процеса на извличане на информация и поради това има най-голяма вероятност да осигури бърз резултат в сравнение с останалите методи.

Подходите, които използват историята на търсене, могат да се разделят на две групи – офлайн и онлайн подходи. Офлайн подходите използват историята в отделна стъпка, при което се налага да се извърши допълнителна обработка на данните. Те се базират на анализ на взаимовръзките между заявките и документите, видени от потребител. За разлика от тях онлайн подходите събират данните от взаимодействието на потребител със системата още в момента на тяхното възникване, като по този начин дават мигновено отражение върху модела на потребителя и оттам на процеса на персонализация. Поради тези особености на двата подхода използваните алгоритми се различават. В общия случай офлайн подходите използват по-сложни

алгоритми, тъй като те обикновено имат по-малко ограничение върху времето на изпълнение отколкото онлайн подходите.

Адаптивното търсене в дигитална библиотека в основата си следва методите, подходите и средствата приложени в останалите типове системи. Поради това при него намират приложение разгледаните подходи - адаптиране на заявката, адаптиране на върнатите резултати и оценяване на резултатите. Адаптивното търсене в дигитална библиотека е насочено към разработване на адаптивни стратегии за откриване на подходящо съдържание за даден потребител. Поради това е широка употребата на персонализационни подходи базирани на съдържание (content-based). Друга група от методи и подходи обхващат колаборативни подходи (collaborative-based) за клъстеризиране резултатите с цел даване на препоръки на потребител.

Съществуват адаптивни стратегии в дигитална библиотека, които се базират на използването на цитати (citation-based) за откриване на връзки между ресурсите в една колекция. Kessler [49] представя изследване за свързаността между научни статии на базата на общи референции в тях. Съществена характеристика на научните ресурси е тяхната библиографска свързаност. Това означава, че ако един научен ресурс се появява в библиографската справка на друг научен ресурс, то между тях е възможно да съществува тематична свързаност. Тази свързаност може да се използва за откриване на сходни ресурси, които да бъдат препоръчани на потребител. Kessler [49] приема, че една обща референция между два документа е една еденица свързаност между тях. Силата на свързаност между два ресурса се измерва чрез броя такива единици свързаност. Свързаността между ресурсите може да бъде пряка, когато един ресурс цитира изрично друг ресурс, а може да бъде и косвена, както е в случаите, когато два ресурса съдържат еднакви референции към трети ресурс. За своето изследване Kessler [49] използва данните от 137000 референции от 8521 статии, публикувани в 36 тома на списание Physical Review. Въпреки, че този подход крие възможности за задаване на неподходящи препоръки, тъй като невинаги референциите са гаранция за тематичната близост между ресурсите, към момента той е признат за ефективна техника за идентифициране на свързани статии.

Изводи и заключения:

В настоящата глава е направен преглед на езиците, стандартите и протоколите на Семантичния уеб. Проследено е актуалното състояние и са очертани развойните тенденции. Направено е аналитично изследване на методите, подходите и стратегиите за реализиране на адаптивност и персонализация, като са проследени средства, стандарти и добри практики. Изследвани са подходите, методите и стратегиите за реализиране на адаптивно търсене. Анализирани са примерни системи, чиито опит и разработки са от съществено значение за настоящия труд. Направено е аналитично изследване на модел на потребител, като са разгледани главните типове модели и са идентифицирани техните характерни особености. Изследвани са съществени характеристики на стереотипните модели, като са разгледани: подходите за определяне на стереотипните групи; методите за класифициране на потребител към стереотип; структурата на стереотипите в системата и др. Разгледани са съществуващи системи, в които адаптивността е базирана на употребата на стереотипи. Направен е сравнителен анализ на шест такива системи, които са съпоставени въз основа на техни съществени признаци. Получените резултати са анализирани и обобщени. Аналитично са изследвани методите, подходите и средствата за реализиране на адаптивно търсене. Разгледана е ролята на модела на потребител в процеса на адаптивно търсене. Изследвани са адаптивни и персонализационни стратегии и подходи в системи за извличане на информация и в дигитална библиотека.

Глава 3. БЕЙСОВИ (ВЕРОЯТНОСТНИ) МРЕЖИ. МЕТОДИ, ПОДХОДИ И СТРАТЕГИИ ЗА ИЗГРАЖДАНЕ НА БЕЙСОВИ МОДЕЛИ

Бейсовите мрежи са способни да изразят отношение между състояние и неговите симптоми, или да предскажат най-вероятния изход на ситуация или събитие. Те намират приложение в много области, като изчислителна биология, био-информатика, документна класификация, обработка на изображения, медицинска диагностика и в системи, подпомагащи вземането на решения. Широкото приложение на вероятностните мрежи е свързано с някои техни предимства, като възможността да се изрази факта, че твърденията си влияят взаимно, разпространение на промяната в мрежата при постъпване на нова информация, намалени изисквания за изчисляване на пълните вероятности, наличие на консистентност и др. Техен общ недостатък е, че не винаги е възможно да се различи несигурността от незнанието.

Една вероятностна мрежа може да се разглежда като база от знания, представени във вид на правила от съждителното смятане, която е допълнена с локални таблици на Съвместно вероятностно разпределение (СВР). Оттук следва, че всяка база знания, основана на съждителното смятане, може да се представи чрез вероятностна мрежа. Обратното не е вярно – вероятностните мрежи са много по-богат и прецизен модел за представяне на знания, отколкото съждителното смятане [104].

Теорема на Бейс

Теоремата на Бейс се използва в теорията на вероятностите за изчисляване на вероятността за настъпване на дадено събитие, след като вече е известна част от информацията за него [104].

Теоремата на Бейс гласи, че:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

където:

P(A) е вероятност за настъпване на събитието A;

P(A|B) е условна вероятност за настъпване на събитието A при положение, че събитието B е настъпило (апостериорна вероятност);

P(B|A) е условна вероятност за настъпване на B при положение, че A е настъпило;

P(B) е вероятност за настъпване на събитието B.

Съвместно вероятностно разпределение (СВР)

СВР е функция, която съпоставя на всяко атомарно събитие някаква вероятност. СВР може да бъде представена чрез таблица от четири стойности или посредством алгебричен израз. В определен момент само едно атомарно събитие може да бъде вярно за дадено множество от променливи. Сумата от вероятностите на всички атомарни събития е 1. Вероятността от две събития случващи се по едно и също време (X;Y) ще бъде: P(X,Y), изразявана също като $P(X \wedge Y)$. Размерът на СВР таблиците нараства експоненциално на броя предшественици на променливата.

Условната вероятност P(X |Y) се дефинира като:

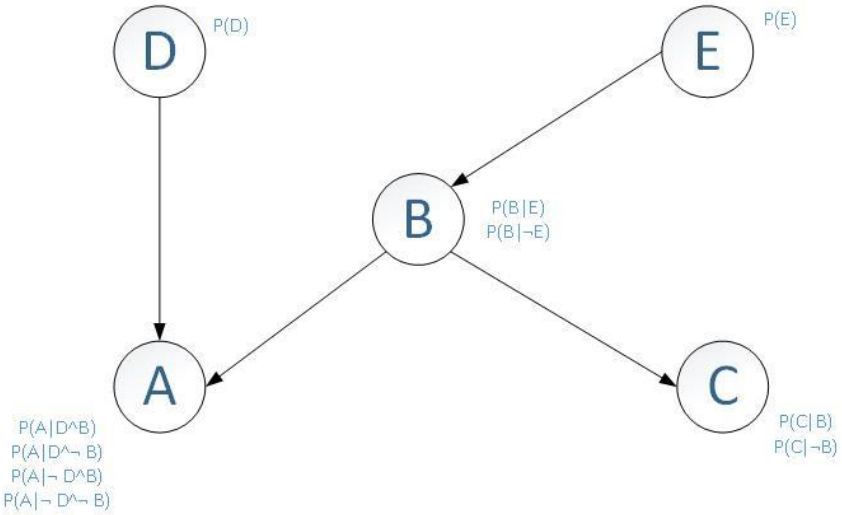
$$P(X|Y) = \frac{P(X \wedge Y)}{P(Y)}$$

Бейсова мрежа – дефиниции и определения

По-гъвкав и икономичен метод за вероятностно съждение е вероятностната мрежа, който използва предметната област за опростяване на логическия извод. Едно от най-удобните представяния на условните вероятности е във вид на вероятностна мрежа на Бейс.

БМ се дефинира като насочен ацикличен граф, представен от наредената двойка (V,E), където V е множество от възли, а E е множество от дъги, съединяващи възлите. Всеки възел от V съдържа име на променлива и таблица на условно СВР, която показва зависимостта на тази променлива от нейните предшественици.

Съществуват дъги само между възли, съответстващи на променливи, които не са условно независими [104].



Фиг.3 Вероятностна мрежа

В показаната на Фиг.3 вероятностна мрежа независимите променливи са две: D и E. Те са първопричината за останалите три събития, представени чрез променливите: A, B и C.

Според дефиницията на Pearl [74] БМ е насочен ацикличен граф, в който всеки възел представя случайна променлива и всяка дъга представя насочена вероятностна зависимост между две променливи. Ако съществува дъга между два възела, това означава, че има зависимост между съответните две променливи, което би било изразено чрез подходяща импликация в съждителното смятане.

БМ може да се разглежда като база от знания, представена във вид на правила от съждителното смятане, допълнена с таблици на СВР. Поради това необходимо условие за създаване на БМ е задаване на определени правила и таблици на СВР към всяко правило. Тази таблица указва каква е вероятността да е вярно следствието на правилото за всички възможни комбинации от истинностни стойности на предпоставките на правилото. Вероятностното съждение съдържа в себе си съждителното смятане, но не и предикатното.

Бейсовите мрежи са разработени през осемдесетте години от Pearl [73] и оттогава се наблюдава силен интерес към тях и съществен напредък в развитието на

нови техники и алгоритми за тяхното приложение. БМ е вероятностен модел с причинно-следствена връзка, който осигурява графично представяне, при което всеки възел представлява променлива и всяка дъга представлява причинно-следствено влияние. В момента те се считат за една от най-добрите съществуващи техники за решаването на диагностични и класификационни проблеми. Независимо от това не са много системите, в които се използват техниките на приблизителен извод, основан на БМ или Размита логика.

БМ се състои от качествена и количествена част. Качествената част е насоченият ацикличен граф, отразяващ причинно-следствената структура на областта. Количествената част представя съвместното разпределение на вероятностите над променливите. Всяка променлива притежава таблица на СВР, съдържаща вероятностите на всяко от състоянията в зависимост от предшестващи променливи. Ако променливата няма предшестващи променливи в графа, тогава СВР представя първоначалното вероятностно разпределение на променливата.

Вероятностно съждение основано на СВР

Вероятностното съждение винаги може да се сведе до оценка на условни вероятности. Вероятностната картина на проблемната област за един интелигентен агент може да се разглежда като множество от случайни булеви променливи, приемащи определени стойности с някаква вероятност. Верността на част от тези променливи, наречени входни, се въвежда от сензорната система (или от човек оператор), като целта на провежданите от агента разсъждения е да се оценят вероятностите на останалите променливи, наречени изходни.

Подходът към вероятностно съждение, основан на СВР не използва структурата на предметната област и взаимовръзките между съжденията в нея. Този модел на вероятностно съждение е пълен и достоверен и може да изчисли произволна условна вероятност. Но той е практически неприложим, поради експоненциално нарастващият размер на таблицата на СВР.

Този недостатък на вероятностното съждение се избягва при БМ, където таблицата на СВР на даден възел се дефинира само въз основа на неговите предшественици. Оттука идва икономията на памет – вместо да се съхранява една

огромна таблица на СВР за всички променливи, всеки възел съхранява собствена малка таблица на СВР, която е дефинирана само върху възлите, от които той е зависим [104].

БМ кодира СВР на набор от краен брой променливи $\{X_1,...,X_n\}$, и го разлага в произведение на условно вероятностно разпределение на всяка променлива при зададени предшественици. В случай на възли без предшественици, се използват априорни вероятности. СВР на $\{X_1, X_2,\ldots,X_n\}$ се получава като се вземе произведението на всички априорни и условни вероятностни разпределения [74]

$$p(X_1,...,X_n) = \prod_{i=1}^{n} p(X_i|\ P(X_i))$$

Вероятностно съждение в Бейсова мрежа

Бейсовото съждение (БС) се различава от статистическото съждение, което извежда заключения на базата на статистически извадки. Този вид съждение използва предварителната оценка на степента на доверие в хипотезата, която в повечето случаи е изведена от статистически данни. Априорната вероятност се използва за оценяване на дадена хипотеза, преди да има налични доказателства. При БС могат да се използват доказателства за обновяване на изчислената преди това вероятност. Наличните доказателства могат драстично да променят степента на доверие в хипотезата.

Основните типове съждение в БМ са диагностично съждение, прогностично съждение, както и съждение от смесен тип.

Използването на модела на БМ позволява да се разсъждава както в диагностична посока (какви са най-вероятните причини при дадените доказателства), така и в прогностична посока (каква е вероятността, че дадена конфигурация от състояния на променливи ще настъпи, при даден набор от доказателства).

Прогностичното съждение е свързано с извеждане на ефектите от причините – при наличие на наблюдаван феномен може да се предположи какви ще са очакваните ефекти от него.

Диагностичното съждение е свързано с извеждане на причините за дадено състояние от наличните факти. Този тип съждение изисква много повече изчислителна сила, в сравнение с прогностичното съждение.

БМ е способна да изчисли постериорните вероятности на несигурни променливи въз основа на доказателства, получени от свързани към тях променливи. Този процес е известен като разпространение на доказателства или обновяване на вероятностите. Това свойство, наред със способността на БМ да моделира комплексни отношения между несигурни променливи я правят много подходяща за изграждане на диагностични модели.

Вероятностното съждение притежава ред предимства пред съждителното смятане:

- Създателят на една вероятностна мрежа може да заложи предпочитания за верността на едни или други съждения във формата на априорни вероятности.
- Правилата може да се настройват прецизно с помощта на таблиците на СВР, така че да отразяват максимално добре причинно-следствените връзки, които моделират.
- Вероятностното съждение не прави допускането за затворен свят, както системите, основани на формална логика. Ако едно твърдение не може да се докаже, че е вярно, системата не приема непременно, че то е невярно [104].

Типове алгоритми за вероятностно съждение

Съществуват два главни типа от *алгоритми за вероятностно съждение* – точни алгоритми и приблизителни алгоритми. И точният, и приблизителният вероятностен извод в БМ е NP труден, поради това при по-големи мрежи в редица случаи се използват модели с ограничена изразителност. За приложението на големи БМ при задачи от практически интерес, са необходими ефективни алгоритми за изчисляване на условни вероятности. Точните алгоритми използват особеностите на графа на вероятностната мрежа, за да изчисляват направо търсените условни вероятности, без да преминават през таблицата на СВР. Приблизителните алгоритми се основават на идеята за стохастична симулация на вероятностното съждение.

Точните алгоритми са структурно-базирани и по този начин експоненциално зависими от т.нар параметър „широчина на дървото“. Този параметър измерва сходството на графа с дървовидна структура [22].

- *Алгоритъмът за клъстеризиране* е най-бързият известен точен алгоритъм за обновяване на Бейсова мрежа. Той е разработен от Lauritzen и Spiegelhalter [55] и впоследствие подобрен от други изследователи, като Jensen и Andersen [47]. Алгоритъмът функционира на два етапа: (1) компилиране на насочения граф в свързано дърво, и (2) вероятностно обновяване на свързаното дърво.

- *Алгоритъмът с поли-дървета* е представен в [74]. Поли-дърво е всеки ацикличен граф, в който между всеки два възела съществува не повече от един път. Изискването графът да бъде поли-дърво ограничава знанията, които може да се представят с мрежата. Това е единственият алгоритъм за обновяване, който е с линейна сложност.

Приблизителните алгоритми редуцират проблема със съждението до ограничен оптимизационен проблем и обикновено са независими от структурата на мрежата. Тази категория от алгоритми изчислява не точната стойност на търсената условна вероятност, а нейното приближение въз основа на компютърна симулация. Сред тези алгоритми са: Probabilistic logic sampling, Likelihood sampling, Self importance sampling и др [22].

3.1 Модели за представяне на причинна независимост

При моделиране на ситуации от реалния свят, често се срещат положения, при които множество причини, предизвикват един и същи ефект [40]. Този вид взаимодействие може да се моделира, посредством вероятностни мрежи. Това взаимодействие не отразява причинна независимост между променливите на причините. За преодоляване на тази непълнота Henrion [41, 42] предлага модел, в който освен променливите на ефекта Y и на причините $\{X_1, X_2, \ldots, X_n\}$ се използват междинни променливи $\{I_1, I_2, \ldots, I_n\}$, представящи независимия принос на всяка от причините върху ефекта. По този начин ефектът Y се представя като детерминистична функция от тези междинни променливи. За всяка причина X_i се обозначава инстанция от съответстващата и променлива, която да бъде изтъкната. Тази инстанция представлява състояние на причината X_i, в която тя няма влияние върху ефекта Y или обозначава "изключено" състояние.

3.1.1 Взаимодействието NoisyOR

В БМ всяка променлива съдържа таблица на СВР, представяща вероятността за всяко състояние при дадено състояние на предшестващата променлива. Ако променливата не притежава предшестваща променлива, нейната СВР таблица представя разпределението на априорната вероятност на променливата. Размера на една СВР таблица нараства експоненциално на броя предшестващи променливи на дадената променлива. Поради това са разработени модели на канонично взаимодействие, които апроксимират СВР таблиците и изискват по-малко параметри. Тези модели на канонично взаимодействие са известни като шлюзове (gates). Комплексността на съждението се избягва, като се използват канонични взаимодействия като NoisyOR, NoisyMAX и NoisyAdder, които ограничават изразителната сила на модела [74].

Най-широко използвания модел за изразяване на причинна независимост е NoisyOR шлюза. Този модел се състои от бинарни променливи на причините $\{X_1, X_2, \ldots, X_n\}$, междинни променливи $\{I_1, I_2 \ldots, I_n\}$ и променлива на ефекта Y, като всяка променлива притежава единствено стойностите *истина* или *лъжа*. При това е вярно следното:

$$f(I_1, \ldots, I_n) = I_1 \vee \ldots \vee I_n$$

както и:

$$p(I_j = \text{true} \mid X_j = \text{false}) = 0 \quad \text{за } I = 1, \ldots, n$$

Този шлюз е представен за първи път от Good [39]. По-късно той се прилага в контекста на БМ от Pearl [73]. Шлюзът NoisyOR става много популярен в БМ, защото той редуцира размера на СВР на дадена променлива от експоненциален до линеен на броя предшестващи променливи. Шлюзът NoisyOR може да се използва само при бинарни променливи, като за променливи, приемащи множество стойности е разработен шлюза NoisyMAX от Henrion [42] и Diez [24].

Шлюзът NoisyOR е член на фамилия от модели, за които е характерно, че влиянието на всяка от моделираните причини X_i, върху ефекта Y е независимо от другите причини. Думата Noisy означава, че причинното взаимодействие не е детерминистично, в смисъл, че всяка причина може да предизвика ефекта с някаква вероятност. Думата OR обозначава логическо OR взаимодействие между независимите причини при продуциране на ефекта. Моделът NoisyOR приема, че наличието на коя да

е от причините X_i е достатъчна за продуциране на ефекта Y и, че тази способност за продуциране на ефекта е независима от другите причини. От друга страна, наличието на причина X_i, не е гаранция, че ефектът Y ще се случи. Това се изразява, чрез:

$$p_i = P\ (y \mid \bar{x}_1, \bar{x}_2, \ldots, x_i, \bar{x}_{n-1}, \bar{x}_n)$$

където p_i е вероятността на причината X_i

NoisyOR се прилага в случаите, когато има няколко възможни логически причини $\{X_1, X_{2,\ldots}, X_n\}$ за бинарната променлива на ефекта Y, при което:

- всяка от тях притежава вероятност p_i да бъде достатъчна, за да се получи ефект в отсъствие на всички други причини
- вероятността за всяка кауза е достатъчна, независимо от наличието на други причини

Пълната условна вероятност за n бинарни предшестващи възела в традиционния случай би изисквала определянето на параметъра, но ако се използва NoisyOR техният брой се редуцира до n, по един за всяка вероятност p_i.

NoisyOR моделира недетерминистичното взаимодействие между бинарна предшестваща променлива X_i и бинарната променлива Y, която представя ефекта. Всяка променлива има две състояния: изтъкнато състояние, което представя нормалното, работно състояние на променливата (обикновено това е *лъжа* или отсъствие) и не-изтъкнато състояние (*истина* или наличие).

Променливата Y, която представя ефекта, функционира като детерминистичен NoisyOR. Това означава, че ако всички предшестващи променливи X_i са неналични, то и променливата Y ще бъде неналична. Ако предшестващата променлива X_i е налична, а всички останали предшестващи променливи са неналични, вероятността да се случи ефекта Y ще бъде p_i.

Вероятността ефектът Y да се случи при наличие на множеството от причини X, които са налични, се задава с формулата [74]:

$$P(y|X) = 1 - \prod_{i=1}^{n} (1 - p_i)$$

Тази формула е достатъчна за извеждане на пълната CBP таблица за ефекта Y, зависещ от неговите предшественици $\{X_1, X_{2,\ldots}, X_n\}$.

Henrion [42] предлага разширение на NoisyOR за моделиране на случаите, в които ефектът Y може да се случи, при условие, че всички възможни причини са отсъстващи. Това разширение се нарича LEAK и представя феномена, че ефекта може да възникне спонтанно, при отсъствие на извънредна причина. LEAK NoisyOR се моделира чрез допълнителен параметър p_0, наречен LEAK вероятност като:

$$p_0 = P\ (y \mid \bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n)$$

3.1.2 Взаимодействието NoisyAdder

Друг модел, представящ взаимодействието причина-следствие е NoisyAdder. Този модел се състои от бинарни променливи $\{X_1, X_{2,\ldots}, X_n\}$ и целочислени променливи $\{I_1, I_2 \ldots, I_n\}$, като всяко I_j може да приема стойности от $-l$ до $+l$. Освен това е в сила:

$$f(I_1, \ldots, I_n) = I_1 + \ldots + I_n$$

както и:

$$p(I_j = 0 | X_j = \text{false}) = 1 \quad \text{за I} = 1, \ldots, n$$

Dagum и Galper [21] дефинират т.нар. адитивни вероятностни модели (ABNM), които се използват за апроксимиране на вероятностни мрежи. Най-общо за адитивните модели е характерно, че те се дефинират чрез:

$$P(Y \mid X_1, X_{2, \ldots} X_n) = \sum_{i=1}^{n} f_i\ (X)$$

където: f_i е произволна функция.

Dagum и Galper [21] дефинират ABNM модела като удовлетворяващ:

$$P(Y \mid X_1, X_{2, \ldots} X_n) = \sum_{i=1}^{n} \alpha_i P(Y|X_i)$$

Те предлагат метод за изчисляване на условните вероятности при ABNM модели с техники за приближение на параметрите.

Radde et al [82] представят метод за изчисляване на условните вероятности, базиран на NoisyAdder.Те изчисляват условното вероятностно разпределение за даден

възел от мрежата като средната тежест на предшестващи възли. За тази цел те сумират теглата на всички входящи дъги към възела и разделят на сумата от всички тежести.

3.2 Бейсови модели – структура и параметризация. Сравнителен анализ на модели с канонично взаимодействие

Съществуват Бейсови модели с различна степен на сложност. При диагностичните модели, най-простият модел е базиран на предположението, че само една повреда може да се случи в едно и също време, и че наблюденията са условно независими. Структурата на този модел е такава, че съществува само една променлива за повреда, която има различни състояния за всяка отделна повреда, и една или повече променливи за наблюдение, като нейни деца. Тази структура е известна като *Елементарен (Naive) Бейсов модел* [33]. Главното предимство на този модел е, че броят на условните зависимости в него е малък, и поради това е изчислително изгоден.

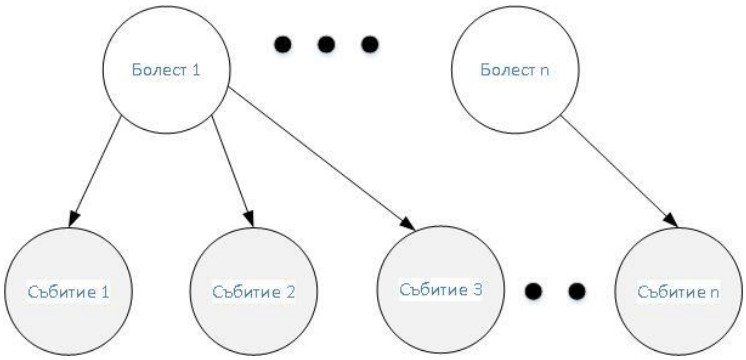
По-реалистични Бейсови модели са тези, в които има повече от една променлива за представяне на повреди. Този тип модели е известен като *Модел на множество повреди*. Недостатък на този тип модел е съществото нарастване на броя на условните вероятности, тъй като СВР на променливите за наблюдение зависи експоненциално от броя на предшестващите променливи. Този модел е базиран на предположението за липса на релации между променливите, представящи повреда, и между променливите, представящи наблюдение.

За моделиране на релациите между променливи от един и същи тип (повреда или наблюдение) е създаден *Разширен модел на множество повреди*. Този модел се характеризира с по-голяма прецизност за сметка на нарастване на броя на параметрите. При много променливи този модел става изчислително неизгоден. За ускоряване на изчислителния процес са разработени канонични модели, които апроксимират СВР и изискват по-малко параметри. Следва представяне на модели, които използват канонични взаимодействия.

Моделът *QMR-DT* [87] е двуслоен каноничен NoisyOR модел, който се състои от два типа бинарни случайни променливи, представящи болести и събития. Променливите на събитията представят симптоми на болести, резултати от тестове, демографски данни и др. Моделът *QMR-DT* съдържа приблизително 5000 променливи,

от които 570 са болести, а останалите представят събития, свързани с тези болести. Събитията, които не са бинарни (могат да притежават повече от две стойности), се представят с набор от бинарни променливи. Променливите на болестите влияят върху променливите на събитията, с които са свързани чрез дъги. Променливите от всеки слой са независими помежду си и между тях няма дъги.

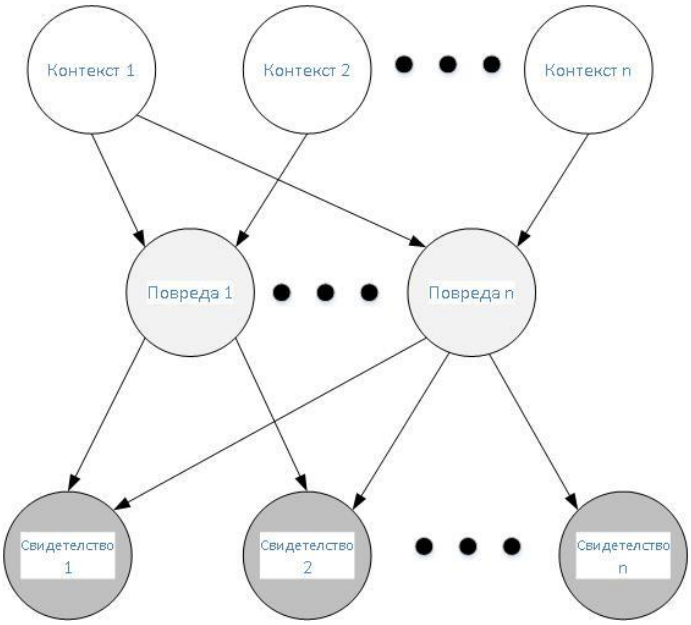
Бейсови модели имащи същия тип структура и параметризация като модела *QMR-DT* се наричат бинарни двуслойни NoisyOR (BN2O) мрежи. Фиг. 4 представя структурата на модела *QMR-DT* [87].



Фиг.4 Структура на модела QMR-DT [87]

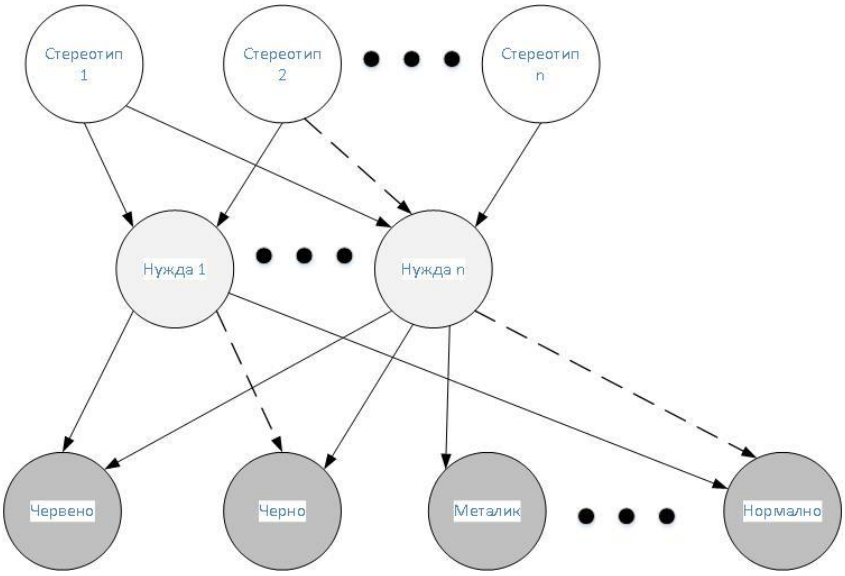
Възможно е към BN2O мрежите да се добави трети слой от променливи за моделиране на междинен слой от отношения. Съществуват модели, използващи канонични взаимодействия, които са изградени на този принцип, като тези в [51, 82].

Kraijeveld et al [51] представят каноничния NoisyMAX модел *BN3M*, който е трислойна мрежа на Бейс, като всеки слой се заема от различен тип променливи. Моделът *BN3M* е изграден на предположението, че променливите от всеки слой са независими помежду си, и поради това не са свързани с дъги. Променливите, използвани в модела могат да приемат множество от стойности и по това се различават от бинарните променливи в *QMR-DT* модела. Фиг.5 представя структурата на модела *BN3M* [51].



Фиг.5 Структура на модела BN3M [51]

Radde et al [76] представя каноничен NoisyOR модел, който е трислойна мрежа на Бейс, като всеки слой се заема от различен тип бинарни случайни променливи. Техния модел е подобен на *BN3M* модела и е сходен с *QMR-DT* модела. В него, подобно на другите два модела, променливите от всеки слой са независими помежду си, и поради това не са свързани с дъги. В него подобно на модела *QMR-DT* моделирането на факт, който може да приема повече от две стойности се реализира посредством набор от бинарни променливи. Например вместо да се създаде един възел за цвят, които може да има различни стойности, всеки отделен цвят се представя с отделна бинарна променлива. В този модел дъгите притежават „тип“, който е положителен за представяне на позитивните влияния и отрицателен, за представяне на негативните влияния. Положителния тип дъги в модела се представят с непрекъснати линии, а отрицателните – с пунктирани линии. Фиг.6 представя структура на модела на Radde et al [76].



Фиг.6 Структура на модела на Radde et al [76]

В Таблица 3 е представено сравнение на каноничните модели на базата на техни съществени характеристики.

Таблица 3 Сравнение между канонични Бейсови модели

Модел	Тип Променливи	Слоеве	Посока на дъгите	Метод за редуциране на CBP	Тип Извод/Съждение
QMR-DT [87]	Булеви	Двуслойна мрежа	Каузална	NoisyOR	диагностичен
BN3M Модел на Kraaijeveld et al [51]	Множество стойности	Трислойна мрежа	Каузална	NoisyMAX	диагностичен
Модел на Radde et al [76]	Булеви	Трислойна мрежа	Каузална	NoisyOR	прогностичен

Теоретично няма ограничение за добавяне на още слоеве от променливи, например четвърти слой и нагоре, но Provan [75] твърди, че това не би довело до подобряване на съждителните възможности на модела. Освен това всеки допълнителен слой от променливи води до нарастване на сложността на модела, което не винаги е оправдано.

3.3 Използване на Бейсови модели в САПТ

Информацията за потребител в системата може да бъде несигурна или неточна, което налага търсенето на методи и способи, способни да се справят с тази несигурност. Главните технологии, които се използват за извод при несигурност, са Бейсова мрежа и Размита логика. Бейсовата мрежа позволява да се разсъждава както в диагностична посока, така и в прогностична посока. Тази способност на БМ за извод в двете посоки е много полезна при моделиране на знанието за обучаем. Тя не само дава възможност за точно диагностициране на актуалното познавателно ниво на обучаемия, но и позволява да се направи информирана препоръка за най-подходящата следващата дейност (например упражнения, тестове) според наличните доказателства за конкретния обучаем (предишни дейности в системата, любима стратегия на обучение и т.н.). Технологиите, използвани в Адаптивните обучителни системи, в редица случаи изпреварват тези в други типове системи. Изводите над знанието за потребител чрез Бейсови модели е най-разпространено в Адаптивните обучителните системи. Бейсовите мрежи са вероятностни графични модели, в които всеки възел представлява случайна променлива и всяка релация изразява причинно-следствено влияние. Те са смятани за една от най-добрите техники за решаване на диагностични и класификационни проблеми [12].

Системата KBS Hyperbook [44] е Адаптивна хипермедийна система, която насочва индивидуално обучаемите през информационното пространство, като им показва следващите стъпки от учебния процес, подбира подходящи проекти, генерира и предлага поредици за четене и полезна информация. Адаптивността в системата се базира на модел, използващ Бейсова мрежа за извод и оценяване на знанието на потребител. Системата класифицира потребителите в зависимост от тяхното ниво на знания в четири категории: новак, начинаещ, средно ниво и напреднал. Стратегиите за адаптиране в системата се основават на т.нар „единици знание“, които обозначават елементарни понятия в областта на приложение. Моделът на потребител в системата съдържа същите „единици знание“, които се съдържат в общия модел на системата. В Бейсовата мрежа „единиците знание“ се представят като възли, а зависимостите между тях се изразяват посредством условни вероятности, които изразяват силата на зависимостта. Бейсовата мрежа позволява използването на наблюдения за работата на потребител със системата и актуализиране на знанието за потребител. За справяне с

несигурността на наблюденията Henze и Nejdl [44] използват вектор от четири вероятностни стойности, които оценяват степента на разбиране на потребителя на дадена „единица знание“. Техният модел се представя като мрежа, съставена от три слоя от възли. На най-долния слой се намират елементарните понятия, които не изискват предварително знание. Средният слой съдържа по-сложни понятия, които се нуждаят от познания на понятията от долното ниво, за да бъдат разбрани. На най-горния слой се разполагат комбинираните понятия, които могат да бъдат разбрани, само когато са разбрани понятията от долните два слоя. Системата функционира, като сравнява актуалното знание на потребител с нужното за разбиране на дадена тема знание. Когато част от нужното знание липсва на потребителя, то се предоставя от системата, която генерира поредица от информационни единици, които водят потребителя стъпка-по-стъпка към тяхното научаване. Освен това, системата осигурява директен достъп до информационните ресурси, нужни за дадена задача.

Моделът на потребител в системата I-PETER [77] е подобен на този в KBS Hyperbook. Системата I-PETER е обучителна система за научаване на английски език. В тази система потребителят също бива класифициран според нивото си на знание в категориите: новак, начинаещ, средно ниво и напреднал. За разлика от KBS Hyperbook, в която потребителят се класифицира само към една от категориите, в системата I-Peter потребителят може да принадлежи към повече от една категория едновременно, като степента, в която принадлежи към дадена категория се представя чрез вероятностна стойност.

Системата CAPIT [59] обучава 8-10 годишни ученици на правилата на пунктуацията и главните букви в английския език. Системата използва Бейсова мрежа за моделране на обучаемия и теория на решенията за избор на обучителни действия. Съществено за системата е извеждането и адаптирането на модела на обучаемия, базиран на БМ на базата на данните за постиженията на обучаемия. За момента системата CAPIT съдържа 45 проблема за разрешаване от обучаемите и 25 ограничения [59]. Проблемите в системата се представят на обучаемия като текст, в който липсва пунктуация и всички думи са с малки букви. Ограниченията в системата покриват следните области: главна буква в изреченията, главна буква при лични имена и наименования на местности, използване на апострофи, завършване на изречението с точка и др. Модела на обучаемия се състои от запис на резултата от предишния опит по отношение на всяко ограничение (кратко-срочен модел) и моментна конфигурация

на Бейсовия модел (дълго-срочен модел). Възлите в модела, представящи резултата от последния опит на обучаемия могат да приемат стойности: (1) удовлетворен, (2) нарушен, (3) нарушен с обратна връзка и (4) не е приложимо. Моделът също включва възли на решения, което го превръща в Мрежа на решенията.

Съществуват Бейсови модели, използвани за моделиране на обучаем в адаптивни обучителни системи, които се основават на т.нар „погрешни схващания“. Stacey et al [91] разработват модел, който има няколко типа възли: (1) доказателствени възли, които могат да бъдат: тестови или поведенчески и (2) възли от тип обучаем, които служат за класифициране на обучаемите в зависимост от вида на погрешно схващане, което имат. Тази класификация се реализира на две нива на детайлност: груба и фина. Моделът разполага с няколко групи поведенчески възли, като групите могат да бъдат с различни размери и различна вътрешна структура (свързани по различен начин). Те служат за моделиране на различни обучителни епизоди. Дъгите в модела имат каузална посока.

В последните години съществува тенденция за разширяване на обхвата от потребителски характеристики, които могат да се измерят и използват в адаптивните системи. Съществуват системи, които използват Бейсови мрежи за моделиране на разнородни потребителски характеристики като: мета-когнитивни умения [66], личностни черти [35], емоционални състояния [19], нагласи и възприятия [3] и др.

Съществуват различни подходи за изграждане на Бейсова мрежа за моделиране на потребител, като някои от тях се опират изцяло на знанията на експерти в областта [76], докато други използват едновременно експерти и автоматизирани методи, както е в системите Andes [20], HYDRIVE [63], Adele [34] и DT-Tutor [67].

Изводи и заключения:

В настоящата глава са разгледани съществени теоретични дефиниции в областта на вероятностните модели и вероятностното съждение. Изложените дефиниции служат за създаване на теоретична основа, върху която е изграден представения в настоящата работа Бейсов модел за адаптивност и персонализация. Представени са характерните черти на Бейсовото съждение, като то е съпоставено с други типове съждение, като съждителното смятане, и са показани неговите предимства. Разгледани са главните типове алгоритми за вероятностно съждение и са

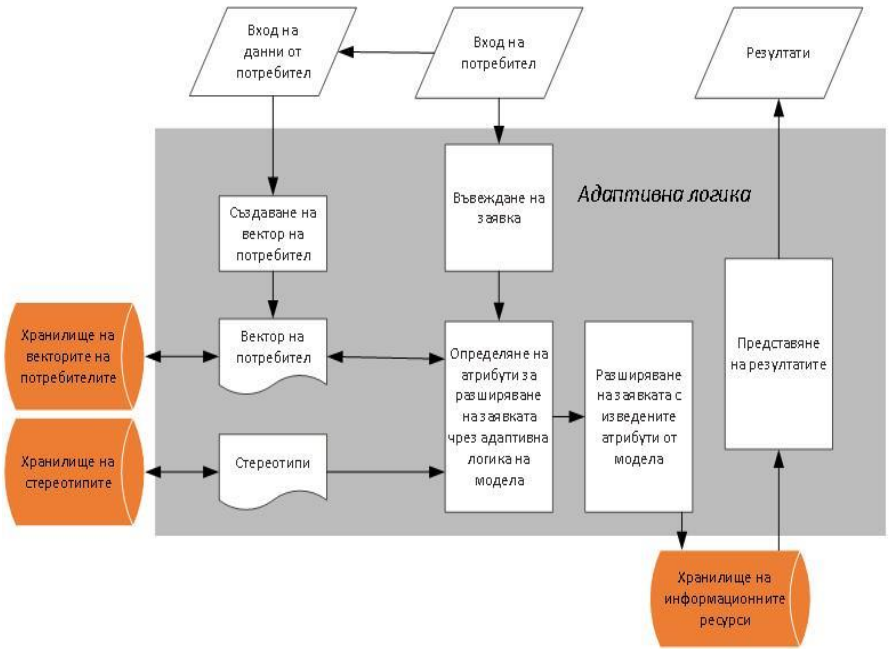
очертани техните предимства и област на приложение. Научно е изследвано каноничното взаимодействие NoisyOR и са анализирани Бейсови модели, използващи Noisy-OR. Направен е сравнителен анализ на съществени канонични модели, като са изследвани техните прилики и различия. Изследвано е приложението на Бейсови модели в редица системи, като фокусът е върху Адаптивни обучителни системи, в които адаптивността се базира на извод в Бейсов модел. Изследвани и анализирани са използваните в тях методи, стратегии и подходи за реализиране на адаптивност на базата на Бейсов модел.

Глава 4. АРХИТЕКТУРА НА ФУНКЦИОНАЛЕН МОДУЛ ЗА АДАПТИВНО ТЪРСЕНЕ В ДИГИТАЛНА БИБЛИОТЕКА

В рамките на настоящия труд предлагаме решение за реализиране на адаптивно търсене в дигитална библиотека. Предложеното решение е насочено към потребител, предпочитащ свободното търсене по ключови думи. Създадената архитектура цели предлагането на такава възможност за търсене, която да адаптира автоматично и прозрачно за потребител неговата заявка за търсене. Термините за това адаптиране на заявката се получават вследствие прилагане на адаптивна логика, реализирана на базата на *Бейсов модел за адаптиране и персонализиране*. В използвания подход всеки потребител в системата се класифицира към един или повече стереотипа. Стереотипите са създадени „ръчно“, съществуват като физически единици в системата, които са представени в Приложение 3.

4.1 Базова схема на адаптивна и персонализираща логика за реализиране на адаптивно търсене

Базовата схема е показана на Фиг.7. Тя проследява основните дейности, осъществявани от потребител и съответните отговори на системата.



Фиг.7 Базова схема на адаптивна и персонализираща логика за реализиране на адаптивно търсене

В първата стъпка на базовата схема, потребителят въвежда данни за себе си (чрез форма за регистрация) – цели в системата, език, области на знание, институция, роля и др. Възможните стойности на данните са предефинирани в онтологията на областта ТЕО [8, 98], описана в Глава 5. Те се използват във формата за регистрация като списъци с данни и падащи менюта. Във втората стъпка се създава потребителски модел (вектор на потребител) от въведените данни. Този вектор е съставен от пет елемента, които могат да имат стойност, ако такава е зададена от потребителя или да са с нулева стойност, при отсъствие на данни. Моделите (векторите) на потребителите се съхраняват в системата. В следващата стъпка се зареждат векторите на стереотипите от съответното хранилище. Тяхната структура е подобна на модела на потребител и възможните стойности са предефинирани в онтологията на областта ТЕО. Това позволява сравняване на вектора на потребител с вектора на всеки от стереотипите. Пълен списък на разработените вектори на стереотипите е представен в Приложение 3 от настоящия труд. В резултат на сравнението се извеждат „сходните" за потребител стереотипи, като това са тези, с които той има сходство над зададена гранична стойност. В следващата стъпка тези стереотипи се подават на *Бейсов модел за адаптиране и персонализиране*. Следва изчисление с помощта на модела. Това изчисление се извършва, като всички „сходни" на потребител стереотипи получават доказателство *истина*, докато всички останали стереотипи – доказателство *лъжа* и се задейства алгоритъм за извод (клъстериращ алгоритъм). Резултат от изчислението е извличане на списък с атрибути, които да се използват за разширяване на заявката за търсене. Извлечените атрибути се записват в модела на потребител.

Друга важна част от схемата е подаването на проста заявка за търсене от страна на потребител. Следва процес на замяна на оригиналната заявка на потребителя с адаптирана заявка. Адаптирането се реализира с помощта на изведените от *Бейсов модел за адаптиране и персонализиране* атрибути и формата за разширено търсене в системата. Детайли за адаптирането на заявката за търсене са представени в секция „Модул за прилагане на адаптивност при търсене" към Глава 4. Адаптираната заявка се изпраща вместо оригиналната заявка към хранилището с информационни ресурси. В резултат на това системата връща резултати, които са адаптирани към интересите на

потребител. В последната стъпка изведените резултатите от търсене се представят на потребителя.

4.2 Компоненти в архитектурата на Функционален модул за адаптивно търсене в дигитална библиотека

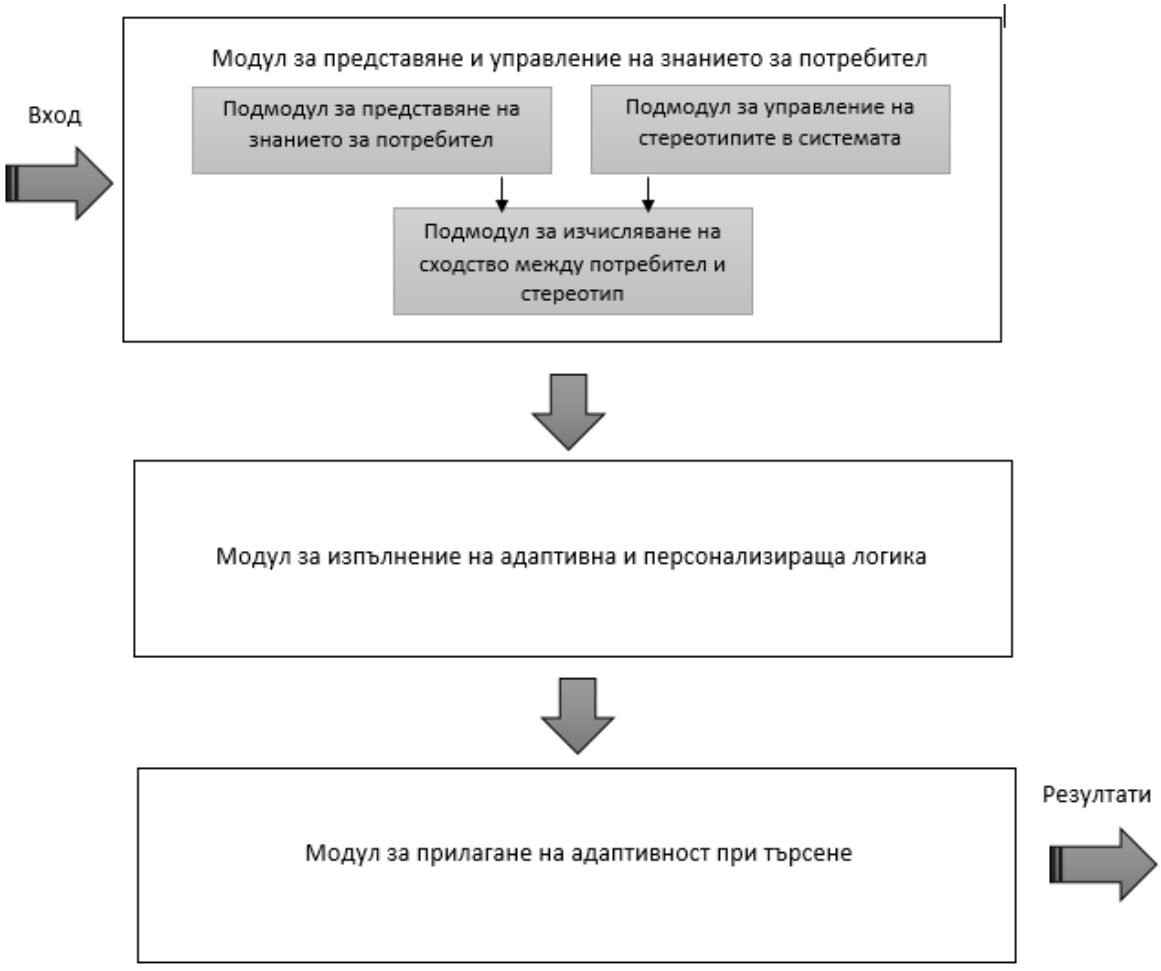
Компонентите в архитектурата на представения *Функционален модул за адаптивно търсене* са набор от главни модули и техни съставни подмодули, които действат съвместно и съгласувано за постигане на зададената цел – реализиране на адаптивно търсене. Главните модули и подмодули са както следва:

- *Модул за представяне и управление на знанието за потребител*. Този модул се състои от три подмодула:
  - *Подмодул за съхранение на стереотипите в системата*. Този модул осигурява съхранението на стереотипите. В рамките на настоящия труд са предложени и създадени стереотипи, на базата на знание за предметната област. Всеки стереотип се представя като вектор с тегла, съставен от същия тип елементи, както векторите на потребителите в системата.
  - *Подмодул за представяне на знанието за потребител*. Това представяне има формата на вектор с тегла, съставен от характеристиките на потребител. Тези характеристики имат обхват от възможни стойности, заложени при дизайна на онтологията ТЕО.
  - *Подмодул за изчисляване на сходство между потребител и стереотип* (класифициране на потребител към стереотип). Използва се моделът вектор-пространство и сходството се измерва като косинусово сходство между два вектора.
- *Модул за изпълнение на адаптивна и персонализираща логика .* В основата на този модул стои *Бейсов модел за адаптиране и персонализиране*, разгледан детайлно в секция „Модул за изпълнение на адаптивна и персонализираща логика посредством извод в Бейсова мрежа“ към Глава 4. Представеният *Бейсов модел за адаптиране и персонализиране* получава като входни данни стереотипа/ите на потребител и връща като изход атрибути. Тези атрибути се

използват за разширяване на заявката за търсене на потребител. Стойностите на атрибутите съставляват част от метаданните, с които са анотирани информационните ресурси в дигиталната библиотека.

- *Модул за прилагане на адаптивност при търсене чрез разширяване на заявката*. Този модул получава като вход заявка от потребител и атрибути за персонализиране от *Модул за изпълнение на адаптивна и персонализираща логика*, произвежда разширена заявка и връща адаптирани резултати от търсенето на потребител.

Архитектурата и компонентите на *Функционален модул за адаптивно търсене* са представени на Фиг.8.



Фиг. 8 Архитектура и компоненти на Функционален модул за адаптивно търсене в дигитална библиотека

4.2.1 Модул за изпълнение на адаптивна и персонализираща логика

Модул за изпълнение на адаптивна и персонализираща логика е компонент от архитектурата на Функционалния модул, съдържащ адаптивната логика. Централно звено в него е *Бейсов модел за адаптиране и персонализиране*, който служи като машина за извод при несигурност над знанието за потребител. Бейсовият модел приема като вход от *Модул за представяне и управление на знанието за потребител* стойност, която представлява изчисления стереотип/и за даден потребител. След изпълнение на извод в модела се извеждат атрибути, за които се приема, че описват съществени характеристики на информационните ресурси, подходящи за даден потребител. Изведените атрибути се подават на *Модул за прилагане на адаптивност при търсене*, където се използват за адаптиране на заявката на потребител.

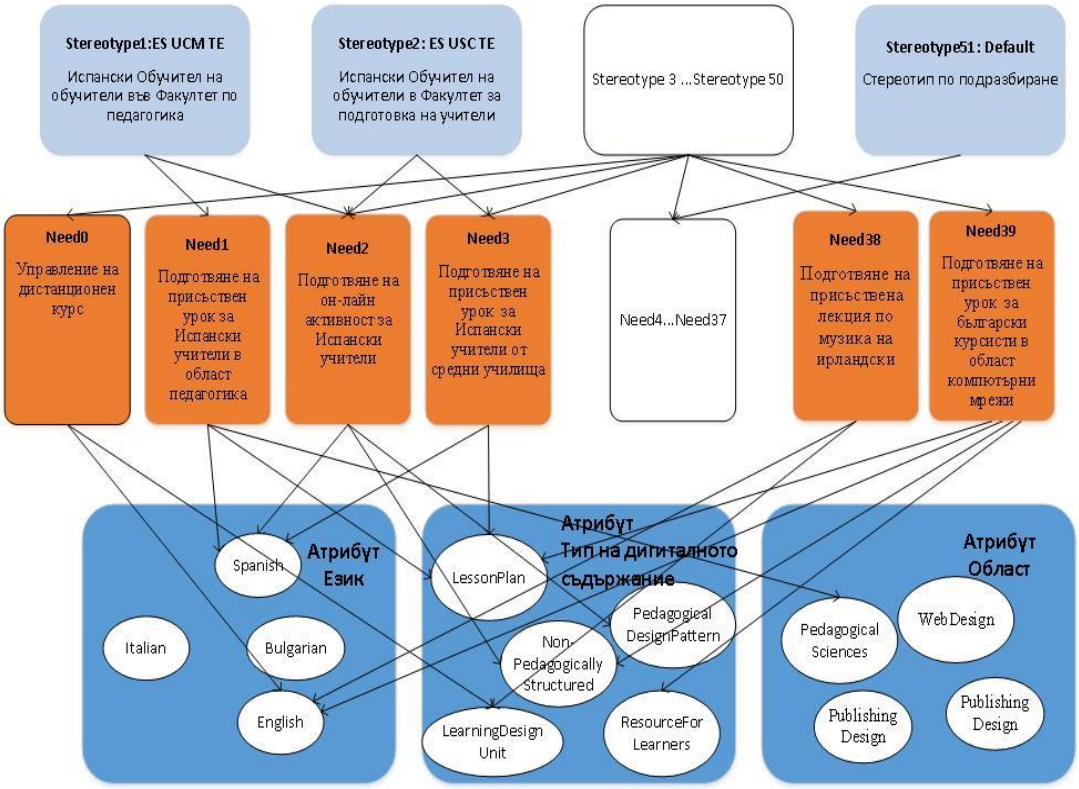
Бейсов модел за адаптиране и персонализиране

Логиката на Модела се основава на свойството на Бейсова мрежа и на каноничното NoisyOR взаимодействие да разпространяват вероятностите към свързаните възли в мрежата. Действието на Модела се базира на задаване на доказателства *истина* или *лъжа* на възлите от най-горния слой (*стереотипи*). Въз основа на тези доказателства и с помощта на алгоритъм за обновяване на вероятностите, Моделът изчислява стойностите на вероятностите на възлите от най-долния слой (*атрибути*).

Моделът представлява трислойна мрежа на Бейс, в която всеки слой се заема от различни бинарни случайни променливи, които могат да приемат стойност *истина* или *лъжа*. Моделът е изграден на предположението, че променливите от всеки слой са независими помежду си, и поради това не са свързани с дъги. Решението да се използва трислоен модел е аргументирано с това, че той притежава нужните изразни възможности да моделира областта от една страна и от друга не е прекалено сложен. Детайли за типове Бейсови модели и техните свойства са представени в секция „Бейсови модели – структура и параметризация“ към Глава 3.

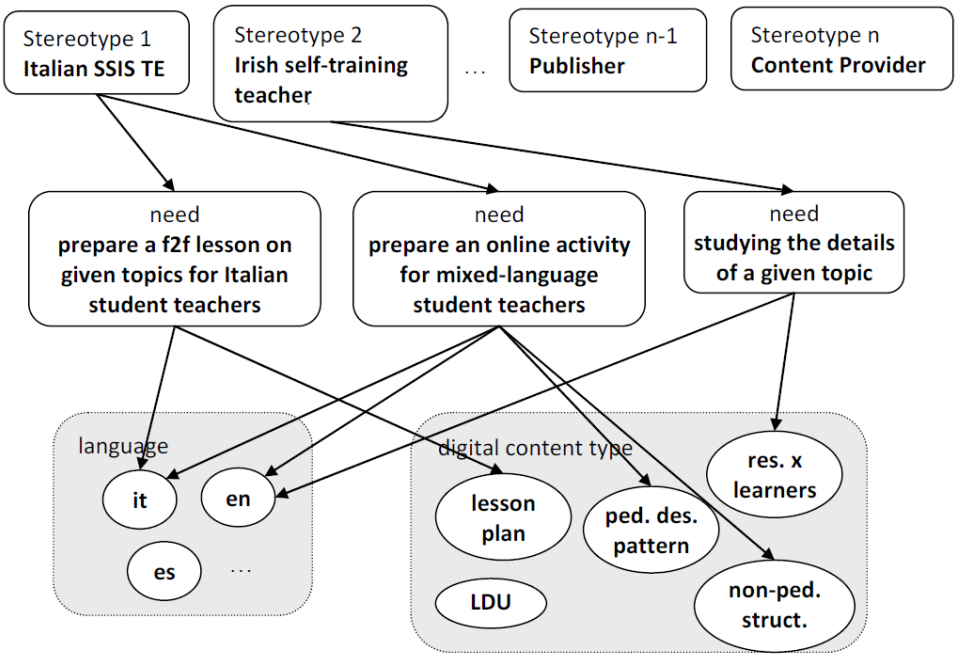
Мрежата на Модела е съставена от възли и насочени, утежнени дъги. Променливите от най-горния слой представят *стереотипите* в Модела, променливите от средния слой – *нуждите* и променливите от най-долния слой – *атрибутите.* Възлите в мрежата са детайлно разгледани в секцията „Възли в Модела“ в настоящия раздел. Случайните променливи на *стереотипите*, *нуждите* и *атрибутите* са бинарни, като могат да приемат стойности *истина* или *лъжа*. По отношение на атрибутите, това се прилага по следния начин: например атрибутите от групата „Език“ са: английски, испански, италиански и български. Всяка от тези стойности е представена като отделна бинарна променлива, която може да приема стойности *истина* или *лъжа.*

За редуциране на условните вероятности (CBP таблици) в Модела се използва каноничното взаимодействие NoisyOR. За изчисляване на условните вероятности на възлите от средния и най-долния слой - *нужди* и *атрибути* се използва метод, основан на техниката NoisyAdder, представен в секция „Взаимодействието NoisyAdder“ към Глава 3 от настоящия труд. На Фиг.9 е представена схема на *Бейсов модел за адаптиране и персонализиране*.



Фиг. 9 Схема на Бейсов модел за адаптиране и персонализиране

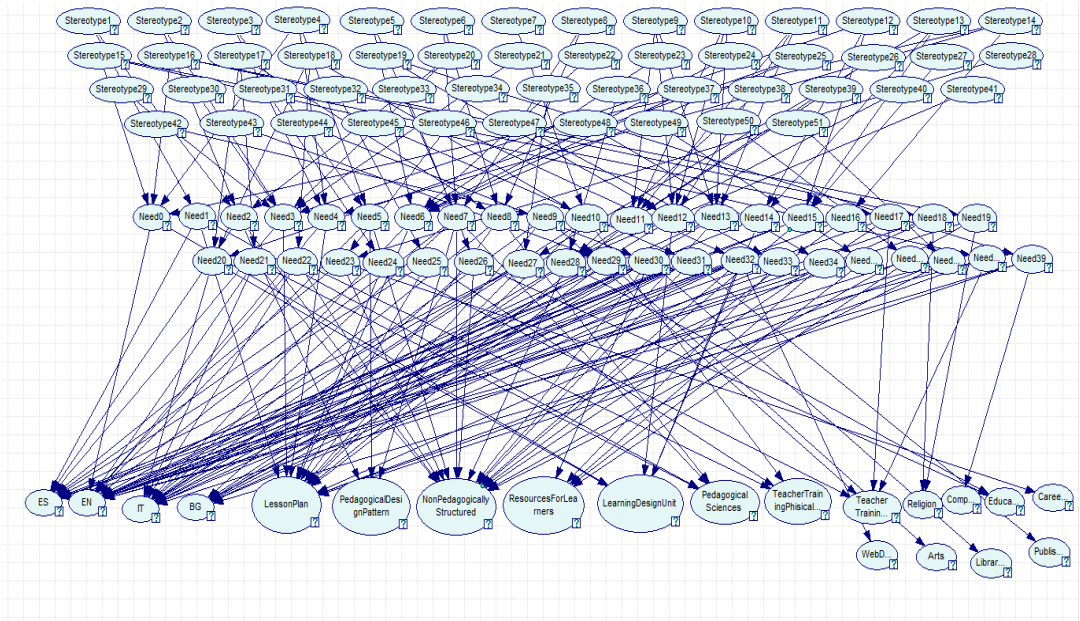
Схемата на *Бейсов модел за адаптиране и персонализиране* е разработена на базата на първоначалната схема, залегнала като насока за бъдещо развитие на системата в проекта Share.TEC [8]. Тази първоначална схема само набелязва насоките и дава поглед „отгоре“, но не представя готово решение. Първоначалната схема е показана на Фиг.10, видяна в [8].



Фиг.10 Първоначална схема за възможна употреба на стереотипи в дигиталната библиотека, според [8]

В рамките на настоящата работа са разработени и представени общо 51 стереотипа, които са създадени на базата на експертно познаване на предметната област и онтологичните класове и инстанции в онтологията TEO. С цел моделиране на областта са разработени съответните *нужди* за всеки *стереотип* и са идентифицирани класовете и инстанциите от онтологията TEO, които да бъдат използвани като *атрибути* в Модела. В първоначалната схема, представена в [8] атрибутите са групирани в две големи групи – „Език“ и „Тип на дигиталното съдържание“. Към тези групи е предложена и реализирана в Модела още една група атрибути: „Област на знание“. Софтуерният код на Бейсовия модел е представен в Приложение 1. Списък на променливите на възлите в Бейсовия модел - *стереотипи, нужди и атрибути* е представен в Приложение 2.

В реализацията на Модела участва възела LEAK node. Неговата употреба е въведена от Henrion [42], за да представи феномена, че дадено събитие може да възникне спонтанно, при отсъствие на извънредна причина. Казано по друг начин, чрез LEAK node се моделира вероятността дадено събитие да се случи, при положение, че нито един от предшестващите му възли не се случва. На Фиг. 11 е представена реализацията на модела в графичната среда за разработка GeNIe 2.0 [14].



Фиг. 11 Реализацията на Модела в графичната среда за разработка GeNIe 2.0

Възли в Модела

Възли от най-горен слой (стереотипи) - Бинарните променливи от най-горния слой в Модела представят стереотипите в системата. В рамките на настоящата работа са разработени и представени 51 стереотипа (Stereotype1 – Stereotype 51), в това число стереотип по подразбиране, за случаите, в които не е възможно класифициране на потребител към нито един от съществуващите стереотипи. Обикновено даден стереотип обхваща набор от характеристики, които се приема, че са верни за членовете на този стереотип. При класифициране на потребител към стереотип/и, променливата/те на този/тези стереотип/и в Модела получава/т стойност *истина*, докато променливите на всички останали стереотипи – стойност *лъжа*.

Възли от среден слой (нужди) - Системите за препоръчване базират своята функционалност на модел на потребител, целящ представяне на характеристиките и нуждите му. Нуждите в системите за препоръчване се описват като ограничение върху вида и типа ресурси, които потребителят има вероятност да иска да види. Бинарните променливи от средния слой в Модела представят нуждите в системата. В рамките на настоящата работа са разработени 40 нужди (Need0 – Need39) за описване на различните потребности на потребителите в системата. Броят на нуждите е по-малък от броя на стереотипите, защото в някои от случаите, едни и същи нужди се използват от няколко стереотипа. Всяка променлива на стереотип в Модела е свързана към една или няколко променливи на нуждите. Всяка променлива, представяща нужда в Модела е свързана към набор от атрибути, които показват какъв тип информационно съдържание ще задоволи тази нужда.

Например: Потребител, класифициран към стереотип „Български университетски преподавател в Софийски университет“ се предполага, че ще има следните нужди в системата:

- Подготвяне на присъствена лекция за български студенти
- Управление на курс в система за електронно обучение Moodle
- Подготвяне на доклад, статия или учебен материал за публикуване

Възли от най-долен слой (атрибути) - Бинарните променливи от най-долния слой в Модела представят атрибутите в системата. Атрибутите и техните възможни стойности са взети от онтологията на областта TEO и участват като елементи в модела за метаданни CMM [85] на Share.TEC, представен в секция „Модул за прилагане на адаптивност при търсене” към Глава 4. Този подход гарантира, че извлечените от Модела *атрибути* могат да се използват за идентифициране на ресурсите, чиито метаданни съвпадат с тези атрибути. Атрибути са общо 20 на брой и са групирани в три групи: „Език“, „Тип на дигиталното съдържание“ и „Област на знание“. Всички те са класове или инстанции в онтологията TEO.

- Атрибути от групата „Език“ са: английски (EN), испански (ES), италиански (IT) и български (BG).

- Атрибутите от групата „Тип на дигиталното съдържание“ са: Учебна единица, План на урок, Педагогически модел на структурата, Не-структуриран педагогически и Ресурси за обучаеми. Следва описание на атрибутите, взето от [98]:

  - Учебна единица (LearningDesignUnit) -– многократно използваема единица, която моделира структурата и потока на учебния процес, включваща актьори, ресурси, инструменти, активности и методи.

  - План на урок (LessonPlan) -– ресурс, който описва урока, чрез неговите цели, стратегии, поддържащи инструменти, ресурси, възможни употреби и т.н. Той се реализира в дадена област на знанието и е насочен към конкретна целева група.

  - Педагогически модел на структурата (PedagogicalDesignPattern) -– използва дадени стратегии или техники с цел да направи добрите практики преносими. Предназначен е да осигурява решения на типични образователни проблеми.

  - Не-структуриран педагогически (Non-Pedagogically Structured) -– Спомагателен материал, който не съдържа специална педагогическа структура или насоченост и е общо-целеви. Тази категория включва материали, които не са разработени с педагогическа цел, но въпреки това могат да се използват в учебния процес като информационен ресурс.

  - Ресурси за обучаеми (ResourceForLearners) -– педагогически структурирано съдържание, предназначено да бъде използвано за обучение.

- Атрибутите от групата „Област на знание“ са: Педагогическа наука (PedagogicalSciences), Обучение на учители - физическо възпитание (TeacherTrainingPhisicalTraining), Обучение на учители по музика (TeacherTrainingInMusic), религия (Religion), компютърни науки (ComputerScience), образователен мениджмънт (EducationalManagement), Кариерно ориентиране (CareerAdvising), уеб дизайн (WebDesign), изкуства (Arts), Библиотечна информация (LibraryInformationArchive) и Печатно оформление (PublishingDesign) [98].

Структура на Модела

Възлите в модела са реализирани като възли от общ тип и възли от тип NoisyAdder, като от общ тип са възлите от най-горния слой (*стереотипи*), а от тип NoisyAdder са възлите от средния (*нужди*) и най-долния слой (атрибути). Посредством употребата на възли от тип NoisyAdder се цели намаляване на размера на CBP таблиците от експоненциален до линеен. Първоначално на възлите от най-горния слой (с*тереотипи*) се назначават априорни вероятности въз основа на експертно знание за областта. От тези априорни вероятности се извеждат предварителните условните вероятности на възлите от долните два слоя на Модела. Впоследствие априорните вероятности се заменят с доказателства, които имат стойност *истина* или *лъжа*. За изчисляване на условните вероятности на възлите от средния и най-долния слой - *нужди* и *атрибути* се използва метод, основан на техниката NoisyAdder, подобна на тази в [76].

Релациите в Модела се представят с утежнени, насочени дъги между възлите от най-горния слой и възлите от средния слой и между възлите от средния слой и възлите от най-долния слой. Няма релации между възлите от най-горния слой и възлите от най-долния слой, както и такива между възлите от един и същи слой.

Параметри в Модела

- Тежести на дъгите в Модела

Тежестта на дадена дъга в Модела показва значимостта на тази релация. Първоначалното предположение, залегнало при разработката на Модела е, че всички релации са еднакво значими, и поради това с еднаква стойност. Ако бъдещите изследвания покайат, че това не е валидно е възможна промяна в Модела на по-късен етап.

- Априорни вероятности на възлите от най-горния слой (*стереотипи*)

Проведени са експерименти с различни стойности на априорните вероятности, разгледани в Глава 6 „Изследвания и експерименти“ от настоящия труд.

Първоначално зададените априорни вероятности на възлите от най-горния слой на модела (*стереотипи*) са със стойности p(x = true) = 0,2 и p(x = false) = 0,8 за всички възли. Тези априорни вероятности служат единствено за предварително определяне на условните вероятности на възлите от другите два слоя в Модела, тъй като впоследствие се заменят със доказателства *истина* или *лъжа*.

Задаването на доказателства *истина* или *лъжа* на възлите от най-горния слой (*стереотипи*) води до разпространяването им към свързаните възли от другите слоеве. В резултат на това разпространение, възлите от долните два слоя получават стойности на вероятностите.

- Изчисляване на условните вероятности за възлите от средния (*нужди*) и най-долния (*атрибути*) слоеве

За изчисляване на условните вероятности на възлите от средния и най-долния слоеве използваме и адаптираме техниката от [76], която е основана на техниката NoisyAdder.

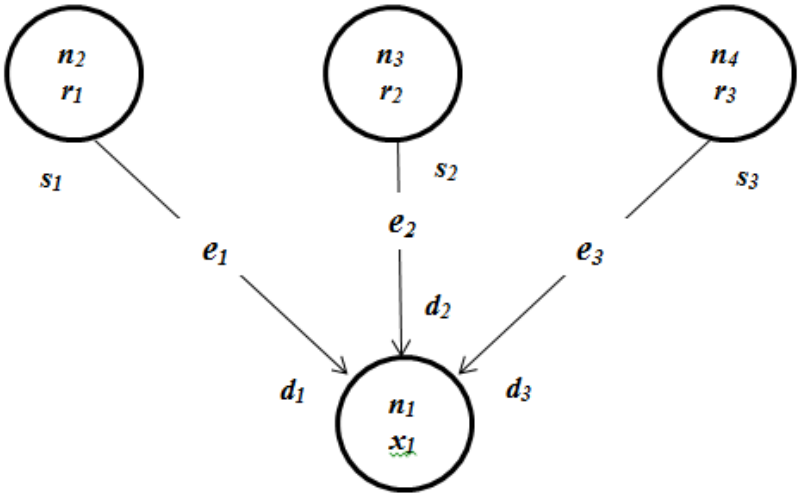
Дефиниции и формули

Нека n да бъде възел в мрежата, като $n \in N$, където N е множеството на всички възли в мрежата.

Нека x да бъде бинарна случайна променлива, съответстваща на възел n от мрежата, като $x \in X$, където X е множеството на всички случайни променливи, съответстващи на множеството N от възли.

Нека e да бъде дъга в мрежата, като $e \in E$, където E е множеството на всички дъги в мрежата.

Всяка дъга e може да бъде представена чрез тройката $e = (s, d, w)$, където: $s \in N$ е началния възел, $d \in N$ е крайния възел и w е положително реално число, представящо тежестта на дъгата. Тежестта w на дъгата e обозначаваме с w_e. Фиг. 12 показва зададените означения на възел в мрежата.



Фиг. 12 Схема на възел от мрежата

За всеки възел в мрежата $n \in N$, дефинираме множеството на входящи към него дъги като D и множеството на предшестващите го възли като R. Тогава: $D := \{(s, n, w) \in E\}$ и $R := \{s \mid (s, n, w) \in E\}$.

Валюацията на логически израз е едно възможно присвояване на стойностите *истина* или *лъжа* на всички елементарни съждения, участващи в него. Нека V(P) е валюацията на случайните променливи в множеството на предшестващи случайни променливи Rp. Валюацията на единична случайна променлива $r \in Rp$ обозначаваме с V(Pr). Използвайки тези валюации, условната вероятност $P(x = true \mid Rp)$, че n се случва, при условие, че предшестващите възли на n се случват, се изчислява с помощта на средната тежест на дъгите на предшестващите случайни променливи. При това тежестта на дъгата е се калкулира, при условие, че стойността на предшестващата случайна променлива не е *лъжа*. Или тежестта w на дъгата e, за дадена стойност V(P) ще се изразява като w(e, V(P)). Тогава w(e, V(P)) ще връща тежестта на дъгата e, тогава когато предшестваща случайна променлива $r \in Rp$ има стойност *истина*. В противен случай тя ще връща стойност 0.

За да изчислим условната вероятност P(x | V(P)) за възел n, при w(e, V(P)), като средната тежест на неговите предшественици, трябва да сумираме тежестите на всички дъги в D, за които имаме стойност на предшестващата случайна променлива *истина* и да ги разделим на сумата от всички тежести в D.

$$P(x = \text{true} \mid Rp) = \frac{\sum_{e \in D} w(e,\ V(P))}{\sum_{e \in D} w_e}$$

$$P(x = \text{false} \mid Rp) = 1 - P(x = \text{true} \mid Rp)$$

При изчисляване на стойността на знаменателя (сумата от всички тежести в D), не трябва да забравяме да добавим към сумата и стойността на тежестта на възела LEAK.

- Алгоритъм за обновяване на вероятностите в Модела

Алгоритъмът, използван за обновяване на вероятностите в мрежата, е т.нар. клъстериращ алгоритъм (Clustering algorithm). Той е избран поради това, че е най-бързият, известен точен алгоритъм, използван за обновяване на вероятностите в Бейсова мрежа. Алгоритъмът за клъстеризиране е предложен първоначално от Lauritzen et al [55] и впоследстие подобрен от Jensen et al [47]. Този алгоритъм работи на два етапа: (1) компилиране на насочения граф в свързано дърво и (2) обновяване на вероятностите в това свързано дърво. Компилирането на Бейсова мрежа и извършване на операции върху компилираната версия е често срещана практика. Клъстериращият алгоритъм е алгоритъмът по подразбиране в графичната среда за разработка GeNIe [14].

4.2.2 Модул за представяне и управление на знанието за потребител

Модул за представяне и управление на знанието за потребител е компонент от архитектурата на Функционалния модул, служещ за представяне на потребител като вектор/и и класифицирането му към стереотип/и. Модулът изпълнява следните главни функции: (1) взима данните от регистрацията на потребител; (2) конструира от тях вектор на потребителя; (3) сравнява този вектор с предварително дефинираните вектори на стереотипите и (4) избира най-сходния/те стереотип/и за даден потребител. Тези стереотипи се подават на *Модул за изпълнение на адаптивна и персонализираща логика* за извършване на извод с тях.

Класифицирането на потребител към стереотип не винаги е лесна задача в една система. Това е така, защото всеки потребител се представя в съответната система чрез широк набор от директни и индиректни характеристики, мнозинството от които не са числови. Едно от възможните решения на този проблем е клъстеризиране или групиране на потребителите на базата на всичките тези характеристики. При този подход са налице трудности за правилното клъстеризиране на не-числови множества от данни, като за това са нужни и голямо количество ресурси [93, 97].

Ние сме приложили друг подход, чиито корени са в областта Извличане на информация, базиран на модела вектор-пространство. Чрез избрания подход, представяме потребител и стереотипи като 5-мерни вектори с тегла, съставени от еднакъв брой и тип елементи, взети от онтологията ТЕО. Тогава изчисляваме тяхното сходство посредством метода на косинусовото сходство (Cosine similarity) и свеждаме задачата за класифициране на потребител към стереотип до просто съпоставяне на вектори.

Този подход позволява всеки потребител да бъде класифициран поне към един стереотип, като в случаите, когато сходството между потребител и всички стереотипи в системата е под зададена минимална стойност, потребителят се класифицира към стереотип „по подразбиране". Детайли за предложеното решение са разгледани в настоящата секция.

Векторно представяне на потребител и стереотипи

За представяне на потребител и стереотипи в *Модул за представяне и управление на знанието за потребител* се използва онтологията ТЕО и модел, базиран на модела вектор-пространство.

Структура на вектор – дефиниции и формули. Сходство между два вектора

Нека U_i да бъде потребител в системата, като $Ui \in U$, където U е множеството на всички потребители в системата. Нека S_i да бъде стереотип в системата, като $S_i \in S$, където S е множеството на всички стереотипи в системата.

Нека L_i да бъде езикът на потребител U_i, като $L_i \in L$, където L е множеството на езиците в системата.

Нека G_i да бъде целта на потребител U_i, като $G_i \in G$, където G е множеството на целите.

Нека R_i да бъде ролята на потребител U_i, като $R_i \in R$, където R е множеството на ролите.

Нека I_i да бъде институцията на потребител U_i, като $I_i \in I$, където I е множеството на институциите в системата.

Нека K_i да бъде областта на знание на потребител, като $K_i \in K$, където K е множеството на всички области на знание в системата.

Нека W_{tj} е тежестта на елемента j в t-я модел на потребител, като $W_{tj} \in W$, където $W \in [0, 1]$.

Тогава всеки вектор на потребител U_i се състои от елементите (L_i, G_i, R_i, I_i, K_i) и може да се представи с вектора:

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

Нека L_j да бъде езикът на стереотип S_i, като $L_j \in L$, където L е множеството на езиците в системата.

Нека G_j да бъде целта на стереотип S_i, като $G_j \in G$, където G е множеството на целите.

Нека R_j да бъде ролята на стереотип S_i, като $R_j \in R$, където R е множеството на ролите.

Нека I_j да бъде институцията на стереотип S_i, като $I_j \in I$, където I е множеството на институциите в системата.

Нека K_j да бъде областта на знание на стереотип S_i, като $K_j \in K$, където K е множеството на всички области на знание в системата.

Нека W_{ij} е тежестта на елемента j в i-я стереотип и $W_{ij} \in W$, където $W \in [0, 1]$.

Тогава всеки вектор на стереотип S_i се състои от елементите (L_j, G_j, R_j, I_j, K_j) и може да се представи с вектора:

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

За определяне на сходните стереотипи за потребител се извършва изчисляване на сходството между вектора/ите на потребителя с векторите на всички стереотипи в системата, посредством метода на косинусовото сходство, разгледан в секция „Модел - вектор от термини“ към Глава 2 от настоящия труд.

Използвайки метода на косинусовото сходство можем да изчислим сходството между потребител и стереотипи чрез:

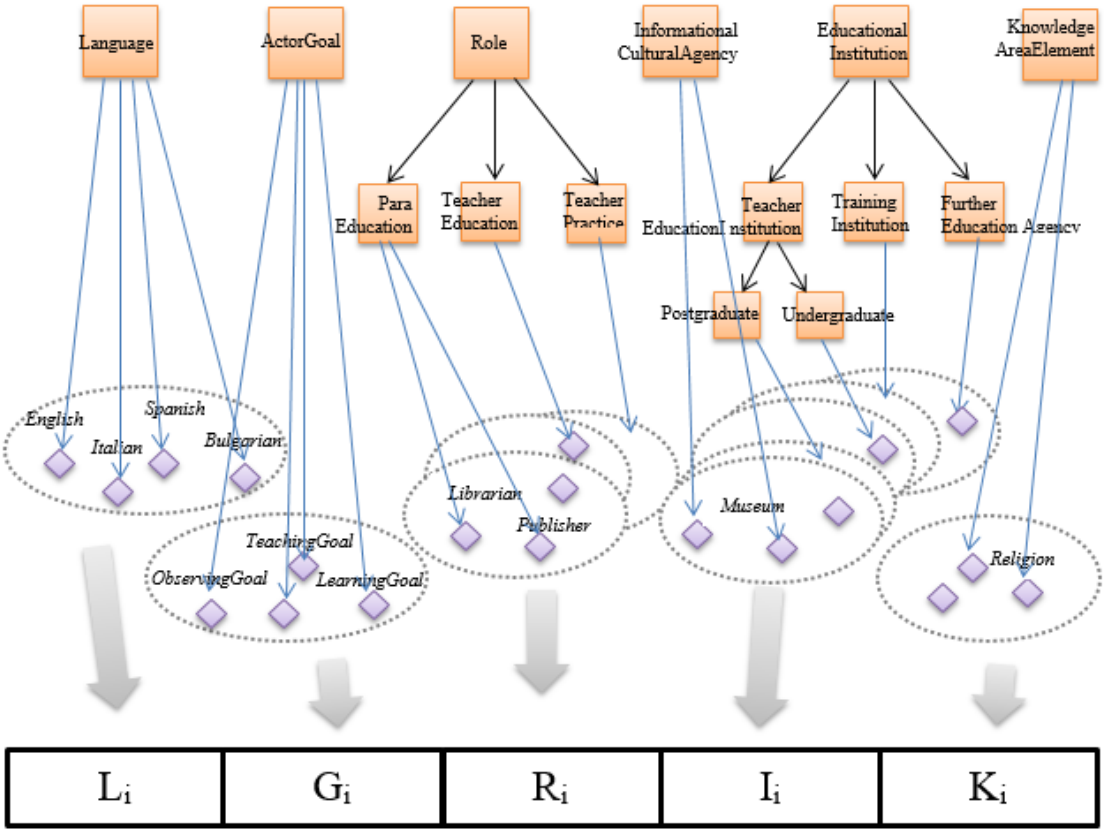
$$\text{sim}(\overrightarrow{s_i}, \overrightarrow{u_t}) = \cos\boldsymbol{\vartheta} = \frac{\overrightarrow{s_i} \cdot \overrightarrow{u_t}}{|\overrightarrow{s_i}||\overrightarrow{u_t}|} = \frac{\sum_j w_{ij} \cdot w_{tj}}{\sqrt{\sum_j w_{ij}^2} \cdot \sqrt{\sum_j w_{tj}^2}}$$

Като уравнението има стойности:

$\text{sim}(\overrightarrow{s_i}, \overrightarrow{u_t}) = 1$, когато $\overrightarrow{s_i} = \overrightarrow{u_t}$

$\text{sim}(\overrightarrow{s_i}, \overrightarrow{u_t}) = 0$, когато $\overrightarrow{s_i}$ и $\overrightarrow{u_t}$ нямат нито един общ елемент.

Потребителят в системата се представя чрез един или повече 5-мерни вектора с тегла. Всеки от разработените и представени в настоящата работа стереотипи се представя точно чрез един 5-мерен, предварително дефиниран вектор с тегла. Елементите от векторите на потребител и стереотипи са инстанции на онтологичните класове *Language, ActorGoal, Role, (EducationalInstitution InformationCulturalAgency) и KnowledgeAreaElement* от онтологията TEO. Поради този подход възможните стойности на елементи са ограничени до крайно множество и правят задачата тривиална. Структурата на вектор и връзката му с онтологията TEO е показана на Фиг. 13.



Фиг. 13 Структура на вектор и връзката му с онтологията ТЕО

Таблица 4 показва инстанциите на онтологичния клас *Language*, които представляват множеството на стойностите за 1-я елемент на векторите.

Таблица 4 Инстанции на класа *Language*, съставящи множеството от стойности на 1-я елемент на вектор

Клас	Инстанция (елемент)	Наименование
Language (Език)	English-EU, Spanish, Bulgarian, Italian	Английски език Испански език Български език Италиански език

Таблица 5 показва инстанциите на онтологичния клас *ActorGoal*, които представляват множеството на стойностите за 2-я елемент от векторите.

Таблица 5 Инстанции на класа *ActorGoal*, съставящи множеството от стойности на 2-я елемент на вектор

Клас	Инстанция (елемент)	Наименование
ActorGoal	TeachingGoal	Обучителна цел

	LearningGoal ObservingGoal LearnerVisualizationGoal	Образователна цел Наблюдателна цел Визуализационна цел

Таблица 6 показва инстанциите на онтологичния клас *Role*, които са множеството от стойностите на 3-я елемент от векторите.

Таблица 6 Инстанции на класа *Role*, съставящи множеството от стойности на 3-я елемент на вектор

Под-клас	Инстанция (елемент)	Наименование
ParaEducation	ContentDeveloper Librarian Publisher ServiceProvider	Разработчик на съдържание Библиотекар Издател Доставчик на услуга
TeacherEducation	StudentTeacher TeacherEducator TeacherTrainer	Университетски преподавател Обучител на обучители Инструктор/ обучител на обучители
TeacherPractice	Coordinator HeadOfSchool Mentor-InductionMentor Teacher TeachingAssistant Technician Trainer Tutor	Координатор Директор Ментор /Въвеждащ наставник Учител Стажант-учител Технически изпълнител Обучител/ инструктор Обучаващ наставник

Таблица 7 показва инстанциите на онтологичния клас *EducationalInstitution,* служещи, заедно с инстанциите на класа *InformationalCulturalAgency* съставящи множеството от стойности на 4-я елемент на вектор.

Таблица 7 Инстанции на класа *EducationalInstitution,* попълващи 4-я елемент на вектор

Под-клас	Инстанция (елемент)	Наименование
FurtherEducation Agency	FurtherEducationAgencyES_4 FurtherEducationAgencyIT_2 FurtherEducationAgencyES_8 FurtherEducationAgencyIT_1 FurtherEducationAgencyIT_4 FurtherEducationAgencyIT_6 FurtherEducationAgencyIT_3 FurtherEducationAgencyIT_7 FurtherEducationAgencyES_2 FurtherEducationAgencyES_1	Centros de formación de adultos Centro Territoriale Permanente per l'Istruzione e la Formazione in età adulta Agenzia Nazionale Sviluppo Autonomia Scolastica Agenzia Nazionale Sviluppo Autonomia Scolastica Centro Iniziativa Democratica Insegnanti Istituto per lo sviluppo della formazione professionale dei lavoratori Associazione Italiana Formatori Conservatorio Instituto Nacional de Empleo

	FurtherEducationAgencyES_6 FurtherEducationAgencyBG_1 FurtherEducationAgencyES_3 FurtherEducationAgencyIT_5 FurtherEducationAgencyES_5 FurtherEducationAgencyES_7 FurtherEducationAgencyIT_8	Ministerio de Educación, Política Social y Deporte Centros de Formación de Enseñanzas deportivas Департамент за информация и усъвършенстване на учителите Empresas de formación Ufficio Scolastico Regionale Conservatorio Centro Nacional de Recursos para la Orientación Profesional Accademia delle belle arti
TeacherEducation Institution - Postgraduate - UnderGraduate	PostgraduateIT_2 PostgraduateIT_4 PostgraduateBG_1 PostgraduateIE_1 PostgraduateES_1 PostgraduateIT_1 PostgraduateIE_2 PostgraduateIT_3 UndergraduateES_1 UndergraduateBG_2 UndergraduateBG_1 UndergraduateIE_2 UndergraduateBG_3 UndergraduateIT_2 UndergraduateES_2 UndergraduateIE_3 UndergraduateIT_1 UndergraduateIE_1	Scienze della Formazione (II ciclo) Scuola di Dottorato riguardante tematiche didattiche e psico-pedagogiche Отдел “Докторанти и следдипломна квалификация" към СУ Secondary teacher training Facultad de Educación (Máster y Doctorado) Scuola di Specializzazione per l'Insegnamento Secondario Online teacher training Scienze motorie (II ciclo) Facultad de Educación Великотърновски университет Софийски университет Training of Secondary School Music Teachers Шуменски университет Scienze motorie (I ciclo) Universidad de Educación a distancia Training of Secondary School Religion Teachers Scienze della Formazione (I ciclo) Primary teacher training
TrainingInstitution	TrainingInstitutionBG_1 TrainingInstitutionES_1 TrainingInstitutionES_2 TrainingInstitutionIT_1 TrainingInstitutionIT_2 TrainingInstitutionIT_3 TrainingInstitutionIT_4	CISCO мрежова академия Instituto de formación Profesional Centros de Cualificación Profesional Inicia Istituto di formazione professionale iniziale Ente di formazione professionale Istituto di formazione iniziale e professionale Ente di formazione a distanza

Таблица 8 показва списъкът на инстанциите за клас *InformationalCulturalAgency*.

Таблица 8 Инстанции на класа *InformationalCulturalAgency*, попълващи 4-я елемент на вектор

Под-клас	Инстанция (елемент)	Наименование

InformationalCulturalAgency	GovernamentAgency InformationCulturalAgencyES_1 InformationCulturalAgencyES_2 InformationCulturalAgencyES_3 InformationCulturalAgencyIE_1 InformationCulturalAgencyIE_2 InformationCulturalAgencyIE_3 InformationCulturalAgencyIE_4 InformationCulturalAgencyIE_5 InformationCulturalAgencyIE_6 InformationCulturalAgencyIT_1 Library Museum PublishingHouse ResearchCenter WebAgency	Governament/State Agency Ministerio de Ciencia e Innovación Ministerio de Educación, Política Social y Deportes Sindicatos Educativos Foras na Gaeilge Consumer protection agencies Archives Welfare groups Advisory bodies Cultural organisations Associazione professionale Библиотека Музей Publishing House ResearchCenter WebAgency

Таблица 9 показва показва инстанциите на онтологичния клас съставящи множеството от стойности на 5-я елемент на вектор

Таблица 9 Инстанции на класа *KnowledgeAreaElement*, съставящи множеството от стойности на 5-я елемент на вектор

Клас	Инстанция (елемент)	Наименование
KnowledgeAreaElement	PedagogicalSciences TeacherTraining-PhisicalTraining TeacherTrainingInMusic Religion ComputerScience EducationalManagement CareerAdvising WebDesign Arts LibraryInformationArchive PublishingDesign	Педагогическа наука Обучение на учители - физическо възпитание Обучение на учители по музика Религия Компютърни науки Образователен мениджмънт Кариерно ориентиране Уеб дизайн Изкуства Библиотечна информация Печатно оформление

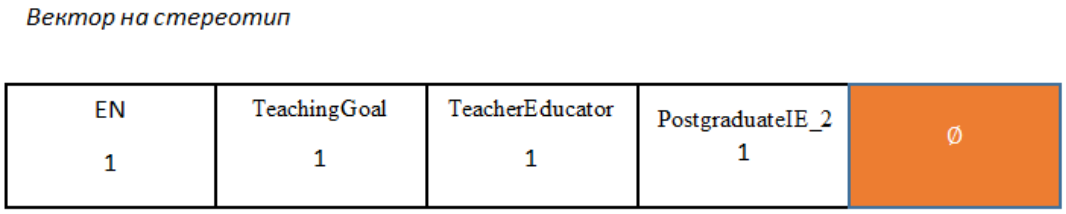
Схема за задаване на тегла на елементите във векторите

- Вектор на стереотип

Елементите във векторите на стереотипите имат предварително зададени тегла, равна на единица за всеки дефиниран елемент във вектора. Част от стереотипите в системата имат празно множество за един или повече елемента.

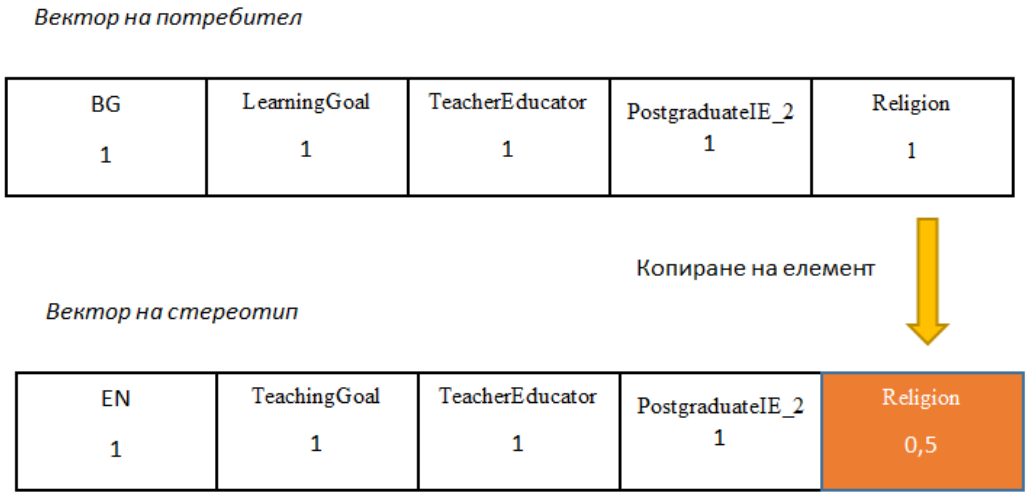
Това е в случаите, когато този елемент/и не е от съществено значение за дефиницията на стереотипа. В такива случаи са възможни три подхода: липсата се приема за нула, липсата се приема за единица и липсата се приема за каквато е другата стойност. Приложен е вариант на последния подход. В процеса на сравняване на вектор на стереотип с вектор на потребител, елементът от вектора на стереотип, който е празно множество, приема стойността от съответния елемент от вектора на потребител, но с намалено тегло – със стойност 0,5 вместо единица.

Логиката зад този подход се корени в идеята, че тъй като този елемент не е от съществено значение за дефиницията на стереотипа, то и теглото му трябва да е по-малко. По-малкото тегло рефлектира върху процеса на изчисляване на сходство (класификацията на потребител). Фиг.14 показва вектор на стереотип от този тип, преди процес на сравнение с вектор на потребител.



Фиг.14 Вектор на стереотип с елемент празно множество

Фиг.15 показва процеса на задаване на стойност на елемента празно множество при сравнение с вектор на потребител.



Фиг. 15 Процес на задаване на стойност на елемента празно множество при сравнение с вектор на потребител

- Вектор на потребител

При задаване на тегла на елементите във вектор на потребител е избран следният подход: тъй като в предложеното решение всички стойности на елементите във вектора на даден потребител се задават директно от самия потребител, те получават най-голямото възможно тегло, равно на единица. Това е така, защото се приема, че тогава когато потребителят задава дадена стойност, той е сигурен в избора си. В бъдещото развитие на системата е възможно да се реализира индиректно събиране на част от данните за вектора на потребител автоматично от системата. В този случай, теглата на индиректно изведените стойности на елементите би трябвало да са по-малки от тези на директно заявените от потребителя.

В случаите, когато елемент от вектора на потребител е празно множество (когато потребител не е задал стойност за този елемент при регистрацията си), тогава този елемент получава тегло 0, което е равно на пълно „несходство“ със съответния елемент от вектора на стереотипите.

В бъдещото развитие на системата се предвижда модификация на теглата от векторите, така че те да отразяват релациите в онтологията TEO. Например при липса на сходство между съответни елементи от векторите на потребител и стереотип, да се проверява за елементи от същия клас, съседен клас или суперклас, в който случай тези елементи да получават по-малко тегло.

Този подход се изразява във следното: ако елемент, свързан с онтологичния клас *Role* от векторите на потребител и стереотип не съвпада, но и двата са инстанции на един и същ под-клас, то те могат да се приемат за сходни, но с по-малко тегло. Например, ако във вектора на потребител елементът е *StudentTeacher*, докато във вектора на стереотип, той е *TeacherEducator*, двата елемента биха могли да се приемат за сходни, но не със тежест единица, както би било в случайч че те са идентични а със по-малка тежест, примерно - 0,2. Логиката зад този подход би била, че тъй като елементите *StudentTeacher* и *TeacherEducator* са инстанции на класа *TeacherEducation,* който е под-клас на класа *Role*, то те споделят някаква близост. За да се приложи този

подход е необходимо да се проведат допълнителни изследвания, които да очертаят случаите и онтологичните класове, за които той би бил валиден. Това е една от насоките за бъдещо развитие на системата.

Конструиране на вектор на стереотипи и потребител

Векторите на стереотипите са създадени предварително и се съхраняват в Хранилище на стереотипите в системата. В Таблица 10 са показани векторите на част от стереотипите. Пълен списък с векторите на всички стереотипи се намира в Приложение 3 на настоящия труд.

Таблица 10 Вектори на стереотипи в системата

Име на Стереотип	Описание на стереотип	Вектор
Stereotype1: ES UCM TE	Испански Обучител на обучители във Facultad de Educación (Máster y Doctorado)	ES(1), TeachingGoal (1), TeacherEducator (1), PostgraduateES_1(1),
Stereotype2: ES USC TE	Испански Обучител на обучители във Факултет за подготовка на учители (Facultad de Formación del Profesorado)	ES(1), TeachingGoal(1), TeacherEducator(1), PostgraduateES_2(1),
Stereotype3: IR SST TE	Ирландски Обучител на обучители на учители от средни училища	EN(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIE_1(1),
Stereotype4: IR OTT TE	Ирландски Обучител на обучители зает с он-лайн обучение на учители	EN(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIE_2(1),
Stereotype5: IT SSIS TE	Италиански Обучител на обучители в Школа за специализация на учители от основни училища (Scuola di Specializzazione per l'Insegnamento Secondario)	IT(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIT_1(1), PedagogicalScience(1)
Stereotype6: IT SF TE	Италиански Обучител на обучители по Образователни науки (Scienze della	IT(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIT_2(1), PedagogicalScience(1)

	Formazione)	
Stereotype7: IT SM TE	Италиански Обучител на обучители в област Физическо възпитание и спорт (Scienze motorie)	IT(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIT_3(1), TeacherTraining-PhisicalTraining(1)
Stereotype8: IT SD TE	Италиански Обучител на обучители в Докторантска Школа за проблеми на образованието и психо-Педагогическа науката (Scuola di Dottorato riguardante tematiche didattiche e psicopedagogiche)	IT(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIT_4(1), PedagogicalScience(1)
Stereotype9: BG SU ST	Български университетски преподавател в Софийски университет	BG(1), TeachingGoal(1), StudentTeacher(1), UndergraduateBG_1(1),
Stereotype10: BG ShU ST	Български университетски преподавател в Шуменски университет	BG(1), TeachingGoal(1), StudentTeacher(1), UndergraduateBG_3(1),

4.2.3 Модул за прилагане на адаптивност при търсене

Модул за прилагане на адаптивност при търсене е компонент от архитектурата на Функционален модул за адаптивно търсене в дигитална библиотека. Модулът приема като вход оригиналната заявка от потребител и атрибути от *Модул за изпълнение на адаптивна и персонализираща логика* и изпълнява адаптиране на заявката. Адаптираната заявка заменя оригиналната заявка на потребител и се изпраща към хранилището с информационни ресурси.

От съществено значение за успеха на реализираното адаптивно търсене е това, че информационните ресурси в дигиталната библиотека са анотирани с

метаданни и машината за търсене в системата е настроена да използва тези полета от метаданни при търсенето.

Метаданните са данни, които описват дадения ресурс, посредством специални тагове. Стандартът LOM е международно признат отворен стандарт, публикуван от асоциацията IEEE, който се използва за анотиране на учебни обекти. Името на стандартът LOM идва от абревиатурата на английското „Learning Objects Metadata“ [29]. В рамките на стандарта понятието „Учебен обект“ е дефинирано като единица – дигитална или не-дигитална, която може да бъде използвана, използвана повторно или реферирана посредством технологично поддържано обучение. Използваните в стандарта атрибути за описване на учебен обект, включват: тип на обекта; автор; притежател; формат и педагогически параметри, като преподаване и стил на взаимодействието. LOM обхваща йерархия от елементи. На първото ниво са девет категории, всяка от които съдържа под-елементи. Тези под-елементи могат да бъдат прости елементи, които съдържат данни или могат да бъдат агрегатни елементи, които на свой ред съдържат под-елементи. Когато се използва LOM не е задължително всички елементи от стандарта да бъдат използвани. Възможно е също елементи от други схеми с метаданни да бъдат интегрирани в LOM. Деветте категории в стандарта LOM са:

- Основна: описва учебния обект като цяло
- Жизнен цикъл: история и настоящ статус на учебния обект; допринесли за създаването на учебния обект
- Мета-метаданни: информация относно метаданните, описващи учебния обект
- Технически: технически изисквания и характеристики на учебния обект
- Образователни: образователни и педагогически характеристики
- Права: Права за интелектуална собственост
- Връзки: връзки между учебния обект и други свързани обекти
- Анотация: коментари относно образователната употреба на учебния обект
- Класификация: класификационни схеми, използвани за описание на различни характеристики на учебния обект [29].

Метаданните в дигиталната библиотека използват разработения в рамките на проекта Share.TEC модел на метаданни "Common Metadata Model" (CMM), базиран на стандарта LOM. Моделът за метаданни CMM включва разширен и допълнен набор от метаданни, специално създадени за описване на педагогическите характеристики на ресурсите [85]. CMM е тясно свързан с онтологията TEO, която служи като концептуална основа за описание на елементите и обектите в областта. Сред метаданните, включени в CMM са: Тип на Интерактивност, Режим на заетост Тип на Дигиталното съдържание, Дидактична стратегия, Ниво на колаборация и др.

Адаптивното търсене е такова търсене, което води до резултати, съобразени с индивидуалните предпочитания на потребителя. Тези предпочитания могат да бъдат заявени директно от потребителя или извлечени индиректно от системата. В настоящата работа е предложено решение, при което адаптивността се постига автоматично, чрез разработената адаптивна и персонализираща логика. За тази цел е използван методът „адаптиране на заявката", разгледан в секция "Адаптивно търсене – дефиниции и характеристики" към Глава 2 от настоящия труд. Адаптирането на заявката се реализира, като оригиналната заявка на потребителя се обогатява с атрибути на избрани елементи метаданни от модела CMM.

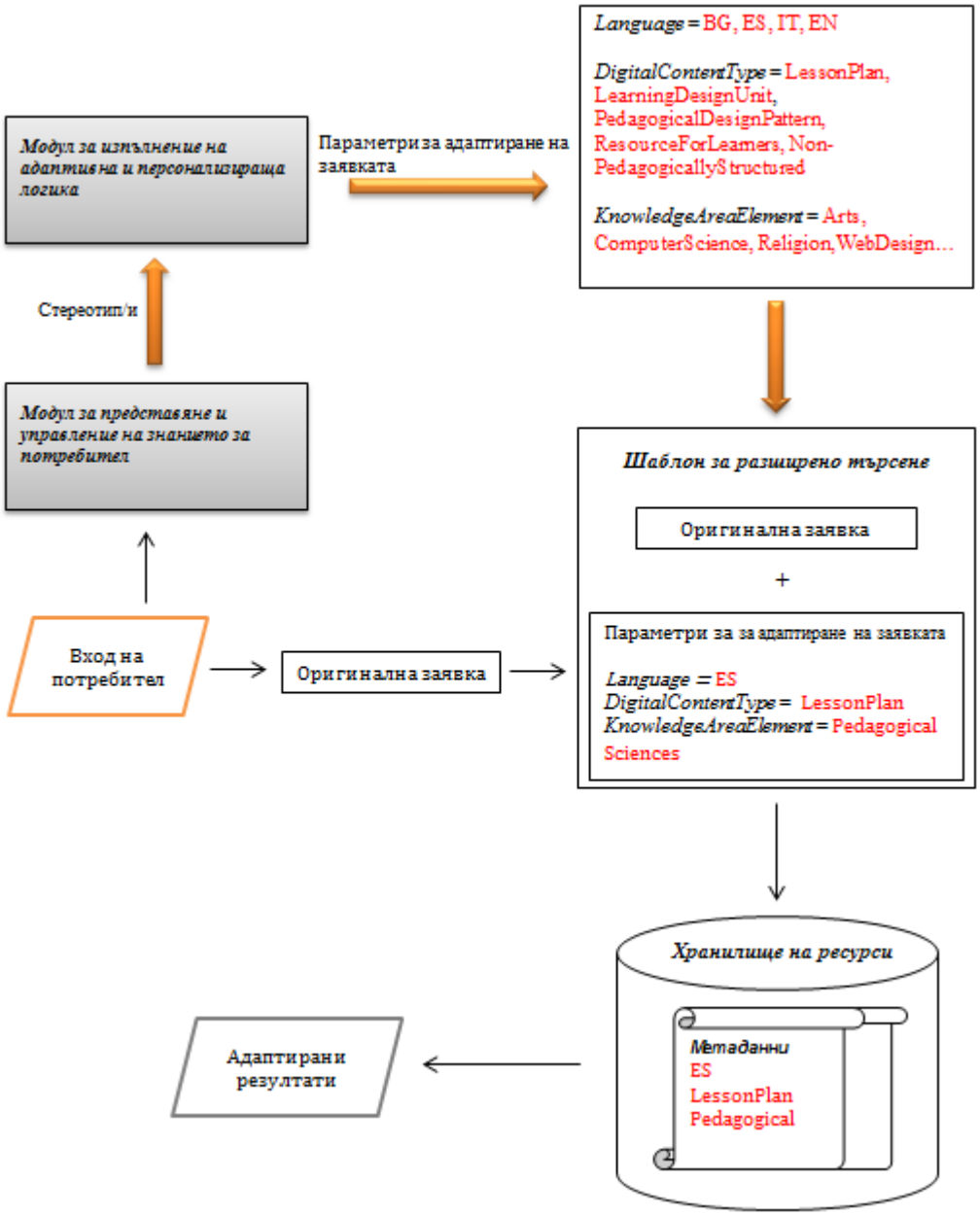
Процесът на адаптиране на заявката за търсене може да бъде представен по следния начин:

1. Потребителят се регистрира в системата
2. Потребителя въвежда заявка Q в системата
3. Потребителят се класифицира към един или повече стереотипа и вследствие на това, се извличат стойности на метаданни за разширяване на заявката посредством Модул за изпълнение на адаптивна и персонализираща логика
4. Зарежда се празен шаблон, представляващ формата за разширено търсене
5. Полетата на шаблона се попълват динамично с изведените от Модула за изпълнение на Адаптивна и персонализираща логика стойности на метаданни и с оригинално въведените от потребителя термини за търсене. При това оригиналната заявка Q се трансформира в адаптирана заявка Q'

6. Q‘ се изпълнява в хранилището с ресурси, водейки до множество от резултати F

7. F се представя на потребителя

Адаптираната по този начин заявка е подобна на функционалността разширено търсене в системата, която осигурява на потребителите шаблон с възможност за избор на атрибути(стойности) за метаданни. Избраните от потребителя атрибутисе използват от машината за търсене за селектиране на тези записи с метаданни, които съвпадат с желаните стойности. По този начин, потребителят има възможност да зададе ръчно ограничение върху избора на изведени информационни ресурси. Предложеното решение в настоящата работа извършва автоматично ограничение върху избора на изформационни ресурси, като задава конкретни стойности на избрани елементи метаданни и по този начин реализира адаптивност на търсенето. Параметрите (стойностите) за метаданните са получени вследствие на изпълнение на адаптивна и персонализираща логика в *Модул за изпълнение на адаптивна и персонализираща логика*. Фиг.16 представя процеса на адаптивно търсене в системата.



Фиг.16 Процес на адаптивно търсене в системата

Извлечените от *Модул за изпълнение на адаптивна и персонализираща логика* атрибутиса за следните елементите от метаданните в CMM:

- Language – попълва се с атрибутиот тип *„Език“* от Бейсов модел
- Digital Content Type – попълва се с атрибутиот тип *„Тип на дигиталното съдържание“* от Бейсов модел
- KnowledgeAreaElement – попълва се с атрибутиот тип *„Област на знание“* от Бейсов модел

Таблица 11 показва мястото на елементите Language, Digital Content Type и KnowledgeAreaElement в модела за метаданни CMM.

Таблица 11 Елементите *Language, DigitalContentType* и *KnowledgeAreaElement* в модела за метаданни CMM

Име	Описание	Възможни стойности	Елемент от CMM схема за метаданни
Език *Language*	Основният човешки език или езици, използвани в рамките на този ресурс.	Инстанциите на класа *Language* от TEO: EN, IT, ES…	Категория 1 от CMM „Основна“
Тип на Дигиталното съдържание *Digital Content Type*	В него се посочва конкретния вид на цифрово съдържание, като се отчита равнището на педагогическата структуриране, нивото на формализация и крайния потребител за който е предназначено.	Под-класовете на класа *DigitalContentType* от TEO: Non-Pedagogically Structured Resource For Learners Pedagogical Design pattern Lesson Plan Learning Design Unit	Категория 10 от CMM „Педагогически метаданни“
Област на знание *KnowlegdeArea Element*	Съдържа информация за областта на знание. Представя се в OMM, като се импортира класът *KnowledgeArea* от онтологията TEO в Категория 9 от LOM „Класификация“	Инстанциите на класа *KnowlegdeAreaelement* от TEO:	Категория 9 от OMM „Класификация“

Пълния CMM модел за метаданни и свързаността му с онтологията TEO са представени в [85].

Изводи и заключения:

В настоящата глава е разгледана архитектурата на Функционален модул за адаптивно търсене, като са представени и неговите компоненти. Представена е базовата схема на адаптивна и персонализираща логика за реализиране на адаптивно търсене в архитектурата. Разгледани са модулите и подмодулите от архитектурата, като съществено значение е отделено на централното звено от архитектурата - Бейсов модел за адаптиране и персонализиране. Представена е както неговата структура, така и логиката му на действие. Показана е взаимовръзката между елементите от

архитектурата и тяхното взаимодействие с цел изпълнение на поставената задача пред архитектурата, а именно - реализиране на адаптивно търсене в дигитална библиотека. Разгледан е моделът за метаданни, реализиран в системата, с чиято помощ се осъществява адаптивното търсене. Представени са разработените „физически“ стереотипи за системата Share.TEC. Описанията са придружени с примери, схеми, диаграми и таблици.

Глава 5. РЕАЛИЗАЦИЯ НА МОДУЛ ЗА ИЗПЪЛНЕНИЕ НА АДАПТИВНА И ПЕРСОНАЛИЗИРАЩА ЛОГИКА ПОСРЕДСТВОМ ИЗВОД В БЕЙСОВА МРЕЖА

С течение на времето броя на средите доставящи знания расте и набира популярност в различни области като бизнес среди, наука, обучение и др. Приложението на този тип среди се оказа особено ползотворно за процеса на обучение.

Използването на дигитална библиотека расте с течение на времето, поради това, че съхраняваната и доставяна информация е надеждна, структурирана и готова за употреба. Дигиталната библиотека все повече се явява съществена част от процеса на обучение, като източник на учебни ресурси за обучаемите. В редица случаи обучаемите не притежават нужното знание и умение да търсят информация в дигитална библиотека, тъй като техния опит в информационното пространство е свързан предимно с търсене по ключови думи в търсещи машини. Това налага внедряването на такава функционалност, която максимално да опрости процеса на търсене и намиране на информация в дигитална библиотека.

В рамките на една дигитална библиотека се осъществява дейност по управление на информационни ресурси; индексиране и анотиране; сортиране; класифициране; добавяне на метаданни към обектите и извличане на информация. Колекциите в дигиталната библиотека се създават, управляват и стават достъпни по такъв начин, че да бъдат удобни за ползване от определени общности. Дигиталната библиотека допълва дигиталните архиви и инициативите за опазване и дългосрочно съхранение на информационните ресурси. Дигиталната библиотека служи като среда, в която се обединяват колекции, услуги и хора, за да се поддържа пълният цикъл от разпространение, използване и съхранение на данни, информация и знания [99]. Ресурсите в една дигитална библиотека трябва да бъдат систематизирани и класифицирани и обектите да са анотирани с общ формат на метаданни, който да съответства на приетите стандарти за цифрови библиотеки.

С дигитализирането на библиотеките на преден план излиза въпросът за предлагане на персонализиращи и адаптивни функционалности, съобразени с потребителите. От съществено значение за потребителите на една дигитална библиотека е достъпа до нейните ресурси. Това води до развитие и реализиране на

набор от възможности за търсене в дигиталните библиотеки, сред които са: семантично-базираното търсене, многокритерийно търсене, контекстно-базирано търсене, адаптивно търсене и др. Тези функционалности често са съпътствани с адаптивно представяне, система за препоръчване и др.

5.1 Системата Share.TEC

В рамките на проекта Share.TEC се изгражда мултимедийна дигитална библиотека, която да служи за обучение на учители, като дава възможност за обмен на информационни ресурси (учебни материали) и опит между участниците [93, 94 96]. Значима част на системата е порталът, предлагащ услуги, осигуряващи персонализиран достъп до широк спектър от учебно съдържание, предназначено за подготовка на учители.

В рамките на проекта Share.TEC е разработена специална онтология, наречена Онтология за обучение на учители – Teacher Education Ontology (TEO) . Целта на онтологията е да осигури устойчив, гъвкав и мощен начин за класифициране на учебни ресурси в централното хранилище на системата. Чрез онтологията TEO се цели поддържане на педагогически обусловено съдържание; представяне на потребителя и компетенциите му; предлагане на многоезиково, мултикултурно и персонализирано взаимодействие с адаптивни приложения и поддържане на йерархично и динамично търсене.

Всички метаданни в хранилището следват формата на общия модел за метаданни (CMM), който е базиран на формата на стандартния модел за описание на учебни обекти Learning Object Model (LOM) [29]. Общият модел на метаданни представлява разширение на LOM, в което са добавени секции за педагогическите характеристики на информационното съдържание [93, 96, 97]. Елементи от CMM модела се използват в фòрмата за разширено търсене в портала Share.TEC, който дава възможност на потребителите да задават изрично стойности на метаданните при търсене в системата [7, 8]. По този начин търсенето от различни потребители, които имат различни нагласи и представи, е поставено в рамките на споделена семантика.

Общността от потребители на системата включва хора с различни квалификации, изисквания и очаквания. Поради това от съществено значение за

системата Share.TEC е възможността за доставяне на адаптивност и персонализация. В системата са идентифицирани три ключови взаимосвързани елемента: адаптивност, моделиране на потребителя и потребителски интерфейс [7, 8, 97].

Основните потребителски дейности, свързани с адаптивността в цифровите библиотеки, са настройките на предпочитанията и търсенето на цифрово съдържание.

Системата за препоръчване в Share.TEC предлага множество от функции, подпомагащи потребителят да намери най-подходящите ресурси, съобразно модела му и историята на неговите действия в системата [94]. Системата за препоръчване използва данните от потребителския модел, за да адаптира функционалността и поведението на системата. Системата за препоръчване поддържа и някои социални функции, като намирането на сходни потребители на базата на Кръгове на сходство [97].

Моделът на потребителя в системата Share.TEC се състои от два основни компонента: (1) явно зададен модел, който включва персонална информация за потребителя, като данни за неговия език, ниво на експертност и др, и (2) косвена информация за потребителя, базирана на неговото използване на системата, включваща историята на търсене, анотиране и разглеждане на ресурси и др [7, 97]. Освен тези данни системата използва и статистическа информация относно начините за използването ѝ от всички потребители, и въз основа на тази информация, могат да се изчислят близостта между ресурси, потребители и други обекти. Моделът на потребителя е тясно интегриран с онтологията TEO, която се съхранява в цифровата библиотека. Специфични данни за потребителя като професионална област, област на експертност, контекст на работа, които са свързани със съответните клонове на онтологията, могат да приемат стойности само според съответните възли от онтологията. Когато потребителят редактира модела си, той избира от списък с дефинирани стойности, изведени от онтологията [97].

Търсенето е една от централните функционалности на Share.TEC портала. То играе важна роля в адаптивността на системата. В Share.TEC системата за препоръчване и машината за търсене използват общия модел на метаданни CMM, онтологията за обучение на учители TEO, както и косвено и пряко зададената информация от модела на потребителя [94, 97].

5.2 Реализация на Модул за изпълнение на адаптивна и персонализираща логика

Среди за проектиране и разработка

В последните години се наблюдава нарастване на броя инструменти и софтуерни среди за разработване и управление на Бейсови мрежи. Сред тези среди са GeNIe & SMILE, HUGIN и JAVABAYES, които позволяват лесна дефиниция на Бейсови модели и безпроблемно интегриране на алгоритми за обновяване на мрежата.

В настоящата работа е избрана средата за графична разработка на модели на решенията и Бейсови мрежи - GeNIe и библиотеката от класове SMILE. Двете са разработени от Decision Systems Laboratory към Университета в Питсбърг и предоставени на широката общественост през 1998 година. GeNIe и SMILE се разпространяват като софтуер с отворен код и нямат ограничения за тяхната употреба. Към момента лиценза на GeNIe и SMILE е закупен от BayesFusion LLC (през 2015г.), които продължават традицията на Университета в Питсбърг и предоставят софтуера безплатно за академични цели [14].

GeNIe служи за графичен потребителски интерфейс към библиотеката SMILE. Предназначен е за създаване на теоретични модели на решението и извършване на диагноза. Главните функции на GeNIe са както следва:

- Графичен редактор за създаване и модифициране на модели
- Поддържа следните типове възли: Общ тип, NoisyOR/MAX и NoisyAND/MIN
- Съвместимост с друг софтуер от същия тип. Поддържа файловите формати на Netica© version 1.06, Hugin© version 3.1.1, ERGO и Microsoft© Bayes Networks version 1.0.1.7
- Пълна интеграция с Microsoft Excel
- Поддръжка на диагностично управление на събития

SMILE (Structural Modeling, Inference, and Learning Engine) е напълно независима от платформата библиотека от C++ класове, която реализира графични вероятностни модели, както и теоретични модели на решението и е подходяща за директно внедряване в интелигентни системи. SMILE е пусната в употреба през 1997 г. и оттогава намира широко приложение. Класове, дефинирани в SMILE дават

възможност за създаване, редактиране, запазване и зареждане на графични модели и за тяхното използване за извършване на вероятностен извод. За достъп до библиотеката от редица други програмни езици, съществуват wrappers за ActiveX, Java и .NET. Основните функции на SMILE са както следва:

- Платформено независим; съществуват версии за Windows, Unix (Solaris), Linux, Mac, PocketPC, и др.
- Съществуват Wrapper за употреба с .NET. Съвместим с всички .NET езици.
- Може да се използва за създаване на уеб-базирани приложения с Бейсови мрежи.
- Притежава пълна и обширна документация.

Към 2015г, комбинацията от SMILE и GeNIe има няколко хиляди потребители по целия свят. Приложенията, базирани на софтуера, съществуват както в областта на индустрията, така и се използват за академични изследвания и проекти [14]. Всички гореизброени характеристики на средата GeNIe, в комбинация с библиотеката на SMILE, я правят изключително подходяща за целите на настоящия труд.

Основни алгоритми в Модела

Основните алгоритми са: *Алгоритъм за съставяне на Бейсова мрежа* и *Алгоритъм за получаване на данни за атрибути за адаптиране на заявката според избран стереотип на потребител.*

- Алгоритъм за съставяне на Бейсова мрежа

В настоящата секция се разглежда *Алгоритъм за съставяне на Бейсова мрежа*, като е представено описанието на алгоритъма, придружено с таблици и фигури, както е разгледана и софтуерната му реализация.

Описание на алгоритъма

Представеният алгоритъмът се състои от набор от последователни стъпки, чрез които се създава качествената и количествената част от мрежата. В първа стъпка на алгоритъма (концептуализация) се избира множество от подходящи променливи $\{X_1, X_2,....X_n\}$, такова че X_1 няма предшественици, а предшествениците на всеки друг възел X_i са измежду $\{X_1, X_2,...X_{i-1}\}$. Във втора стъпка от алгоритъма се въвежда наредба над елементите в това множество. Третата стъпка е свързана с последователно добавяне на възлите към мрежата. В нея за всеки елемент от множеството X_i се осъществяват следните подстъпки: добавя се нов възел в мрежата, който се обозначава с X_i; избират се предшествениците на възела $Pred(X_i)$, които да бъдат минималното множество от вече съществуващи възли, такива че X_i да бъде условно независима от всички вече добавени възли, които не са в $Pred(X_i)$; дефинират се вероятностите – априорни или условни [104].

Следва описание на стъпките от алгоритъма.

1. Концептуализация, при която се определят променливите на БМ

Мрежата на Бейс се състой от бинарни променливи. В първа стъпка от алгоритъма за съставяне на БМ задаваме (определяме) променливите, които ще съставят БМ. Представени са част от тези променливи. Пълен списък на всички променливи е представен в Приложение 2 на настощия труд.

ES = „Испански“

BG = „Български“

LearningDesignUnit = „Учебна единица“

LessonPlan = „LessonPlan“

PedagogicalDesignPattern = „Педагогически модел на структурата“

Need0 = „Управление на курс в платформа за електронно обучение на английски език“

Need1 = „Подготвяне на присъствен урок за испански учители в област Педагогическа наука“

Need5 = „Подготвяне на онлайн активност за ирландски обучаеми“

Stereotype1 = „Испански Обучител на обучители във Facultad de Educación (Máster y Doctorado)“

Stereotype2 = „Испански Обучител на обучители във Факултет за подготовка на учители (Facultad de Formación del Profesorado)“

Stereotype3: = „Ирландски Обучител на обучители на учители от средни училища“

2. Наредба на мрежата

Във втора стъпка от алгоритъма нареждаме променливите, дефинирани в първата стъпка. За тази цел създаваме три списъка, във всеки от който поставяме даден тип от променливи въз основа на признака, дали те са причина или следствие за/от настъпване на събития.

Независимите от причинно-следствена гледна точка са променливите на стереотипите – от Stereotype1 до Stereotype51 включително. Тях ги поставяме в първия списък. Те са предпоставка за настъпването на събитията, представени от останалите променливи. Променливите ES,…,PublishingDesign са крайните събития (следствия), тях ги поставяме в третия списък. Променливите Need0,…,Need39 са междинни – те са причина за настъпването на някои събития и следствие от други, затова ги поставяме във втория списък. В резултат на подреждането се получават списъците:

Първи списък, съдържащ променливите на стереотипите: {Stereotype1,...,Stereotype51};

Втори списък съдържащ променливите на нуждите: {Need0,…,Need39};

Трети списък, съдържащ променливите на атрибутите: {ES, EN,…,PublishingDesign}

3. Последователно добавяне на възли към мрежата

В трета стъпка от алгоритъма добавяме възли към мрежата и ги свързваме с техните предшественици (ако имат такива) посредством дъги. Тази стъпка се състои от три подстъпки: добавяне на възли от най-горния слой; добавяне на възли от средния слой и добавяне на възли от най-долния слой.

Нека S е множеството на всички стереотипи, N е множеството на всички нужди, а A е множеството на всички атрибути. Тогава S_i ще бъде елемент от множеството S, N_i елемент от множеството N, а A_i елемент от множеството A.

След добавяне на възел N_i се избират предшествениците на възела $Pred(N_i)$, които са част от множеството S, такива че N_i да бъде условно независима от всички вече добавени възли, които не са в $Pred(N_i)$. След добавяне на възел A_i се избират

предшествениците на възела Pred(A_i), които са част от множеството N, такива че A_i да бъде условно независима от всички вече добавени възли, които не са в Pred(A_i).

В първа подстъпка от алгоритъма добавяме възлите от най-горния слой на Модела. За тази цел, използваме променливите от първия списък, които са независими от всички останали променливи.

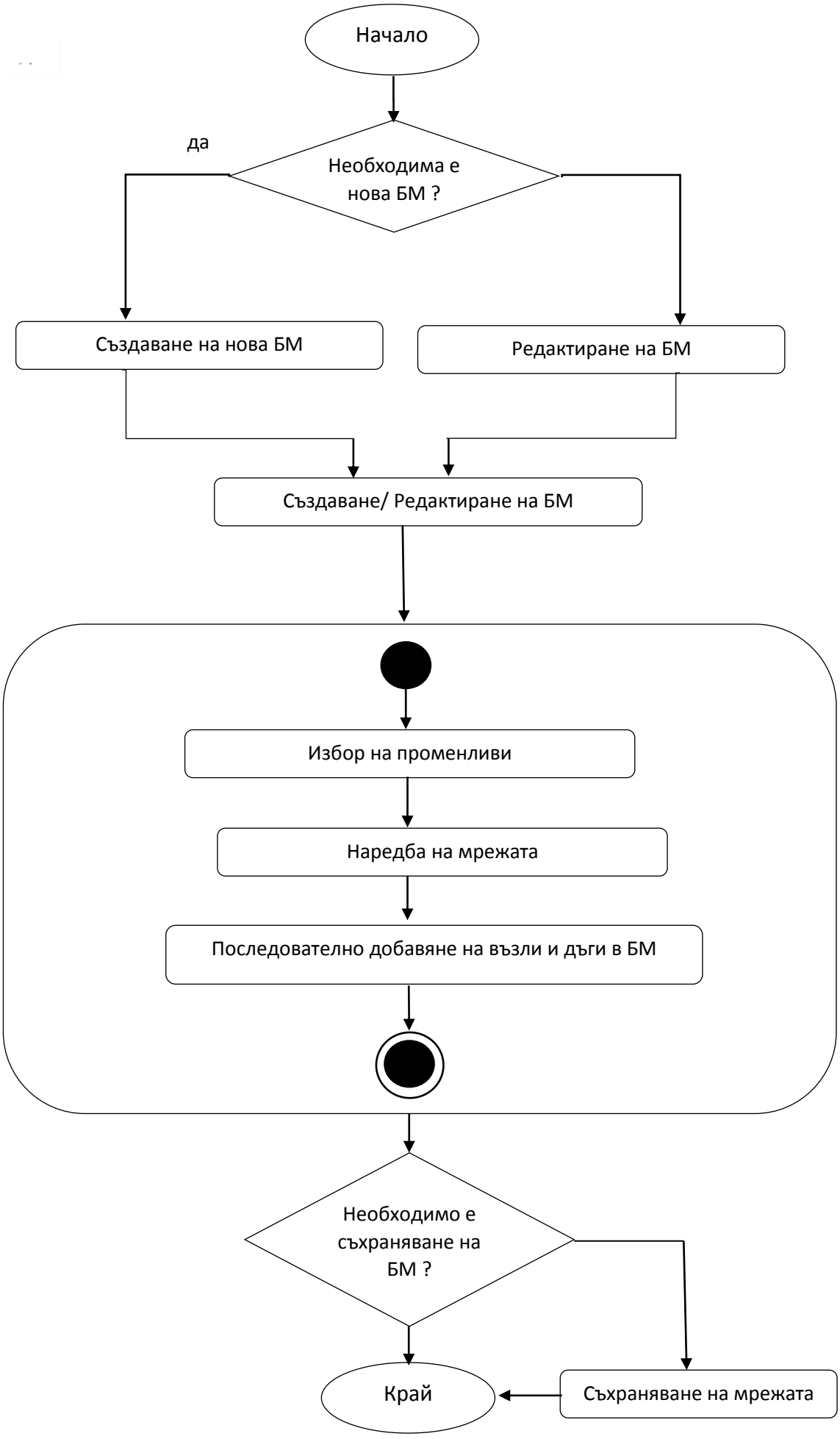
Създаваме възел Stereotype1 „Испански Обучител на обучители във Facultad de Educación“ като възел от тип DSL_cpt (възел от общ тип). Той е независим от всички останали възли и е достатъчно да зададем само неговите априорни вероятности, със стойност 0,02 за *истина* и 0,08 за *лъжа*. Продължаваме с последователно добавяне на всички променливи от първия списък, като създаваме възлите от Stereotype2 до Stereotype51, подобно на възела Stereotype1. С това приключваме създаването на възлите от най-горния слой.

Във втората подстъпка създаваме възлите от средния слой. За тази цел, използваме променливите от втория списък, които са зависими от преките си предшественици. Създаваме възел Need0 „Управление на курс в платформа за електронно обучение на английски език“, като възел от тип DSL_ noisyadder. Възелът Need0 е зависим от предшествениците си - възлите Stereotype4 „Ирландски Обучител на обучители зает с он-лайн обучение на учители“, Stereotype14 „Ирландски университетски преподавател на про-гимназиални учители“, Stereotype15 „Ирландски университетски преподавател на гимназиални учители по музика“, Stereotype16 „Ирландски университетски преподавател на гимназиални учители по религия“ и LEAK. Създаваме дъги между възела Need0 и възлите Stereotype4, Stereotype14, Stereotype15, Stereotype16 (възела LEAK не се задава графично, защото той съществува по подразбиране в GeNIe). Продължаваме с добавяне последователно на всички променливи от втория списък, при което създаваме възлите от Need1 до Need39, подобно на възела Need0. С това приключваме създаването на възлите от средния слой на Модела.

Трета подстъпка се състои в създаване на възлите най-долния слой на Модела. За тази цел, използваме променливите от третия списък, които са зависими от предшествениците си от средния слой на Модела. Създаваме възела LearningDesignUnit „Учебна единица“ като възел от тип DSL_ noisyadder. Той е зависим от своите

предшественици: Need0 „Управление на курс в платформа за електронно обучение на английски език“, Need12 „Управление на курс в платформата за електронно обучение на български език“, Need14 „Управление на курс в платформата за електронно обучение на испански език“, Need20 „Управление на курс в платформа за електронно обучение на италиански език“, Need32 „Разработване на курсове за електронно обучение на различен език“ и LEAK. Създаваме дъги между възела LearningDesignUnit и неговите предшественици. Продължаваме като създаваме всички възли от най-долния слой на Модела, подобно на възела LearningDesignUnit. С това приключваме създаването на възлите от най-долния слой на Модела.

В последната стъпка от алгоритъма се запазват направените промени. С това алгоритъмът приключва. Описаният алгоритъм е представен схематично на Фиг.17.


Начало
да
Необходима е нова БМ ?
Създаване на нова БМ
Редактиране на БМ
Създаване/ Редактиране на БМ
Избор на променливи
Наредба на мрежата
Последователно добавяне на възли и дъги в БМ
Необходимо е съхраняване на БМ ?
Край
Съхраняване на мрежата

Фиг.17 Схема на алгоритъм за съставяне на Бейсова мрежа

Таблица 12 показва част от релациите между променливите и техните предшественици в Модела. Посоката на дъгите е StereotypeX → NeedX → AttributeX. Пълен списък с всички релации е представен в Приложение 5 от настоящия труд.

Таблица 12 Релации между променливите в Модела

Възел на Стереотип	Възел на Нужда	Възел на Атрибут
Stereotype1: ES UCM TE	Need1	ES, LessonPlan, PedagogicalSciences
	Need2	ES, PedagogicalDesignPattern, Non-PedagogicallyStructured
Stereotype2: ES USC TE	Need2	ES, PedagogicalDesignPattern, Non-PedagogicallyStructured
	Need3	ES, LessonPlan
Stereotype3: IR SST TE	Need4	EN, LessonPlan
	Need5	EN, PedagogicalDesignPattern, Non-PedagogicallyStructured
Stereotype4: IR OTT TE	Need0	EN, LearningDesignUnit
	Need5	EN, PedagogicalDesignPattern, Non-PedagogicallyStructured
Stereotype5: IT SSIS TE	Need6	IT, LessonPlan
	Need7	IT, EN, PedagogicalDesignPattern, Non-PedagogicallyStructured, PedagogicalScience
Stereotype6: IT SF TE	Need7	IT, EN, PedagogicalDesignPattern, Non-PedagogicallyStructured, PedagogicalScience
	Need8	IT, LessonPlan, PedagogicalScience
Stereotype7: IT SM TE	Need10	IT, LessonPlan, TeacherTraining-PhisicalTraining

Реализация на алгоритъма

За софтуерната реализация на описания е използвана графичната среда за разработка GeNIe, представена в секция „Среди за проектиране и разработка“ към Глава 6 от настоящия труд.

Софтуерният код за създаване на възела Stereotype1 и задаване на априорните му вероятности (истина = 0.2 и лъжа = 0.8) е показан по-долу. Възелът е създаден като възел от общ тип.

```
<cpt id="Stereotype1">
      <state id="True" />
      <state id="False" />
      <probabilities>0.02 0.08</probabilities>
</cpt>
```

Листинг 1: Софтуерен код на възел Stereotype1

Софтуерният код за създаване на възела Need1, като възел от тип NoisyAdder е показан по-долу. Предшествениците на възела Need14 са три, изрично зададените Stereotype12 и Stereotype13 и подразбиращия се възел LEAK. Тежестите на дъгите са weights = 1 с всички предшественици.

```
<noisyadder id="Need14">
      <state id="True" />
      <state id="False" />
      <parents>Stereotype12 Stereotype13</parents>
      <dstates>1 1 1</dstates>
      <weights>1 1 1</weights>
      <parameters>1 0 0 1 1 0 0 1 0 1</parameters>
</noisyadder>
```

Листинг 2: Софтуерен код на възел Need14

Софтуерният код за създаване на възела EducationalManagement като възел от тип NoisyOR е показан по-долу. Зададени са предшестващи възли Need26 и Need28 и тежести на дъгите към тях със стойност weights = 1.

```
<noisyadder id="EducationalManagement">
      <state id="True" />
      <state id="False" />
      <parents>Need26 Need28</parents>
      <dstates>1 1 1</dstates>
      <weights>1 1 1</weights>
      <parameters>1 0 0 1 1 0 0 1 0 1</parameters>
```

```
</noisyadder>
```

Листинг 3: Софтуерен код на възел EducationalManagement

Пълният софтуерен код на Бейсовата мрежа е представен в Приложение 1 от настоящия труд.

Тестване на алгоритъма

Извършени са серии от тестове, които доказаха правилното функциониране на алгоритъма и неговата способност да изпълни заложените изисквания. Тестовете се проведоха от разработчици.

- Алгоритъм за получаване на данни за атрибути за адаптиране на заявката в дигитална библиотека според избран стереотип на потребител

Описание на алгоритъма

Представеният алгоритъм е процедура, чрез която се извеждат атрибути за адаптиране на заявката. Алгоритъмът е базиран на извод в Бейсова мрежа, който се извършва при задаване на свидетелсва на избрани възли от Модела и обновяване на вероятностите на останалите възли. Алгоритъмът работи с входни данни за стереотипи, който задават какви да са поставените доказателства. Резултата от изпълнението на алгоритъма е връщането на списък със стойности на атрибути. Следва описание на стъпките от алгоритъма.

Стъпка 1: Зареждане на мрежата

```
DSL_network theNet;
theNet.ReadFile("BM.xdsl");
```

Стъпка 2*:* Подаване на стойност **s** за стереотип на потребителя

Стъпка 3: Изпълнение на извод над мрежата, при зададен стереотипа на потребителя, чрез извикване на функция:

```
InfereceWithBayesNet(s);
```

Стъпка 3.1: Задаване на доказателство *истина* на стереотипа s на потребителя и *лъжа* за всички останали стереотипи в мрежата

theNet.GetNode(s)->Value()->SetEvidence(0) - задава се доказателство *истина* на стереотипа **s**

theNet.GetNode(v)->Value()->SetEvidence(1) – задава се доказателство *лъжа* на всички стереотипи, с изключение на стереотипа **s**

Стъпка 3.2: Обновяване на мрежата

theNet.UpdateBeliefs();

Стъпка 4: Идентифициране на атрибутите за адаптиране на заявка, при зададен стереотип **s** на потребител и след изпълнение на извод

Стъпка 4.1: Получаване на резултатни стойности за вероятностите на атрибутите

```
int attr1 = theNet.FindNode("X");
theCoordinates.LinkTo(*theNet.GetNode(attr1)->Value());
theCoordinates[0] = 0;
theCoordinates.GoToCurrentPosition();
double P = theCoordinates.UncheckedValue();
```

Стъпка 4.2: Избор на атрибути, които изпълняват условието да бъдат селектирани за адаптиране на заявката

```
if (P >= 0.01)
```

Стъпка 4.3: Запис на избраните атрибути

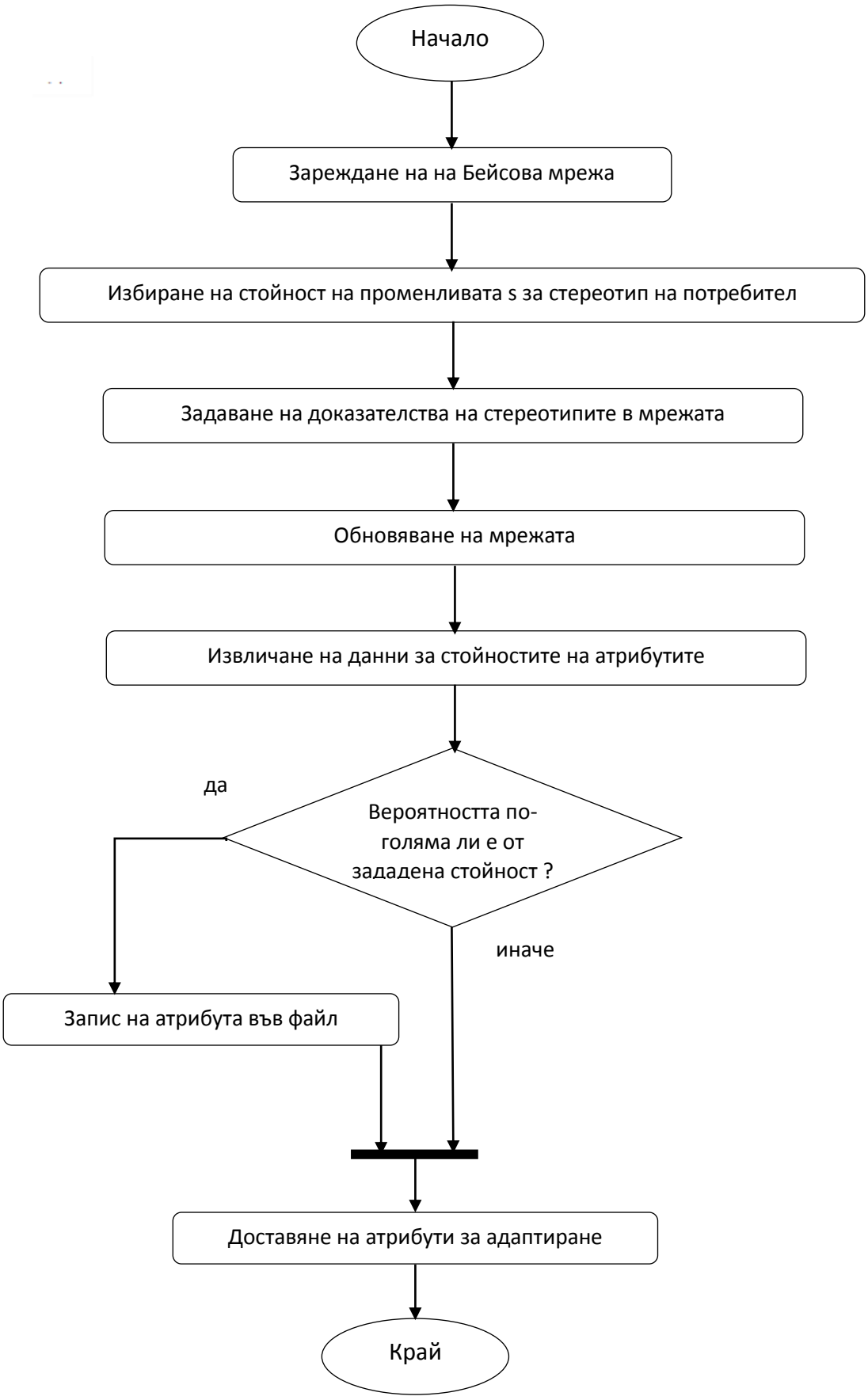
```
myfile.open("f.txt");
myfile << "X";
myfile.close()
```

Стъпка 5: Изчистване на зададените в Стъпка 3.1 доказателства на стереотипите

```
theNet.GetNode(x)->Value()->ClearEvidence();
```

Стъпка 6: Край на алгоритъма

Описаният алгоритъм е представен схематично на Фиг.18.


Начало
Зареждане на на Бейсова мрежа
Избиране на стойност на променливата s за стереотип на потребител
Задаване на доказателства на стереотипите в мрежата
Обновяване на мрежата
Извличане на данни за стойностите на атрибутите
Вероятността по-голяма ли е от зададена стойност ?
да
иначе
Запис на атрибута във файл
Доставяне на атрибути за адаптиране
Край

Фиг.18 Схема на алгоритъм за получаване на данни за атрибути за адаптиране на заявката в дигитална библиотека според избран стереотип на потребител

Реализация на алгоритъма

За софтуерната реализация на описания алгоритъм е използвана C++ библиотеката на SMILE, представена в секция „Среди за проектиране и разработка“ към Глава 6.

Основните класове, реализиращи алгоритъма са: DSL network, DLS sysCoordinates, DSL node, DSL nodeValue и DSL Dmatrix [14]. Тези класове, заедно с основните им методи са представени в таблиците по-долу.

Таблица 13 Основни методи на класа DSL network

Клас: DSL network Този клас съдържа повечето от функционалността на библиотеката. Мрежата съдържа възли. Всеки възел има зададено уникален идентификатор, който може да се използва за достъп до този възел. Идентификатора е целочислена стойност, която се запазва за времето на живот на мрежата.	
Операции:	***Описание:***
int AddNode(int thisType, char *thisId)	Добавя възел от тип thisType към мрежата. thisId е индикатор за идентификатора на новия възел. Ако thisId е NULL, ще бъде създаден уникален идентификатор. thisId показва идентификатора на новия възел. Ако thisId е NULL, тогава ще бъде генериран идентификатор за възела. При грешка се връщат: DSL_OUT_OF_MEMORY и DSL_OUT_OF_RANGE
int AddArc(int theParent, int theChild)	Добавя дъга между theParent и theChild. За да е успешен метода, трябва да са изпълнени следните условия: theParent и theChild да са валидни идентификатори, дъгата да не е предпоставка за създаване на цикъл в мрежата, да не съществува такава дъга и theParent да бъде от подходящ дефиниционен тип (не всички дефиниционни типове са съвместими). При грешка се връща: DSL_OUT_OF_RANGE
int RemoveArc(int theParent, int theChild)	Премахва дъгата от theParent към theChild. При грешка връща DSL_OUT_OF_RANGE
int DeleteNode(int thisNode)	Изтрива thisNode от мрежата, като thisNode е идентификатора, създаден при създаването

	на възела.
int FindNode(char *withThisID)	Този метод връща идентификатора на възела withThisID. При грешка връща: DSL_OUT_OF_RANGE
int GetFirstNode(void)	Този метод връща идентификатора на първия възел в мрежата. Може да се използва в комбинация с метода GetNextNode за обхождане на цялата мрежа.
int GetNextNode(int ofThisNode)	Този метод връща идентификатора на следващия възел ofThisNode
int GetAllNodes(DSL_intArray &here)	Този метод връща идентификаторите на всички възли от мрежата
int NumParents(int ofThisNode)	Връща броя предшественици на ofThisNode
int NumChildren(int ofThisNode)	Връща броя следстващи възли на ofThisNode
void SetDefaultBNAlgorithm(int theAlgorithm)	Задава алгоритъмът, който да бъде използван за обновяване на вярата в бейсовата мрежа. Възможните стойности за алгоритъма са: DSL_ALG_BN_LAURITZEN DSL_ALG_BN_HENRION DSL_ALG_BN_PEARL DSL_ALG_BN_LSAMPLING DSL_ALG_BN_SELFIMPORTANCE DSL_ALG_BN_BACKSAMPLING DSL_ALG_BN_AISSAMPLING DSL_ALG_BN_EPISSAMPLING DSL_ALG_BN_LBP DSL_ALG_BN_LAURITZEN_OLD
int UpdateBeliefs(void)	Този метод ще актуализира мрежата посредством алгоритъма по подразбиране, определен от метода на SetDefaultBNAlgorithm. Само тези възли, които са маркирани като цели ще бъдат актуализирани. Ако няма възли маркирани като цели, всички възли на мрежата ще бъдат актуализирани.
int ReadFile(char *thisFile, int fileType = 0)	Попълва мрежата със съдържанието на thisFile. fileType специфицира формата на файла в съответствие със следните стойности: DSL_DSL_FORMAT (Старият формат на SMILE, преустановен) DSL_ERGO_FORMAT (Старият ERGO формат) DSL_NETICA_FORMAT (Netica© версия 1.06) DSL_HUGIN_FORMAT (Hugin© версия 3.1.1) DSL_INTERCHANGE_FORMAT (Microsoft© Bayes Networks версия 1.0.1.7) DSL_XDSL_FORMAT (Новият препоръчителен формат за SMILE) Ако файловият формат не е зададен, системата ще се отита да познае формата, на базата на файловото разширение.

int WriteFile(char *thisFile, int fileType = 0)	Запазване на мрежата в thisFile в специфицирания формат
DSL_creation &Creation(void)	Връща структурата DSL_creation , която съдържа информация за мрежата

Таблица 14 Основни методи на класа DLS sysCoordinates

Клас: DLS sysCoordinates Този клас осигурява лесна навигация из елементите на обекта DSL_Dmatrix. DSL_Dmatrix е многоизмерна матрица, подходяща за съхранение на таблиците на условните вероятности, както и на постериорните вероятности. За да бъде използван, той трябва първо да бъде свързан към матрицата посредством някой от конструкторинте или методите LinkTo. Класът съдържа DSL_intArray, с толкова елементи, колкото са измеренията на матрицата. Посредством него ние можем да попълним желаната стойност на всяка координата и да достъпим специфицирания елемент на матрицата. Класът също съдържа „плосък“ индекс на елементите от матрицата.	
Операции:	***Описание:***
int GoToCurrentPosition(void)	Обновява стойността на currentIntPosition. Този метод се използва след промяна в съдържанието на currentPosition посредством [] оператор.
void LinkTo(DSL_nodeValue &theLink)	Ако стойността, която подаваме като параметър, съдържа DSL_Dmatrix, то тогава този метод ще ни свърже с нея. Например, ако подаваме DSL_beliefVector то тогава ще се свържем с матрицата която съдържа състоянията на вяра на възела. Ако стойността не притежава матрица, то тогава няма да се свържем към нищо.
void LinkTo(DSL_nodeDefinition &theLink)	Ако дефиницията, която подаваме като параметър, съдържа DSL_Dmatrix, то тогава този метод ще ни свърже с нея. Например, ако подаваме DSL_cpt ще се свържем към матрицата, която съдържа вероятностното разпределение на възела. Ако дефиницията не притежава матрица, то тогава няма да се свържем към нищо.
void LinkTo(DSL_Dmatrix &theLink)	Този метод ни свързва с матрицата, специфицирана в theLink.
double &UncheckedValue(void)	Този метод осигурява достъп до елементите на матрица
int Next(void)	Позволява да се отиде към следващия елемент от матрицата. При грешка връща DSL_OUT_OF_RANGE
int Previous(void)	Позволява да се отиде към предишния елемент от матрицата. При грешка връща DSL_OUT_OF_RANGE

Таблица 15 Основни методи на класа DSL node

Клас: DSL node Този клас задава методи за задаване (промяна) на дефиницията или стойността на възел от мрежата	
Операции:	***Описание:***
int ChangeType(int newType)	Променя дефиниционния тип на възел на зададения в newType Сред дефиниционните типовете са: DSL_CPT, DSL_NOISY_OR, DSL_NOISY_AND
DSL_nodeDefinition *Definition(void)	Връща пойнтер към дефиницията на възела
DSL_nodeInfo &Info(void)	Връща информация за възела, като идентификатор на възела и име на възела
DSL_network *Network(void)	Връща пойнтер към мрежата

Таблица 16 Основни методи на класа DSL nodeValue

Клас: DSL nodeValue	
Операции:	***Описание:***
int ClearEvidence(void)	Премахва зададените доказателства
int SetEvidence(int theEvidence)	Задава доказателство на възел

Таблица 17 Основни методи на класа DSL Dmatrix

Клас: DSL Dmatrix Матрицата е базова структура, която се използва в SMILE за съхранение на вероятности, вярвания и др. Той съдържа методи, които улесняват операциите над възлите в мрежата.	
Операции:	***Описание:***
DSL_Dmatrix(DSL_intArray &theseDimensions)	Конструира матрица с размер и брой измерения, специфицирани в theseDimensions

Следва представяне на части от софтурният код. За неговото реализиране е използвана готовата C++ библиотека на SMILE "Smile.lib", чрез включване на "smile.h"в програмния код.

Главната функция в софтуерния код е представена в Листинг 4.

```
int main()
{
        char *s = "Stereotype44";  //Подава се стереотип
        InfereceWithBayesNet(s);
        ofstream myfile;
        return(DSL_OKAY);
};
```

Листинг 4: Главната функция в софтуерната реализация

Изводът в Бейсовата мрежа се извършва посредством изпълнение на функцията InfereceWithBayesNet().

Главните етапи в изпълнението на кода са:

- Задаване на БМ, над която ще се извършва извод:

```
DSL_network theNet;
theNet.ReadFile("BM.xdsl"); // Прочита се мрежата на Бейс
```

- Задаване на алгоритъм за извод над мрежата (клъстериращ алгоритъм)

```
theNet.SetDefaultBNAlgorithm(DSL_ALG_BN_LAURITZEN);
```

- Задаване на доказателства *лъжа* на всички възли от най-горния слой

```
int ste1 = theNet.FindNode("Stereotype1");
theNet.GetNode(ste1)->Value()->SetEvidence(1);
int ste2 = theNet.FindNode("Stereotype2");
theNet.GetNode(ste2)->Value()->SetEvidence(1);
:
int ste51 = theNet.FindNode("Stereotype51");
theNet.GetNode(ste51)->Value()->SetEvidence(1);
```

- Задаване на доказателство *истина* на избран възел от най-горния слой

```
const char *a = b;
int stereotype = theNet.FindNode(a);
```

```
theNet.GetNode(stereotype)->Value()->SetEvidence(0);
```

- Обновяване на мрежата

```
 theNet.UpdateBeliefs();
```

- Получаване на резултатната стойност на вероятността на всички възли от най-долния слой

```
int attr = theNet.FindNode("IT");

DSL_sysCoordinates theCoordinates;
theCoordinates.LinkTo(*theNet.GetNode(attr)->Value());

theCoordinates[0] = 0; // 0 е индекс на състоянието,  съответстващ на
истина

theCoordinates.GoToCurrentPosition();

double P = theCoordinates.UncheckedValue(); // Взима се стойността на
вероятността на възел "IT"

int attr1 = theNet.FindNode("EN");

theCoordinates.LinkTo(*theNet.GetNode(attr1)->Value());

theCoordinates[0] = 0;

theCoordinates.GoToCurrentPosition();

double P1 = theCoordinates.UncheckedValue();

:

int attr19 = theNet.FindNode("PublishingDesign");

theCoordinates.LinkTo(*theNet.GetNode(attr19)->Value());

theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();

double P19 = theCoordinates.UncheckedValue();
```

- Проверка за изпълнение на условие, за резултатната стойност на вероятността за всички възли от най-долния слой

```
if (P >= 0.01)
```

- Отваряне и запис във файл на възлите от най-долния слой (атрибути), които изпълняват условието

```
ofstream myfile;
myfile.open("last.txt,");
if (P >= 0.01)
      myfile << "IT" << endl;
      myfile.close();
```

- Изчистване на зададените в хода на изпълнение на програмата доказателства

```
theNet.GetNode(stereotype)->Value()->ClearEvidence();
system("pause");
```

Пълният софтуерен код е представен в Приложение 4 от настоящия труд.

Тестване на алгоритъма

Извършени са серии от тестове, които доказаха правилното функциониране на алгоритъма и неговата способност да изпълни заложените изисквания. Тестовете се проведоха от разработчици.

Изводи и заключения:

В настоящата глава е предствена софтуерната реализация на избрани модули от архитектурата, като са изложени основните алгоритми и тяхната реализация. Главата започва с описание на системата Share.TEC, в което са очертани основни характеристики на системата. Представена е избраната среда за разработка с обосновката за нейния избор. Показани са основните класове от средата за разработка, използвани за реализацията на разработените алгоритми. Описанията са придружени със схеми, таблици и листинги със софтуерния код.

Глава 6. ИЗСЛЕДВАНИЯ И ЕКСПЕРИМЕНТИ

Експериментът е средство за доказване на дадена постановка, на базата на конкретни резултати. В настоящата работа е предложено решение за осигуряване на адаптивно търсене чрез използване на онтология. Архитектурата на решението представлява Функционален модул за адаптивно търсене, съставен от набор от главни модули и техни съставни подмодули, които действат съвместно и съгласувано. Настоящата глава е фокусирана върху изследванията и експериментите, проведени на различни етапи от разработката на решението, като получените резултати са използвани за допълване и за усъвършенстване на вече постигнатото.

Целта на проведените експерименти е както намиране на конкретни решения при разработката, така и валидиране на разработеното, и доказване на възможностите му да изпълни първоначално зададените изисквания към него.

Изследванията и експериментите са проведени на три главни етапа, като постигнатите резултати от всеки етап се явяват необходимо условие за началото на следващия етап. Тези етапи са:

1. Валидиране на разработения набор от стереотипи и оценяване на тяхната способност да описват групите от потребители на системата
2. Оценяване на ефективността на предложения метод за класифициране на потребителите към стереотипите
3. Оценяване на ефекта от прилагане на адаптивност при търсене, посредством адаптиране на заявката. В тази група спадат и експериментите, проведени с Модела, имащи за цел да валидират неговите възможности да достави данни (атрибути), в съответствие с първоначалните очаквания и изисквания. Проведени са и набор от експерименти, целящи да оптимизират Модела и да настроят неговите параметри.

Получените данни от различните етапи на изследването са обобщени и анализирани. Следва представяне на отделните етапи на изследването:

1. Първи етап на изследването: Валидиране на разработения набор от стереотипи

Изследванията и експериментите на този етап бяха насочени към откриване дали разработеният набор от стереотипи е способен да представи и обхване потребителите на системата. Друг важен момент от този етап, беше идентифицирането на тези условия, при които да се налага промяна в набора от стереотипи, свързан с добавяне/премахване на стереотип/и от набора.

При разработката на стереотипите следвахме подход, свързан със: (1) събирането на данни за създаване на стереотипите; (2) анализ на събраните данни; (3) дизайн и (4) оценяване и валидиране.

Разработката на стереотипи е свързана с предварителна работа по събиране на данни, относно профила на потенциалните потребители. В настоящата работа, част от предварителната работа беше спестена, тъй като ние имахме преимуществото да работим с вече изградена система, в чийто център се намира разработена онтология. Съществуващите данни за потребителите и данните, закодирани в онтологията TEO, под формата на класове, инстанции и анотации, послужиха като първоначални данни при създаване на стереотипите.

В онтологията TEO са въведени данни за повече от 58 образователни институции и агенции, 15 типа роли, 4 цели, набор от езици и области/предмети на интерес.

- Представените в онтологията институци и агенции (списъкът не е пълен):

Centros de formación de adultos
Centro Territoriale Permanente per l'Istruzione e la Formazione in età adulta
Agenzia Nazionale Sviluppo Autonomia Scolastica
Agenzia Nazionale Sviluppo Autonomia Scolastica
Centro Iniziativa Democratica Insegnanti
Istituto per lo sviluppo della formazione professionale dei lavoratori
Associazione Italiana Formatori
Conservatorio
Instituto Nacional de Empleo
Ministerio de Educación, Política Social y Deporte
Centros de Formación de Enseñanzas deportivas
Департамент за информация и усъвършенстване на учителите
Empresas de formación
Ufficio Scolastico Regionale
Conservatorio
Centro Nacional de Recursos para la Orientación Profesional
Accademia delle belle arti
Scienze della Formazione (II ciclo)
Scuola di Dottorato riguardante tematiche didattiche e psico-pedagogiche

Отдел “Докторанти и следдипломна квалификация" към СУ
Secondary teacher training
Facultad de Educación (Máster y Doctorado)
Scuola di Specializzazione per l'Insegnamento Secondario
Online teacher training
Scienze motorie (II ciclo)
Facultad de Educación
Великотърновски университет
Софийски университет
Training of Secondary School Music Teachers
Шуменски университет
Scienze motorie (I ciclo)
Universidad de Educación a distancia
Training of Secondary School Religion Teachers
Scienze della Formazione (I ciclo)
Primary teacher training
CISCO мрежова академия
Instituto de formación Profesional
Centros de Cualificación Profesional Inicia
Istituto di formazione professionale iniziale
Ente di formazione professionale
Istituto di formazione iniziale e professionale
Ente di formazione a distanza
Government/State Agency
Ministerio de Ciencia e Innovación
Ministerio de Educación, Política Social y Deportes

- Представените в онтологията роли са:

Разработчик на съдържание
Библиотекар
Издател
Доставчик на услуга
Университетски преподавател
Обучител на обучители
Инструктор/ обучител на обучители
Координатор
Директор
Ментор /Въвеждащ наставник
Учител
Стажант-учител
Технически изпълнител
Обучител/ инструктор
Обучаващ наставник

- Целите, зададени в онтологията са: Обучителна цел, Образователна цел, Наблюдателна цел и Визуализационна цел.

- Дефинираните езици включват: английски, испански, италиански и български и др.

Тези първоначални данни, послужиха като изходна точка на изследването, като впоследствие беше проведено допълнително проучване, състоящо се в он-лайн събиране на информация за отделните институции и роли в тях, разговори с преподаватели, целящи да задълбочат и конкретизират събраното знание и запознаване с други примерни системи. Например, събираната информация за учебните заведения целеше да идентифицира: език/ци на провеждане на занятията, застъпените предмети, методите на преподаване (присъствено/дистанционно/смесено), съществуващите роли и техните задължения, използваните платформи за електронно обучение и тяхната функция и т.н.

Следващият етап в работата, след събирането на данните, беше логичното им обединяване в шаблони (стереотипи), способни да групират потребителите въз основа на значими техни прилики. Пряко свързани с дефинирането на стереотипите стояха идентифицирането и задаването на типичните нужди за всяка от стереотипните групи.

Например за дефиниране на стереотипа „Български Инструктор от CISCO мрежова академия“ се запознахме с начина на работа в академията, което включваше разговори с преподаватели и он-лайн проучване на платформите за обучение в академията (сайтът cisco.netacad.com). Това ни позволи да идентифицираме езика на преподаване, ролята на инструктура и темите на преподаване, и оттам да дефинираме нуждите на този стереотип, а именно: (1) Управление на курс в платформата за електронно обучение на български език и (2) Подготвяне на присъствен урок за български курсисти в област компютърни мрежи.

Във фазата на анализ беше осъзнато, че е възможно с течение на времето да има промяна в състава на потребителите, която да доведе до отпадане на част от стереотипите и необходимост от създаване на нови. Тъй като стереотипите са създадени „статично“, а не динамично те остават в системата, независимо дали се „използват“ или не и се изисква администраторска намеса за тяхната промяна (добавяне или премахване). Поради това е запланувано провеждането на периодични тестове и изследвания, които да установят дали има съществени изменения в състава на потребителите, изискващи предефиниране на набора от стереотипи.

Въз основа на предварителните проучвания и събраните данни бяха дефинирани 50 (петдесет стереотипа). На по късен етап от изследването попаднахме на потребители, който не могат да бъдат класифицирани към нито един от наличните стереотипи. Вследствие на това осъзнахме, че е нужно създаването на допълнителен стереотип „по подразбиране“, който да обхваща потребителите, не спадащи към нито един от наличните стереотипи. Този стереотип „по подразбиране“ послужи за оценяване на ефективността на създадения набор от стереотипи да групира потребителите. Методът, който използвахме се основава на оценка на това, какъв процент от потребителите попадат в стереотипа „по подразбиране“. Ако този процент е твърде висок, това ще е доказателство, че наборът не е достатъчен да обхване потребителите и трябва да се създадат допълнителни стереотипи. Ние приехме за начална допустима стойност 20% от потребителите да бъдат класифицирани към стереотипа „по подразбиране“, без да се налага добавяне на нови стереотипи или промяна в съществуващите.

Друг момент от изследването беше да установим дали някой от стереотипите не е „излишен“, като това е свързано с оценяване на минималната утилизация на стереотип. Това налага дефинирането на минимален брой потребители, които да бъдат класифицирани към стереотип. Ако този брой е под определен минимум, то тогава този стереотип не е нужен в системата. За оценка на минималния процент от потребители, класифицирани към даден стереотип може да се приложи подход, който се основава на усреднено разпределение на потребителите към съществуващите стереотипи. Този подход предполага приблизително 2% от потребителите да бъдат класифицирани към всеки от стереотипите в системата (50 на брой стереотипа и един „по подразбиране“). Този подход не отразява разликите в демографското разпределение на потребителите, свързани с работен език, роли и др. Например потребителите, говорещи английски език са повече от тези говорещи български, някои роли са по-разпространени от други, има разлики и в големината на институциите. Тези разлики рефлектират пряко върху процента от потребители, класифицирани към даден стереотип. От друга страна е естествено да се очакват вариации при потребителите, използващи системата в даден момент. Вземайки в предвид тези съображения, ние се спряхме на работна стойност от 0,5%, като приемлива утилизация за стереотип.

След така, направената постановка на очакванията и изискванията към системата, пристъпихме към оценяване на резултатите. Вследствие на проведените експерименти се оказа, че на този етап създаденият набор от стереотипи удовлетворява поставените цели и изисквания.

2. Втори етап на изследването: Оценяване на ефективността на метода за класифициране на потребител към стереотип

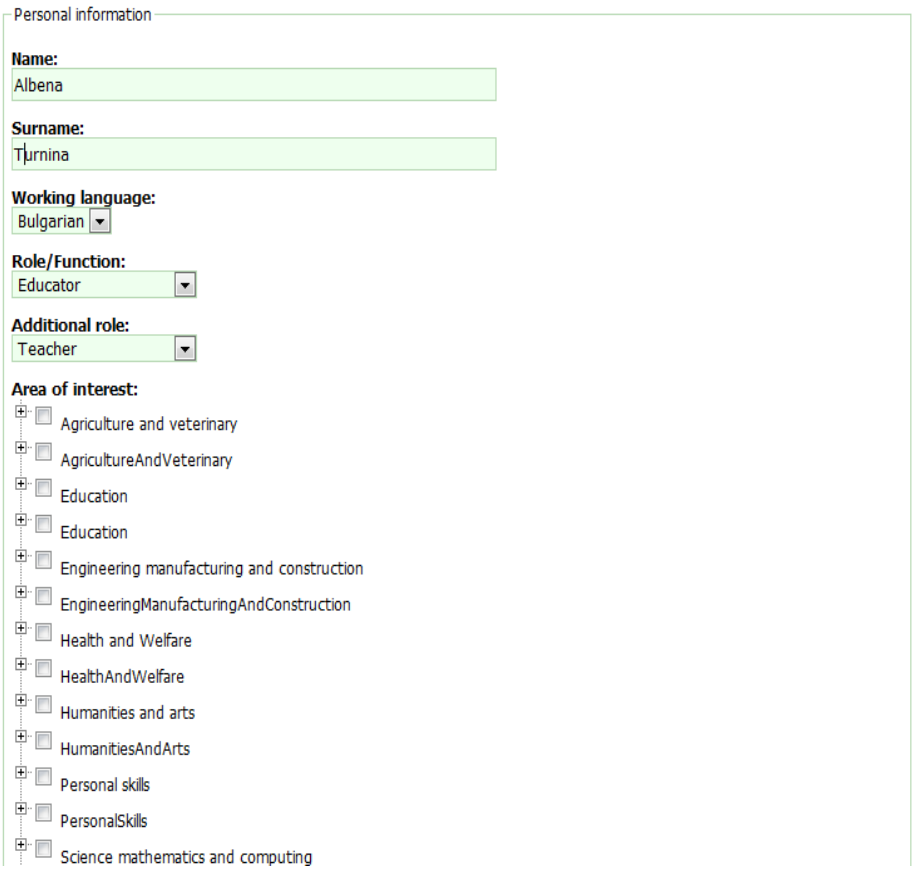
Този етап е свързан с оценяване дали потребителя бива класифициран към най-подходящия за него стереотип. Валидиране на ефективността на метода се осъществява, като се съпоставят резултатите, получени от автоматичното класифициране на потребител към стереотип/и с лично избрания от самия потребител стереотип/и.

Предложеният метод за класифициране се базира на създаване на вектор на потребител от въведените от него данни и сравняване на този вектор с предварително дефинираните вектори на стереотипите в системата. Резултат от това сравнение е получаване на числена стойност между нула и сто процента. При този процес е възможно получаването на стойност за сходство, което обаче е твърде малко и поради това от несъществено значение. Това налага необходимостта от дефиниране на стойност на сходствота, която да се приеме за значима (съществена) и да води до класифицирането на потребител към вектор. Изчисляването на сходството между потребител и всички стереотипи в системата води до извеждане на набор от стереотипи, с които потребителят е сходен в някаква степен. Например, потребителят uk може да бъде 100% сходен със стереотип Stereotype1, 80% сходен със стереотип Stereotype5, 60% сходен със стереотип Stereotype15, 20% сходен със стереотипи Stereotype20, Stereotype26 и Stereotype30 и т.н. Имайки предвид, че векторите се състоят от 5 елемента ние като работна стойността 0,8 или 80% и проведохме тестове с нея.

Цел на експеримента: Оценяване на класифицирането на потребител към стереотип.

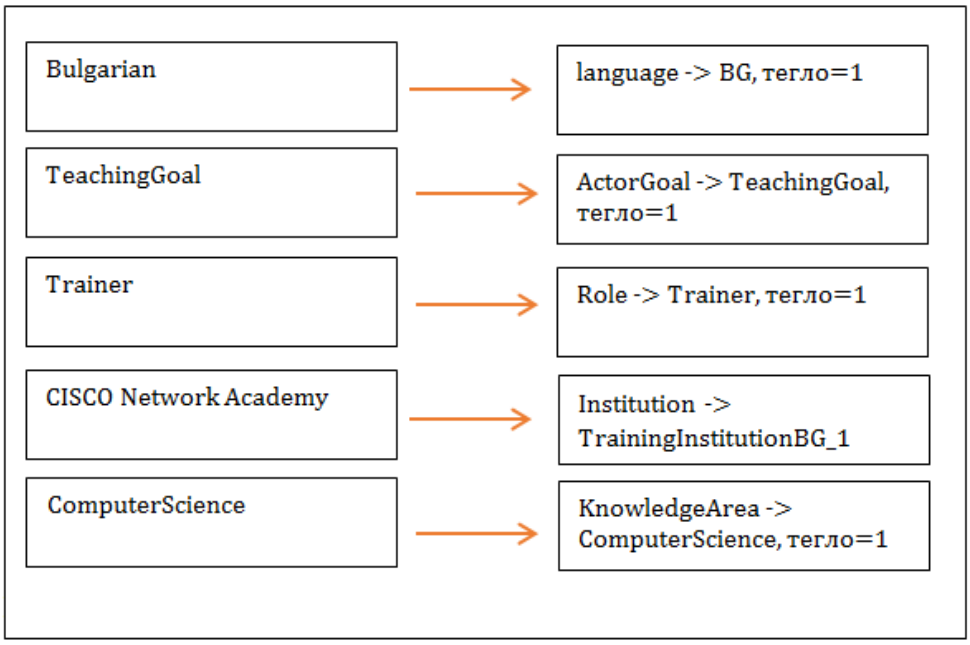
Постановката на експеримента: Извършва се автоматично класифициране на потребител към стереотип. Този резултат се сравнява с лично избран от същия

потребител стереотип. Методът за класифициране се основава на следното: потребителят предоставя данни за себе си, свързани с неговия език, целта му в системата, институцията му, изпълняваната от него роля и областта му на интерес. Тези данни се представят в онтологията TEO с онтологичните класове *Language, ActorGoal, Role, (EducationalInstitution* ∪ *InformationalCulturalAgency)* и *KnowledgeAreaElement*. От тези данни се конструира един или повече вектори на потребителя. В първоначалната версия на метода, беше планирано всеки потребител да се представя точно от един вектор. Впоследствие, това беше променено, тъй като се срещаха потребители, които задават повече от една стойност за всяка от възможностите (език, роля…). Например, потребители, който изпълняват едновременно две роли в една и съща или различни институции. Решението за тези случай е да се конструират повече от един вектори за такива потребители. Фиг.19 показва формуляр за регистрация на потребител, където отделните полета служат за въвеждане/избиране на стойности.



Фиг.19 Формуляр за регистрация на потребител

Примерен експеримент: Потребителят X е инструктор в Сиско академията на Софийски университет. При регистриране той задава следните данни за себе си: български език, обучителна цел, инструктор, Сиско академия, компютърни науки. От тези данни се създава векторът, чиито елементи съответстват на избрани онтологични класове и техните инстанции: [BG(1), TeachingGoal(1), Trainer(1), TrainingInstitutionBG_1(1), ComputerScience(1)]. Фиг. 20 показва процеса на конструиране на вектор на потребителя X от зададени от него данни.



Фиг.20 Конструиране на вектор на потребителя X от зададени от него данни

Конструираният вектор се сравнява с векторите на всички стереотипи в системата, използвайки метода на косинусовото сходство. Методът е демонстриран по-долу, като е изчислено сходството между вектора на потребителя X и два от стереотипите.

Таблица 18 показва използване на модела вектор-пространство за изчисление на сходство между вектор на потребител и два стереотипа - Stereotype15 и Stereotype19 с цел изчисляване на тяхното сходство. В таблицата са показани теглата на елементите на векторите. Всеки уникален елемент от векторите е координата в Евклидовото пространство и се използва като измерение в модела вектор-пространство.

Вектор на потребителя:

$\overrightarrow{u_1}$ = [BG(1), TeachingGoal(1), Trainer(1), TrainingInstitutionBG_1(1), ComputerScience(1)]

Вектор на Stereotype15:

$\overrightarrow{s_{15}}$ = [EN(1), TeachingGoal(1), StudentTeacher(1), UndergraduateIE_2(1), TrainingInMusic(1)]

Вектор на Stereotype19:

$\overrightarrow{s_{19}}$ = [BG(1), TeachingGoal(1), Trainer(1), TrainingInstitutionBG_1(1), ComputerScience(1)]

Таблица 18 Представяне на вектор на потребител и стереотипи в модела вектор-пространство

Елементи / Вектор	BG	TeachingGoal	Trainer	TrainingInstitutionBG_1	Computer Science
$\overrightarrow{u_1}$ Потребител	1	1	1	1	1
$\overrightarrow{s_{15}}$ (Stereotype15)	0	1	0	0	0
$\overrightarrow{s_{19}}$ (Stereotype19)	1	1	1	1	1

За изчисляване на сходството между вектор на потребител и стереотип, използваме формулата за косинусово сходство.

- Изчисляване на сходство между потребителя и стереотипа Stereotype15

$$\text{sim}(\overrightarrow{s_{15}}, \overrightarrow{u_1}) = \cos\boldsymbol{\vartheta} = \frac{\overrightarrow{s_{15}} \cdot \overrightarrow{u_1}}{|\overrightarrow{s_{15}}||\overrightarrow{u_1}|} = \frac{1.0+1.1+1.0+1.0+1.0}{\sqrt{1^2+1^2+1^2+1^2+1^2}\sqrt{0+1^2+0^2+0^2+0^2}} \sim 0{,}31$$

- Изчисляване на сходство между потребителя и стереотипа Stereotype19

$$\text{sim}(\overrightarrow{s_{19}}, \overrightarrow{u_1}) = \cos\boldsymbol{\vartheta} = \frac{\overrightarrow{s_{19}} \cdot \overrightarrow{u_1}}{|\overrightarrow{s_{19}}||\overrightarrow{u_1}|} = \frac{1.1+1.1+1.1+1.1+1.1}{\sqrt{1^2+1^2+1^2+1^2+1^2}\sqrt{1+1^2+1^2+1^2+1^2}} = 1$$

Изчисленията показват, че е налице пълно сходство от 100% между потребителя и стереотипа Stereotype19 и едва 31% между потребителя и стереотипа Stereotype15. Вследствие на изчисляване на сходството и при дефинирана гранична

стойност от 80% за приемливо сходство, потребителят X е класифициран към Stereotype19.

В последния етап на експеримента на потребителят X е представен списък със всички стереотипи, от който той да избере този който приема за най-сходен със себе си. Резултатите между изчисления и избрания стереотип съвпадат.

Очаквани резултати: Очакваните резултати са да има съвпадение между автоматичното класифициране на потребителя и избрания от него стереотип в мнозинството от случаите.

Получени резултати: Получените резултати показват съвпадение между изчисления по предложения метод стереотип и стереотипа, избран лично от потребителя.

Изводи: Проведените експерименти показват недвусмислено, че предложеният метод за изчисляване на сходство между потребител и стереотип е изключително надежден и ефективен. Благодарение на предвиденият стереотип „по подразбиране“ е възможно класифицирането на всички потребители на системата.

3. Трети етап в изследването: Оценяване на адаптивността при търсене; експерименти с Модела

Третият етап в изследването е свързан с набор от експерименти, които от една страна целят да валидират операциите, реализирани от Модела и от друга - да оценят предложения метод за адаптивно търсене. Тези експерименти се групират в две групи: експерименти с параметрите на Модела и експерименти с адаптиране на заявката.

Валидирането на първата серия от експерименти се реализира, като се съпоставят прогнозируемите, очаквани резултати от действието на Модела с реално получените. Валидирането на втората серия от експерименти се осъществява, като на потребител, преминал през процес на класифициране към стереотип се представят резултатите от адаптивно търсене и той оцени тази адаптация като положителна.

- Първа серия от експерименти: експерименти с Модела

- Експерименти с априорните вероятности и задаването на доказателства на възлите от най-горния слой

Цел на експеримента: Експериментиране със стойностите на априорните вероятности на възлите от най-горния слой с цел избор на подходящи стойности. изследваме как промените в априорните вероятности ще се отразят на разпределението на вероятностите при останалите възли от Модела. Изследване поведението на Модела при задаване на доказателства на възлите от най-горния слой.

Постановка на експеримента: Задаваме еднакви стойности на априорните вероятности на всички възли от най-горния слой на Модела в даден момент. Извършва се извод и се отчитат стойностите на вероятностите при възлите от останалите два слоя в Модела. Задаваме доказателства *истина* или *лъжа* на всички възли от най-горния слой. Отчитаме резултатите при останалите два слоя. В следваща серия от експерименти, променяме априорните вероятности, но така че отново да са еднакви за всички възли. Отново се извършва извод и се отчитат стойностите на вероятностите на възлите от долните два слоя. Задаваме същите доказателства *истина* или *лъжа* на всички възли от най-горния слой, както в предходната постановка. Отчитаме резултатите при останалите два слоя. Сравняваме получените резултати от двата етапа на експеримента.

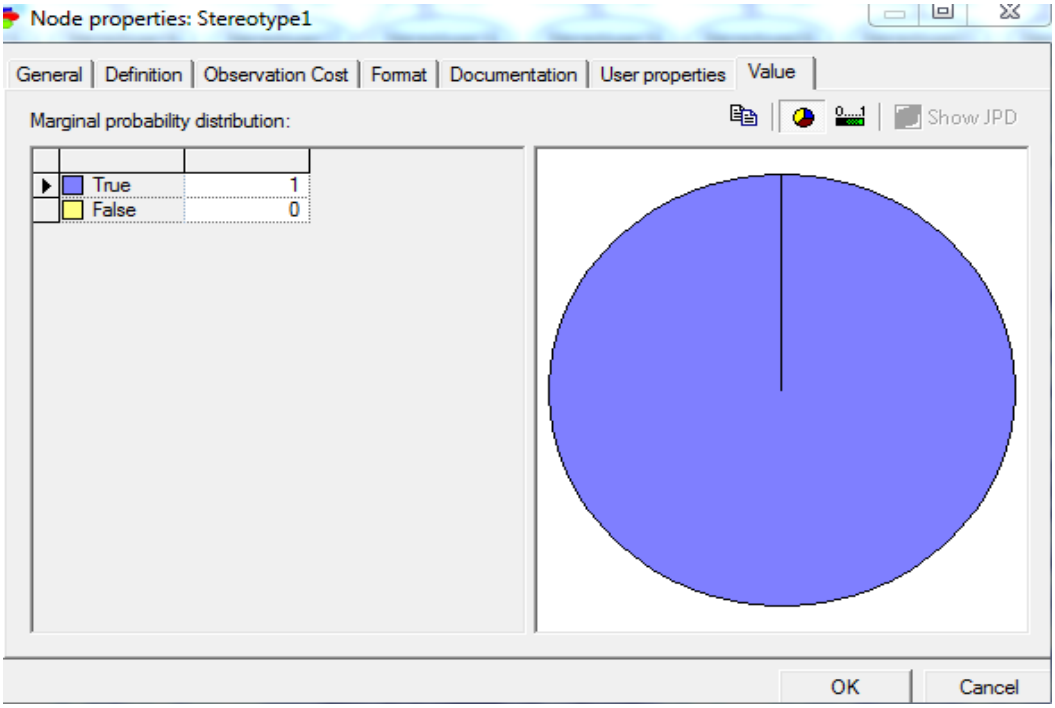
Примерен експеримент: задаваме стойности на априорните вероятности 0,2 за *истина* и 0,8 за *лъжа* или p(x = true) = 0,2 и p(x = false) = 0,8. Извършва се извод с Модела, в резултат на който се обновяват вероятностите на възлите от долните два слоя.

В Таблица 19 са показани тези вероятности за избрани за целите на експеримента възли от Модела.

Таблица 19 Вероятности на избрани възли от Модела, при задаване на априорни вероятности със стойности 0,2 за *истина* и 0,8 за *лъжа* на възлите от най-горния слой

Възел	p(x = true)	p(x = false)
ES	0.599	0.401
EN	0.678	0,322
LessonPlan	0,697	0,303
PedagogicalDesignPattern	0,312	0,688

В следващия етап на експеримента задаваме доказателства *истина* на възел Stereotype1 и *лъжа* на всички останали възли от най-горния слой. Фиг. 21 показва променените стойности на вероятностите след задаване на доказателство *истина* на Stereotype1.



Фиг.21 Вероятности на състоянията на възел Stereotype1 след зададено доказателство *истина*

В Таблица 20 са показани стойностите на вероятностите на същите възли, показани в Таблица 19, но при зададено доказателство *истина* на възел Stereotype1 и *лъжа* на всички останали възли от най-горния слой.

Таблица 20 Вероятности на избрани възли от Модела, при задаване на доказателство *истина* на възел Stereotype1 и *лъжа* на всички останали възли от най-горния слой при априорни вероятности със стойности 0,2 за *истина* и 0,8 за *лъжа*

Възел	p(x = true)	p(x = false)
ES	0,417	0,583
EN	0	1
LessonPlan	0,25	0,75
PedagogicalDesignPattern	0,25	0,75

Повтаряме експеримента, с променени стойности на априорните вероятности - p(x = true) = 0,5 и p(x = false) = 0,5 и отчитаме резултатите за същите възли. Резултатите са показани в Таблица 21. На следващия етап задаваме доказателство *истина* на възел Stereotype1 и *лъжа* на всички останали възли от най-горния слой. Резултатите са показани в Таблица 22. Сравняваме получени резултати от различните етапи на експеримента.

Таблица 21 Вероятности на избрани възли от Модела, при задаване на априорни вероятности със стойности 0,5 за *истина* и 0,5 за *лъжа* на възлите от най-горния слой

Възел	p(x = true)	p(x = false)
ES	0,806	0,194
EN	0,857	0,143
LessonPlan	0,851	0,149
PedagogicalDesignPattern	0,528	0,472

Таблица 22 Вероятности на избрани възли от Модела, при задаване на доказателство *истина* на възел Stereotype1 и *лъжа* на всички останали възли от най-горния слой при априорни вероятности със стойности 0,5 за *истина* и 0,5 за *лъжа*

Възел	p(x = true)	p(x = false)
ES	0,417	0,583
EN	0	1
LessonPlan	0,25	0,75
PedagogicalDesignPattern	0,25	0,75

Очаквани резултати: Очакваните резултати са при задаване на доказателства, те да надграждат стойностите на априорните вероятности, като последните губят своето значение.

Получени резултати: Получени са резултатите за избрани възли от Модела, при задаване на различни. Стойностите на вероятностите на избраните възли съвпадат при задаване на едни и същи доказателства на възлите от най-горния слой, без значение, че дефинираните априорни вероятности се различават.

Изводи: Задаването на стойности на априорните вероятности рефлектира върху разпределението на вероятностите на свързаните възли, като колкото по-високи

са тези стойности, толкова по-високи вероятностите на свързаните възли. Задаването на доказателства надгражда априорните вероятности.

- Експерименти с разпространяването на доказателства към възлите от долните два слоя

Зададените стойности на *стереотипите – истина* или *лъжа* се разпространяват до възлите от останалите два слоя, чиито вероятности се обновяват в съответствие със стойностите на предшестващите ги възли. В резултат на този процес можем да идентифицираме възлите от най-долния слой (*атрибути*), които да бъдат изведени от модела и подадени на Модул за прилагане на адаптивност при търсене и използвани за адаптиране на заявката. Това са тези атрибути, които имат стойност, различна от *лъжа*.

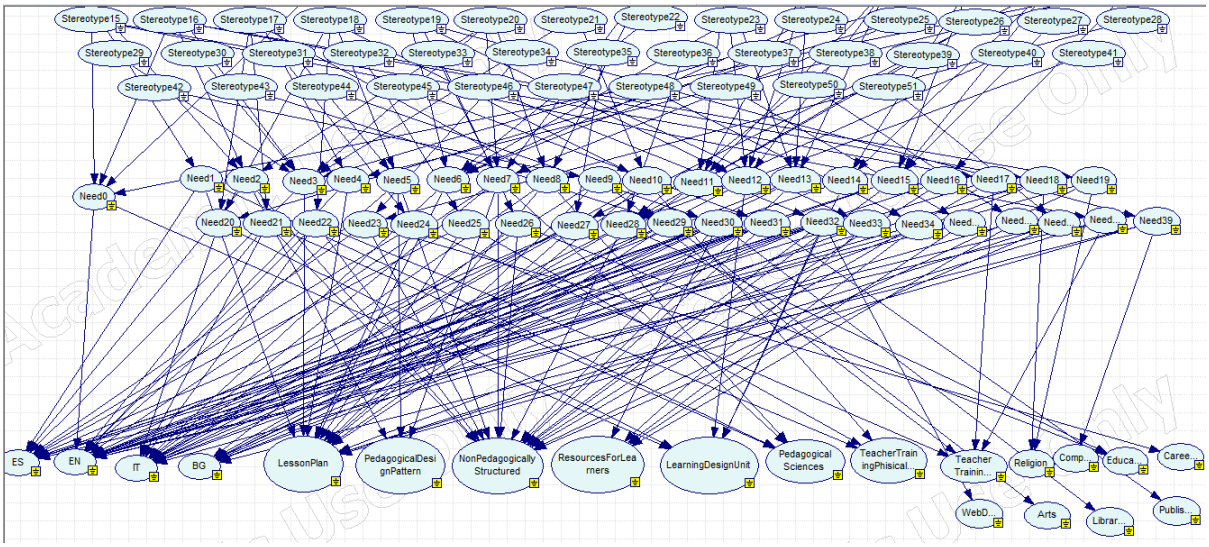
Експеримент 1

Цел на експеримента: Идентифициране на поведението на Модела при разпространението на доказателствата от най-горния слой към останалите два слоя от Модела. Валидиране на правилното разпространение на доказателствата, при зададената постановка на задачата.

Постановка на експеримента: Задаваме доказателство *лъжа* на всички възли от най-горния слой и изследваме разпространението на тези доказателства към останалите възли.

Очаквани резултати: Очакваните резултати са всички останали възли от Модела, също да получат стойност лъжа. Ако това не е така, то това ще означава, че Моделът не е изграден правилно и не отговаря на очакванията.

Наблюдавани резултати: Наблюдаваните резултати, потвърждават предварителните очаквания, а именно че при задаване на доказателство *лъжа* на всички възли от най-горния слой, тези доказателства се разпространяват към свързаните възли и като резултат всички останали възли от Модела също получават стойност *лъжа*. Резултата от експеримента може да се проследи на Фиг.22, където възлите маркирани с жълто квадратче, показват резултатна стойност *лъжа*.



Фиг.22 Резултат от задаване на доказателство *лъжа* на всички възли от най-горния слой

Експеримент 2

Цел на експеримента: Идентифициране на поведението на Модела, при разпространението на доказателствата. Валидиране на правилното разпространение на доказателствата, при зададената постановка на задачата.

Постановка на експеримента: Задаваме доказателство *истина* на избран възел от най-горния слой и *лъжа* на всички останали възли от този слой и изследваме разпространението на тези доказателства.

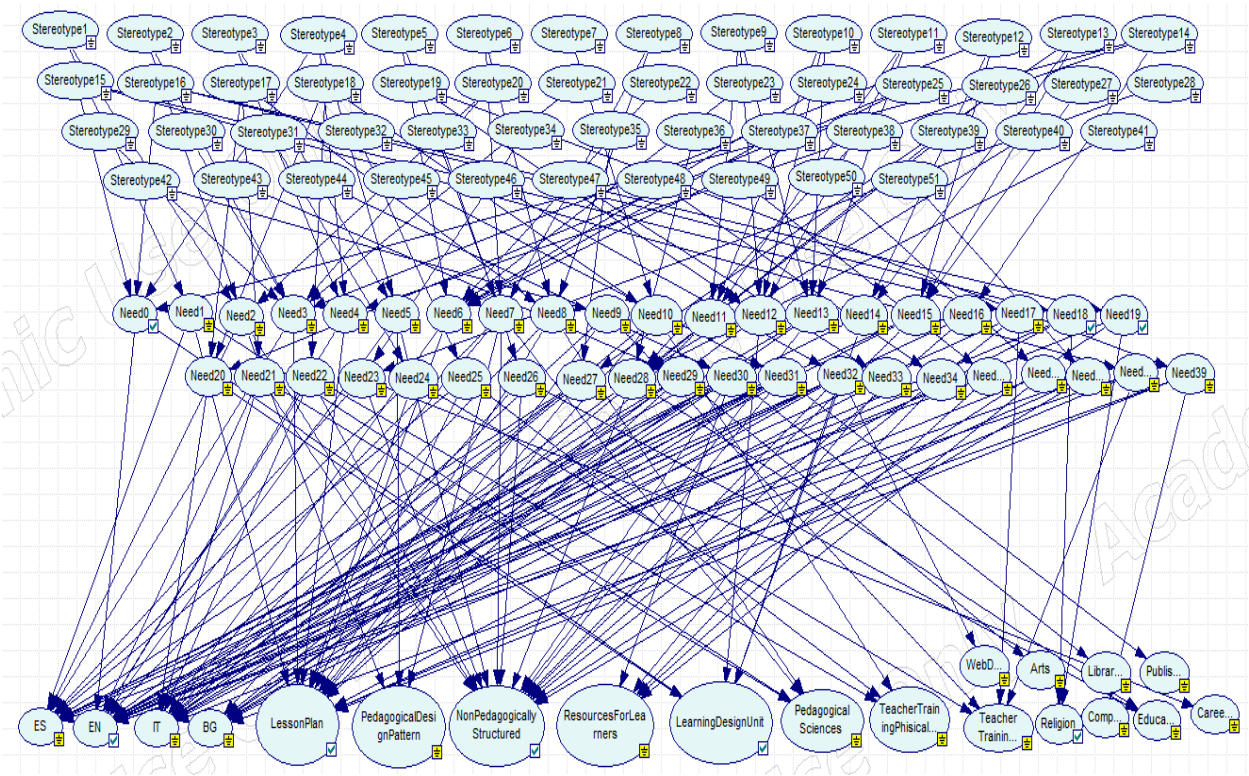
Примерен експеримент: Възелът на Stereotype16 получава доказателство *истина*, а всички останали възли на стереотипи – доказателство *лъжа*.

Очаквани резултати: Очакваните резултати са потока от доказателства да се разпространи към възлите от останалите два слоя от Модела, при което свързаните възли към възела със доказателство *истина* да получат вероятности за *истина* и *лъжа*, а всички останали възли – само стойност *лъжа*. Ако това не е така, то това ще означава, че Моделът не е изграден правилно и не отговаря на очакванията.

Наблюдавани резултати: Вследствие на разпространение на доказателствата към възлите от другите два слоя, наблюдаваме следното: възлите от средния слой - Need0, Need18 и Need19 са със стойност, различна от *лъжа*, а от най-

долния слой, това са възлите: EN, LessonPlan, NonPedagogicallyStructured, LearningDesignUnit и Religion.

Валидиране на експеримента: Установяваме, че възлите Need0, Need18 и Need19 са единствените възли от средния слой, които са директно свързани с възела Stereotype16, а възлите: EN, LessonPlan, NonPedagogicallyStructured, LearningDesignUnit и Religion – с някой от възлите Need0, Need18 и Need19. Тези резултати са доказателство, че поведението на Модела е в съответствие с очакванията. Фиг.23 показва Моделът при зададени доказателства *истина* на Stereotype16 и доказателства *лъжа* за всички останали стереотипи.



Фиг.23 Моделът при зададени доказателства *истина* на Stereotype16 и доказателства *лъжа* за всички останали стереотипи

Изводи: Задаването на доказателства е ефективен метод за идентифициране на възлите от най-долния слой (*атрибути*), които притежават вероятност, различна от *лъжа*.

- Втора серия от експерименти: експерименти с адаптиране на заявката

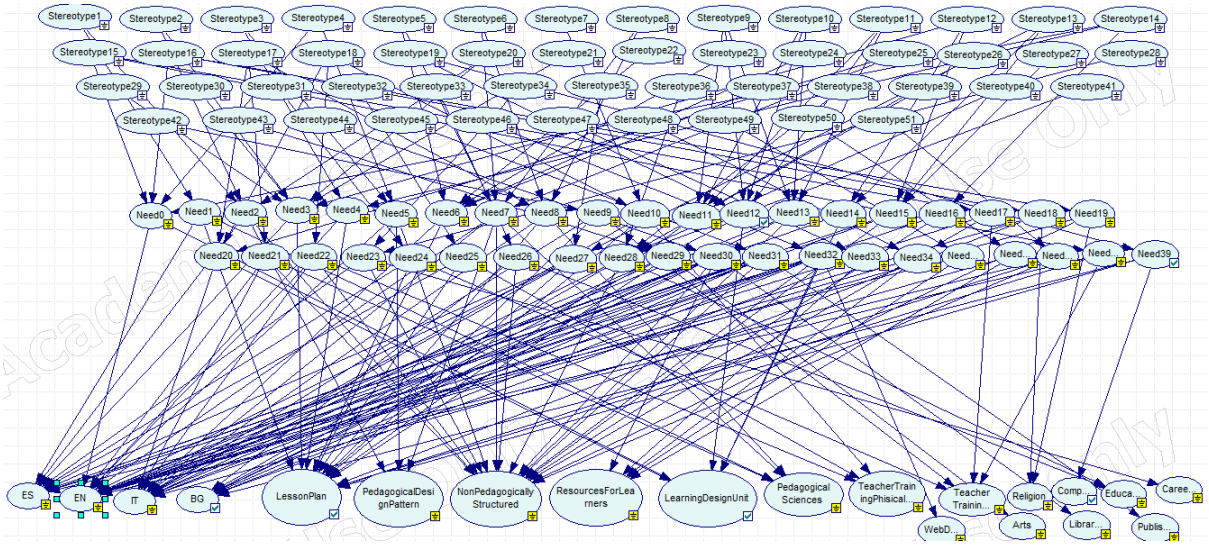
Адаптивност на търсенето е процес, посредством който отделните потребители получават резултати, съобразени с личните им предпочитания. Следствие от този процес е извеждането на различни резултати от една и съща заявка, тогава когато тези потребители се различават помежду си.

Цел на експеримента: Идентифициране на успешно реализирана адаптивност на търсене.

Постановка на експеримента: Потребител, класифициран към даден стереотип и вследствие на това, системата подава атрибути за адаптиране на заявката. Замяна на оригиналната заявка на потребител с адаптирана заявка. Съпоставяне на резултатите при два различни потребители, които първоначално са извършили търсене с една и съща ключова дума.

Примерен експеримент: Потребителят X е класифициран към Stereotype19 „Български Инструктор от CISCO мрежова академия“, а потребителят Y - към Stereotype7 „Италиански Обучител на обучители в област Физическо възпитание и спорт“. Двамата потребители извършват търсене с ключовата дума „lesson”.

Вследствие на извод с Модела, което в случая с потребителя X е свързано със задаване на доказателство истина на Stereotype19 и лъжа на всички останали стереотипи, се извеждат атрибутите: BG, LessonPlan, LearningDesignUnit, ComputerScience. Фиг.24 показва Модела при зададените условия.



Фиг. 24 Моделът след извършване на извод за потребителя X, класифициран към Stereotype19 „Български Инструктор от CISCO мрежова академия“

Подобен процес протича за потребителя Y, който е класифициран към стереотип Stereotype7. След извършване на извод, Моделът извежда атрибутите за потребител Y, които са: IT, LessonPlan и TeacherTraining-PhisicalTraining.

Изведените атрибути от Модела се използват за адаптиране на заявката, при което простата заявка за търсене по ключова дума се преобразува в разширена. Това се реализира, като се зарежда формата за разширено търсене в портала на Share.TEC и нейните полета се попълват със съответните за потребителя атрибути, според методът:

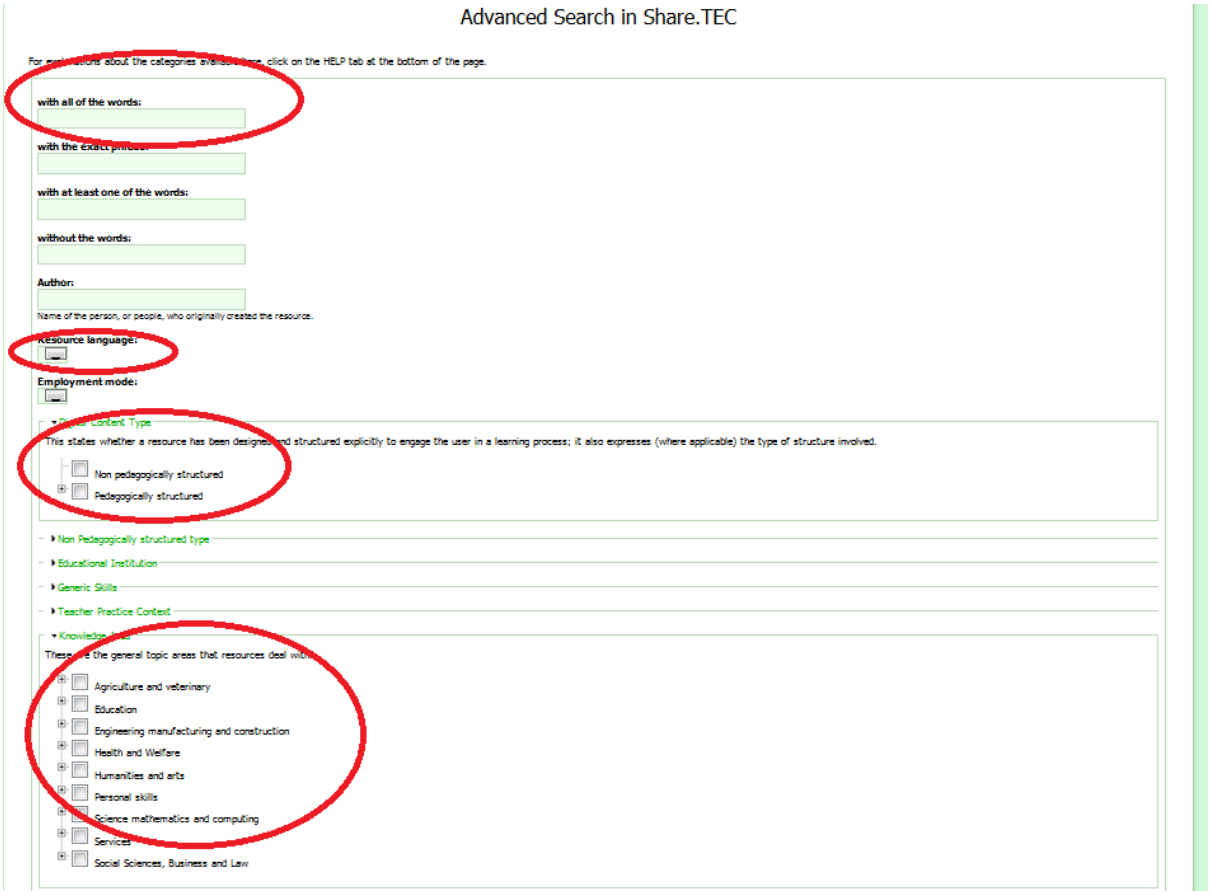
1. Въведените от потребителя ключови думи попълват полето - With all of the words.

2. Изведените атрибути от Бейсов модел за адаптиране и персонализиране попълват следните полета:

- Атрибутите от тип „Език“ попълват полето *Resource language*
- Атрибутите от тип „Тип на дигиталното съдържание“ попълват полето *DigitalContentType*
- Атрибутите от тип „Област на знание“ попълват полето *KnowledgeAreaElement*

Адаптираната заявка за търсене се изпраща в хранилището с ресурси. Разширената по този начин заявка ще бъде подобна на формата за Разширено търсене в

Share.TEC, в която полетата: *Resource language, DigitalContent Type и KnowledgeAreaElement* са попълнени автоматично със стойности, получени от Бейсов модел за адаптиране и персонализиране. Фиг.25 показва как се модифицира оригиналната заявка в разширена.



Фиг.25 Адаптиране на заявката с формата за разширено търсене в Share.TEC

Адаптираната заявка за потребител X ще има формата:

With all of the words: lesson

Resource language: BG

DigitalContent Type: LessonPlan, LearningDesignUnit

KnowledgeAreaElement: ComputerScience

Адаптираната заявка за потребител Y ще има формата:

With all of the words: lesson

Resource language: IT

DigitalContent Type: LessonPlan

KnowledgeAreaElement: TeacherTraining-PhisicalTraining

Адаптираните заявки за двата потребителя се изпращат към хранилището с информационни ресурси. Получените резултати се представят на потребителите.

Резултати от експеримента: Представяне на адаптиране резултати на поребителите, според техния стереотипен профил.

Валидиране на експеримента: Валидирането на успеха на реализираната адаптация може да се осъществи по два начина: или лично от потребителите X и Y или автоматично от системата. За автоматично валидиране е предложен метод, който се основава на свойството на БМ да извършват извод както в диагностична, така и в прогностична посока. Първата стъпка е свързана със събирането и обобщаването на съдържанието, разгледано от потребителя в резултат на търсенето му в системата. Информационните ресурси в дигиталната библиотека Share.TEC са анотирани с метаданни, представени в модела за метаданни CMM, който е базиран на стандарта LOM и онтологията TEO. В тези метаданни влизат аргументи на полетата: *Language, DigitalContent Type* и *KnowledgeAreaElement.* Тези аргументи се използват като възли от най-долния слой (*атрибути*) на Бейсовия модел за адаптиране и персонализиране. Като втора стъпка трябва да се изведат метаданните на разгледаните от потребителя ресурси. От тези метаданни се извеждат тези, които са от групите *Language, DigitalContent Type* и *KnowledgeAreaElement.* В третата стъпка, възлите от Модела, които съответстват на изведените метаданни, получават доказателство *истина*, а всички останали възли от най-долния слой на Модела – доказателство *лъжа***.** Следващата стъпка е свързана с активиране на алгоритъма за извод над Модела, в резултат на който се изчисляват постериорните вероятности на възлите от най-горния слой (*стереотипи*). Идентифицират се стереотипите за даден потребител – това са тези, които притежават най-голяма постериорна вероятност. Чрез съпоставяне на изведените по този начин стереотипи с изчислените на базата на векторно сходство стереотипи ще може числово да се измери, дали и до каква степен реализираната адаптивност е била успешна.

Изводи и заключения:

Проведените експерименти служат са доказателство за възможностите на предложеното решение да реализира успешно и ефективно адаптивно търсене в дигитална библиотека.

Изследванията и експериментите показват недвусмислено, че разработеното решение удовлетворява поставените цели, като предоставя:

- точни и подходящи стереотипи, базиране на наличните данни за потребителите на системата
- ясно дефинирани нужди на всяка от стереотипните групи, които обхващат набора от активности в системата
- атрибути за разширяване на заявката, съобразени с наличните метаданни в системата, които позволяват извеждане на нужното съдържание
- ефективен Модул за изпълнение на адаптивна и персонализираща логика, който удовлетворява нуждата от създаване на адаптивен и персонализиран поток от данни, в съответствие със стереотипа на потребителя
- гъвкава и преносима система, която позволява интегриране в различни среди и отговаряща на изискванията за многократна използваемост

ЗАКЛЮЧЕНИЕ

Основните приноси на дисертационния труд са както следва:

Научни приноси

Разработена е архитектура на Функционален модул за адаптивно търсене в дигитална библиотека, като са представени главните компоненти на архитектурата. Представени са отделните модули и тяхната функционалност.

Създадена е Базова схема на адаптивна и персонализираща логика за реализиране на адаптивно търсене в дигитална библиотека, която проследява основните дейности, осъществявани от потребител и съответните отговори на системата.

Дефинирана е адаптивна логика, която се базира на извод в Бейсова мрежа. За изпълнението на тази логика е необходимо класифициране на конкретен потребител към набор от дефинирани стереотипи. Класифицирането се реализира въз основа на набор от потребителски характеристики, предоставени директно от потребителя при неговата регистрация в системата. Стереотипите, към които потребителят е класифициран получават доказателства *истина* в предложения *Бейсов модел за адаптиране и персонализиране*, създаден за извод над знанието за потребител. *Бейсовият модел за адаптиране и персонализиране* използва свойството на каноничното взаимодействие NoisyOR да разпространява вероятностите към свързаните възли. За дефиниране на представянето на потребител, представянето на стереотипите и стойностите на атрибутите за адаптиране на заявката е използвана онтологията на предметната област TEO. Същата онтология е използвана за дефиниране на част от метаданните, с които са анотирани ресурсите в дигиталната библиотека Share.TEC. Вследствие на изпълнение на адаптивната логика на *Бейсов модел за адаптиране и персонализиране* се идентифицират тези атрибути, които да се използват за адаптиране на заявктата. Изведените от *Бейсов модел за адаптиране и персонализиране* атрибути съставляват част от реализираните метаданни в дигиталната библиотека. Това гарантира, че разширената с атрибутите заявка ще върне смислени реулатати.

Идентифицирани са факторите, които оказват влияние на съвместимостта на предложения *Функционален модул за адаптивно търсене в дигитална библиотека* със съществуващите модули в дигиталната библиотека.

Научно-приложни приноси

Разработени са стереотипи (51 на брой, включително стереотип по подразбиране) за групово представяне на потребители, споделящи общ набор от характеристики.

Разработен е списък с нуждите, които потребителите от дадена стереотипна група могат да притежават.

Идентифицирани са атрибути от съществено значение, които да гарантират подбор на подходящо информационно съдържание според нуждите на потребител.

Стереотипите, нуждите и атрибутите са включени в *Бейсов модел за адаптиране и персонализиране* за извод над знанието за потребител.

Разработен е Бейсов модел, който представлява трислойна мрежа, в която дъгите са утежнени и се използва каноничното взаимодействие NoisyOR за намаляване на CBP таблиците. Представена е структурата на модела, състояща се от възлите и тяхната наредба в мрежата. Разработени са параметрите на Бейсовия модел, състоящи се от априорни вероятности за възлите от най-горния слой и условни вероятности за възлите от долните слоеве.

Предложен е подход са класифициране на потребител към стереотип на базата на Модела вектор-пространство, който е точен и доказано ефективен метод. За реализиране на този подход е разработено представяне на стереотипите като вектори, съставени от елементи на онтологията ТЕО в дигиталната библиотека.

Разработени са три функционални модули от представената архитектура: модул за представяне и управление на знанието за потребител, модул за изпълнение на адаптивна и персонализираща логика и модул за прилагане на адаптивност при търсене. Те осигуряват доставяне на адаптирано и персонализирано съдържание при търсене в дигитална библиотека. Реализираната адаптивност и персонализация се базира на предложените адаптивни стратегии и методи, както и на използването на

онтология за представяне на знанието за потребител, за представяне на разработените стереотипи и за описание на използваните в библиотеката метаданни.

Предложен е подход за адаптивно търсене, базиран на метода разширяване на заявката, който използва реализираното в дигиталната библиотека разширено търсене. С това се предлага решение, което е оптимално, ефективно и съгласувано с функционалността на останалите модули в дигиталната библиотека.

Насоки за бъдещо развитие на системата

В бъдеще е възможно да се реализира разширяване на Бейсовия модел, посредством добавяне на нови възли, към всеки от слоевете, както и създаване на нови релации. Чрез добавянето на нови възли би било възможно моделирането на допълнителни вероятностни събития. Възможни са промени в числовите параметри на Модели, като задаване на други стойности на тежестите на дъгите между възлите, дефиниране на различни стойности на вероятностите и др.

Тези промени биха възникнали като резултат от наблюдаването на промени в потребителите и техното поведение в системата.

Друга насока за бъдещо развитие е свързана с модификация на теглата на елементите от векторите, така че те да отразяват релациите в онтологията ТЕО. Например, при липса на сходство между съответни елементи от векторите на потребител и стереотип, да се проверява за елементи от същия клас, съседен клас или суперклас, в който случай тези елементи да получават по-малка тежест. В настоящата реализация сме се спрели на подход, който е доказано ефективен, но не използва тези релациите в онтологията. Това е така, защото смятаме, че това би усложнило решението, а ползата от него не би била съществена.

В настоящата реализация, данните за конструиране на вектор на потребител се предоставят от самия потребител (директно). В бъдеще е възможно да се експериментира с данни, събрани от самата система (индиректно), като такива данни биха получавали по-малка тежест във вектора на потребителя, отколкото тези, които той сам е предоставил.

Бъдещото развитие на системата може да включва функционалност за препоръчване на съдържание на потребител, което е разгледано от други потребители, които са категоризирани към същия стереотип/и. Възможно е да се разшири

функционалността на Системата за препоръчване, като се препоръчва не само потенциално релевантно съдържание, но и потенциало „подходящи за контакт“ потребители.

В бъдещото развитие на системата е възможно да се експериментира с подход, при който изведените стойности за сходство между потребителя и стереотипите се използват при подреждане (класиране) на резултатите от търсенето. При този подход най-отгоре в списъка с резултати ще се поставят резултатите, получени от адаптиране на заявката с атрибути от първия „най-сходен“ стереотип, след това с атрибути от следващия по сходство стереотип и т.н.

ДЕКЛАРАЦИЯ ЗА ОРИГИНАЛНОСТ

Декларирам, че настоящата дисертация е изцяло авторски продукт и в нейното разработване не са ползвани чужди публикации и разработки в нарушение на авторските им права.

ДЕКЛАРАЦИЯ ЗА ОРИГИНАЛНОСТ

Декларирам, че представената във връзка с провеждането на процедура за придобиване на образователната и научна степен „доктор" в Софийски университет "Св. Климент Охридски" дисертация на тема: "Семантично-ориентирани архитектури и използване на онтология за организиране на адаптивно търсене" е мой труд и в нейното разработване не са ползвани чужди публикации и разработки в нарушение на авторските им права.

Цитиранията на всички източници на информация, текст, илюстрации, таблици, изображения и други са обозначени според стандартите.

Резултатите и приносите на проведеното дисертационно изследване са оригинални и не са заимствани от изследвания и публикации, в които нямам участие.

ДЕКЛАРАЦИЯ ЗА ОРИГИНАЛНОСТ

Във връзка с провеждането на процедура за придобиване на образователна и научна степен „доктор" в СУ „Св. Кл. Охридски" и защита на представения от мен дисертационен труд декларирам, че:

- Резултатите и приносите на проведеното дисертационно изследване, представени в дисертационния труд на тема „Семантично-ориентирани архитектури и използване на онтология за организиране на Адаптивно търсене ", са оригинални и че те не са заимствани от изследвания и публикации, в които нямам участие.
- В текста на този дисертационен труд не са използвани неправомерно текстове и други обекти на авторското право, без да бъде посочен източникът или без да съществува разрешение или законово право за това.
- Настоящата дисертация не е прилагана за придобиване на научна степен в друго висше училище, университет или научен институт.
- Представената от мен информация – списък с публикации, копия на документи и получени по време на експериментите резултати – отговаря на обективната истина.

БИБЛИОГРАФИЯ

[1] Ardissono, L., Console, L., Torre, I. (2001). An adaptive system for the personalized access to news. Journal AI Communications, Volume 14 Issue 3, pages 129 – 147. IOS Press Amsterdam, The Netherlands

[2] Ardissono, L. and Sestero, D. (1996). Using dynamic user models in the recognition of the plans of the user. User Modeling and User-Adapted Interaction, 5(2):157--190

[3] Arroyo, I. and Woolf, B.P. (2005). Inferring learning and attitudes from a Bayesian Network of log file data. In: Looi, C.-K., McCalla, G., Bredeweg, B., Breuker, J. (eds.) Proc. of 12th International Conference on Artificial Intelligence in Education, AIED'2005. IOS Press (2005) 33-40

[4] Baeza -Yates R., Ribeiro-Neto, B.(1999).Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA ©1999 ISBN:020139829X

[5] Balabanovic, M., & Shoham, Y. (1997). Fab:Content-based, collaborative recommendation. Communications of the ACM, 40(3), 66-72

[6] Barra, M., Negro, A. & Scarano, V. (1999). When the Teacher learns: a Model for Symmetric Adaptivity. Proceedings of Second Workshop on Adaptive Systems and User Modeling on the World Wide Web, Banff, Canada

[7] Boytchev, P., Grigorov, A., Earp, J., Stefanov, K., and Georgiev, A. (2010). Adaptability Approaches in Digital Libraries Available at: http://hdl.handle.net/10867/55 (достъпен на 06 юни, 2016)

[8] Boytchev, P., Bocconi, S., Stefanov, K. (2010). An Ontological Approach for Multicultural Digital Repositories, Available at: http://hdl.handle.net/10867/77 (достъпен на 06 юни, 2016)

[9] Brajnik, Cl., Guida, G., & Tasso, C. (1990). User modeling in expert man-machine interfaces: a case study in intelligent information retrieval. IEEE Transacfions on Systems, Man and Cybernefics, 2O(I), 166-185

[10] Brusilovsky P. (2001). User Modeling and User-Adapted Interaction, 11: 87-110

[11] Brusilovsky, P., Tasso, C. (2004). Preface to special issue on user modeling for Web information retrieval. User Model. User-Adapt. Interact. 14, 147–157

[12] Brusilovsky, P., Millán, E. (2007). User Models for Adaptive Hypermedia and Adaptive Educational Systems. The Adaptive Web. Lecture Notes in Computer Science Volume 4321, 2007, pp 3-53

[13] Callahan, E. (2005). Cultural similarities and differences in the design of university websites. Journal of Computer-Mediated Communication, 11(1)

[14] GeNIe and SMILE, Available at: http://www.bayesfusion.com/ (достъпен на 06 юни, 2016)

[15] Chang, S. K., Costabilet, M. F.. & Leviald, S. (1993). Modeling users in an adaptive visual interface. Journal of Visual Languages and Compuring, 4(2), 143-159

[16] Chen, L., Sycara, K. (1998). A Personal Agent for Browsing and Searching. In: Proceedings of the 2nd International Conference on Autonomous Agents, Minneapolis/St. Paul, May 9-13, 132-139

[17] Chin, D. N. (1989). KNOME: modeling what the user knows in UC. In A. Kobsa, & W. Wahster (Eds), Ilser Models in Dialog Systems (Ch. 3, pp. 74-107). Berlin: Springer-Verlag

[18] Chirita, P.A., Firan, C.S., Nejdl, W. (2007). Personalized query expansion for the Web. In: 30th Annual International ACMSIGIR Conference on Research and Development in Information Retrieval (SIGIR 2007), pp. 7–14. ACM, Amsterdam

[19] Conati, C., Maclaren, H. (2005). Data-Driven Refinement of a Probabilistic Model of User Affect. In: Proc. of 10th International Conference UM'05. Lecture Notes in Computer Science, Vol. 3538. Springer-Verlag 40-49

[20] Conati, C., Gertner, A., VanLehn, K. (2002). Using Bayesian Networks to Manage Uncertainty in Student Modeling. User Modeling and User-Adapted Interaction 12, 4 371-417

[21] Dagum, P., Galper, A. (1993). Additive belief-network models, in Proc. of the 9th Annual Conference on Uncertainty in Artifiticial Intelligence (UAI)

[22] Darwiche. A. (2010) What are bayesian networks and why are their applications growing across all fields? Communications of the ACM, 53:80–90, December

[23] De Bra, P., Calvi L. (1998). AHA: A Generic Adaptive Hypermedia System". Proc. of the 2nd Workshop on Adaptive Hypertext and Hypermedia at the 9th ACM Conference on Hypertext and Hypermedia (Hypertext 98), Pittsburgh, USA.

[24] Dıez, F. (1993). Parameter adjustement in Bayes networks. The generalized NoisyOR–gate. In Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence, pages 99–105, Washington D.C., Morgan Kaufmann, San Mateo, CA

[25] Dolog, P. (2004). Identifying Relevant Fragments of Learner Profile on the Semantic Web, In the Proceedings of the SW-EL'04 Workshop at ISWC 2004 Conference, Hiroshima, Japan

[26] Dolog, P., Nejdl, W. (2007). Semantic Web Technologies for the Adaptive Web. The Adaptive web. Lecture Notes in Computer Science, Volume 4321/2007, 697-719, DOI: 10.1007/978-3-540-72079-9_23. p.697-719

[27] Dumais, S., Cutrell, E., Cadiz, J., Jancke, G., Sarin, R., Robbins, D.C. (2003). Stuff i've seen: a system for personal information retrieval and re-use. In: SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, New York, NY, USA, ACM Press, 72–79

[28] IEEE P1484.2.22/D8, PAPI Learner - Learner Relations Information, Available at: http://metadata-standards.org/Document-library/Meeting-reports/SC32WG2/2002-05-Seoul/WG2-SEL-046_SC36N0182_papi_learner_relations_info.pdf (достъпен на 06 юни, 2016)

[29] IEEE Standard for Learning Object Metadata 1484.12.1-2002. IEEE LTSC, Available at: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf (достъпен на 06 юни, 2016)

[30] IMS LIP (2001). IMS Learner Information Package, Available at: http://www.imsproject.org/aboutims.html (достъпен на 06 юни, 2016)

[31] Fensel, D. (2004). Ontologies: A Selver Bullet for Knowledge Management and Electronic Commerce, Second edition

[32] Finin, W. (1989). GUMS: a general user modeling shell. In A. Kobsa & W. Wahster (Eds), user models in dialog systems (Ch. 15, pp. 41 l-430). Berlin: Springer-Verlag

[33] Friedman, N., Geiger, D., and Goldszmidt, M. (1997). Bayesian network classifiers. Machine Learning, 29(2-3):131–163

[34] Ganeshan, R., Johnson, W., Shaw, E., Wood, B.P. (2000). Tutoring Diagnostic Problem Solving. In: Proc. of 7th International Conference on Intelligent Tutoring Systems, ITS'2004. Vol.1839. Springer-Verlag 33-42

[35] García, F., Amandi, A., Schiaffinoa, S., Campoa, M. (2006). Evaluating Bayesian networks' precision for detecting students' learning styles. Computers & Education

[36] Gasparini, I., Weitzel, L., Pimenta, M.S. & Oliveira, J.P.M.d. (2011). Adaptive e-learning for all: integrating cultural-awareness as context in user modeling. In T. Bastiaens & M. Ebner (Eds.), Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2011, pp. 1321-1326

[37] Gauch, S., Speretta, M., Chandramouli, A., Micarelli, A. (2007). User profiles for personalized information access. In: Brusilovsky, P., Kobsa, A., Neidl, W. (eds.): The Adaptive Web: Methods and Strategies of Web Personalization. Lecture Notes in Computer Science, Vol. 4321. Springer-Verlag, Berlin Heidelberg New York

[38] Ghorab, M., Zhou, D., O'Connor, A., Wade, V. (2012). Personalised Information Retrieval: survey and classification. User Model User-Adap Inter DOI 10.1007/s11257-012-9124-1. © Springer Science+Business Media B.V. 2012

[39] Good, I. J. (1961). A causal calculus. The British Journal for the Philosophy of Science, 12 (45), 43–51

[40] Heckerman, D. Causal Independence for Knowledge Acquisition and Inference, Available at: https://arxiv.org/ftp/arxiv/papers/1303/1303.1468.pdf (достъпен на 06 юни, 2016)

[41] Henrion, M. (1987). Some practical issues in constructing belief networks. In Proceedings of the Third Workshop on Uncertainty in Artificial Intelligence, Seattle, WA, pages 132–139. Association for Uncertainty in Artificial Intelligence, Mountain View, CA. Also in Kanal, L., Levitt, T., and Lemmer, J., editors, Uncertainty in Artificial Intelligence 3, pages 161–174. North-Holland, New York, 1989

[42] Henrion, M. (1989). Some practical issues in constructing belief networks. In L. N. Kanal, T. S. Levitt, and J. F. Lemmer, editors, Uncertainty in Artificial Intelligence 3, pages 161–173. North-Holland, Amsterdam

[43] Herlocker, J., Konstan, J., & Riedl, J. (2000). Explaining collaborative filtering recommendations. In ACM Conference on Computer Supported Cooperative Work

[44] Henze, N., Nejdl, W. (2004). Student modeling for KBS Hyperbook system using Bayesian networks. In Adaptive Hypermedia and Adaptive Web-Based Systems: Third International Confeerence, AH 2004 Eindhoven, The Netherlands, August 2004 Proceedings

[45] Henze, N., Nejdl, W. (2003). Logically characterizing adaptive educational hypermedia systems. In Proc. of the AH

[46] Jameson, A. (1992). Generalizing the double-stereotype approach: A psychological perspective. In UM92: Third International Workshop on User Modelling., 69-83

[47] Jensen, F., Andersen, S. (1990). Approximations in Bayesian Belief Universes for Knowledge-based Systems. In Proceedings of the 6th Conference on Uncertainty in Artificial Intelligence, Cambridge, MA, 162-169

[48] Jiang, X., Tan, A. (2009). Learning and inferencing in user ontology for personalizedSemantic Web search. Information Sciences 179 (2009) 2794–2808. p. 2794 – 2808

[49] Kessler, M. (1963). Bibliographic coupling between scientific papers, American Documentation, 14, 10-25

[50] Kim, J. and Pearl, J. (1983). A computational model for causal and diagnostic reasoning in inference engines. In Proceedings Eighth International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, pages 190–193. International Joint Conference on Artificial Intelligence.

[51] Kraaijeveld, P., Druzdzel, M. (2005). GeNIeRate: An Interactive Generator of Diagnostic Bayesian Network Models. In 16th International Workshop on Principles of Diagnosis

[52] Kobsa, A., Koenemann, J., Pohl, W. (2001). Personalised hypermedia presentation techniques for improving online customer relationships, Available at: www.ics.uci.edu (достъпен на 06 юни, 2016)

[53] Koutrika, G., Ioannidis, Y. (2004). Rule-based query personalization in digital libraries. Int. J. Digit. Libr. 4, 60–63

[54] Kruschwitz, U., Albakour, M., Niu, J., Leveling, J., Nanas, N., Kim, Y., Song, D., Fasli, M., Roeck, A. (2011). Moving towards Adaptive Search in Digital Libraries. Advanced Language Technologies for Digital Libraries Lecture Notes in Computer Science Volume 6699, 2011, pp 41-60

[55] Lauritzen, S., Spiegelhalter, D. (1988). Local Computations with Probabilities on Graphical Structures and their Application to Expert Systems. Journal of the Royal Statistical Society, 50, No. 2

[56] Liu, F., Yu, C., Meng, W. (2004). Personalized Web search for improving retrieval effectiveness. IEEE Trans. Knowl. Data Eng. 16, 28–40

[57] Malone, T. Grant, K., Turbak, F., Brobst, S., Cohen, M. (1987). Intelligent information sharing systems. Communication of the ACM, 30(5), 390-402.

[58] Manning, C., Raghavan, P., Schutze, H. (2008). Introduction to Information Retrieval p. 100. doi:10.1017/CBO9780511809071.007. ISBN 9780511809071.

[59] Mayo, M., Mitrovic, A. (2001). Optimising ITS behaviour with Bayesian networks and decision theory. International Journal of Artificial Intelligence in Education 12, 124-153 CAPIT

[60] Micarelli, A., Sciarrone, F. (2004). Anatomy and empirical evaluation of an adaptive Web-based information filtering system. User Model. User-Adapt. Interact. 14, 159–200

[61] Micarelli, A., Gasparetti, F., Sciarrone, F., Gauch, S. (2007). Personalized Search on theWorld Wide Web. The adaptive web, pages 195-230. Springer-Verlag Berlin, Heidelberg ISBN: 978-3-540-72078-2

[62] Minsky, M. (1975). A framework for representing knowledge. In P. Winston (Ed.), The psvchology of computer vision. New York: McGraw-Hill

[63] Mislevy, R., Gitomer, D.H. (1996). The Role of Probability-Based Inference in an Intelligent Tutoring System. User Modeling and User-Adapted Interaction 5, 3-4 253-282

[64] Morik, K., & Rollinger, C. R. (1985). The real estate agent: modeling users by uncertain reasoning. AI Maga:ine, 6, 44-52.

[65] Moukas, A. (1997). Amalthaea: Information Discovery And Filtering Using A Multiagent Evolving Ecosystem. In: Applied Artificial Intelligence 11(5) 437-457

[66] Muldner, K., Conati, C. (2005). Using Similarity to Infer Meta-Cognitive Behaviours During Analogical Problem Solving. In: Proc. of 10th International Conference UM'05. Vol. 3538. Springer-Verlag 134-143

[67] Murray, R.C., VanLehn, K., Mostow, J. (2004). Looking ahead to select tutorial actions: A decision-theoretic approach. International Journal of Artificial Intelligence in Education 14, 3-4 235-279

[68] Oard, D.W., Diekema, A.R. (1998). Cross-language information retrieval. In: Williams M. (ed.) Annual Review of Information Science (ARIST), pp. 472–483. Information Today Inc., Medford

[69] Paneva D. (2006). Ontology-based Student Modeling, In the Proceedings of the Fourth CHIRON Open Workshop "Ubiquitous Learning Challenges: Design, Experiments and Context Aware Ubiquitous Learning", 20 - 21 September, 2006, Turin, Italy, pp. 17-21

[70] Paliouras, G., Karkaletsis, V., Papatheodorou, C., Spyropoulos, C. (1999). Exploiting Learning Techniques for the Acquisition of User Stereotypes and Communities, Proceedings of the International Conference on User Modelling

[71] Pazzani, M. (1999). A framework for collaborative, content-based and demographic filtering. In Artificial Intelligence Review, 13(5-6), 393–408.

[72] Pazzani, M., Billsus, D. (2007). Content-Based Recommendation Systems. In The Adaptive Web, 325-341.

[73] Pearl, J. (1986). Fusion, propagation, and structuring in belief networks. Artificial Intelligence, 29(3):241–288

[74] Pearl, J. (1988) Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgen Kaufmann Publishers, Inc., San Mateo, CA

[75] Provan, G. (1995). Abstraction in belief networks: The role of intermediate states in diagnostic reasoning. In Proceedings of the 11th Annual Conference on Uncertainty in Artificial Intelligence (UAI-95), pages 464–471, San Francisco, CA, Morgan Kaufmann Publishers

[76] Radde, S., Kaiser, A., & Burkhard, F. (2008). A Model-Based Customer Inference Engine. In: Zanker, M., Felfernig, A, & Burke, R. (eds.), ECAI'08 Workshop on Recommender Systems, University of Patras, 2-7

[77] Read, T., Bárcena, E., Barros, B., Verdejo, F. (2002). I-PETER: Modelling Personalised Diagnosis and Material Selection for an Online English Course. In: Proc. of 8th Ibero-American Conference IBERAMIA'2002. Lecture Notes in Computer Science, Vol. 2527. Springer-Verlag 734-744

[78] Rich, E. (1979). User modeling via stereotypes. Cognitive Science, 3, 329-354.

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

[80] Rich, E. (1989). Stereotypes and user modeling. In A. Kobsa & W. Wahster (Eds), User models in dialog systems (Ch.2. pp. 75-85). Berlin: Springer-Verlag.

[85] Share.TEC Project Deliverable D2.2 Common Metadata Model (CMM): version 1

[86] Share.TEC Deliverable 5.5 Development and pilot deployment of the integrated system

[87] Shwe, M., Middleton, B., Heckerman, D., Henrion, M., Horvitz, E., Lehmann, H., Cooper, G. (1991). Probabilistic diagnosis using a reformulation of the INTERNIST–1/QMR knowledge base: I. The probabilistic model and inference algorithms. Methods of Information in Medicine, 30(4):241–255

[88] Smeaton, A., Callan J. (2005). Personalisation and Recommender System in Digital Libraries, International Journal „Digital Libraries", vol. 5, pp. 288-308

[89] Speretta, M., Gauch, S. (2005). Personalized search based on user search histories. In: Web Intelligence (WI2005), France, IEEE Computer Society 622–628 http://dx.doi.org/10.1109/WI.2005.114

[90] Spertus, E., Sahami, M., Buyukkokten, O. (2005). Evaluating similarity measures: A large-scale study in the Orkut Social Network. Proceedings of KDD'05

[91] Stacey, K., Sonenberg, L., Nicholson, A., Boneh, T., Steinle, V. (2003). A Teaching Model Exploiting Cognitive Conflict Driven by a Bayesian Network. In: Brusilovsky, P., Corbett, A., Rosis, F.d. (eds.) Proc. of 9th International User Modeling Conference. Lecture Notes in Computer Science, Vol. 2702. Springer-Verlag 352-362

[92] Stadnyk, I., Kass, R. (1992). Modeling users' interests in information filters. Communication of the ACM, 35(12). 49-50

[93] Stefanov, K., Boytchev, P., Grigorov, A., Georgiev, A., Petrov, M., Gachev, G., and Peltekov, M. (2009), "Share.TEC System Architecture", In Proceedings of 1st International conference S3T'2009 Software, Services and Semantic Technologies, Ed. D. Dicheva, R. Nikolov, E. Stefanova, pp. 92-99, ISBN 978-954-9526-62-2

[94] Stefanov, K., Boytchev, P., Grigorov, A., Georgiev, A., Petrov, M., Gachev, G., Peltekov, M., (2009). Recommender systems and repository search: the Share.TEC proposal, Available at: http://hdl.handle.net/10867/36 (достъпен на 06 юни, 2016)

[95] Stefanov, K., Boytchev, P., Grigorov, A., Georgiev, A., Petrov, M., Gachev, G., Peltekov, M., (2009) Share.TEC Repository System, Available at: http://hdl.handle.net/10867/36 (достъпен на 06 юни, 2016)

[96] Stefanov, K., Nikolov, R., Boytchev, P., Stefanova, E., Georgiev, A., Koychev, I., Nikolova, N., Grigorov, A. (2011). Emerging Models and e-Infrastructures for Teacher Education, 2011 International Conference on Information Technology Based Higher Education and Training ITHET 2011, paper 33, IEEE Catalog Number: CFP11578-CDR, ISBN: 978-1-4577-1671-3

[97] Stefanov, K., Boytchev, P., Stefanova, E., Georgiev, A., Nikolova, N., Grigorov, A. (2011). Digital Libraries in Teacher Education, Available at: https://research.uni-sofia.bg/handle/10506/762 (достъпен на 06 юни, 2016)

[98] Teacher Education Ontology (TEO): version 1

[99] UNESCO (2010). Манифест на ИФЛА/ЮНЕСКО за дигиталните библиотеки, Available at: http://www.ifla.org/files/assets/digital-libraries/documents/ifla-unesco-digital-libraries-manifesto-bg.pdf (достъпен на 06 юни, 2016)

[100] Tsianos N., Germanakos P., Lekkas Z., Mourlas, C., Samaras, G. (2009). An Assessment of Human Factors in Adaptive Hypermedia Environments Intelligent User Interfaces, 2009,pp 1-35

[101] Vassileva, J. (1994). A practical and adaptable user model to support browsing in hypermedia information systems. Proceedings of Intelligent Multimedia Information Retrieval Systems and Management. RIAO, New York.

[102] Widyantoro, D.H., Yin, J., El Nasr, M., Yang, L., Zacchi, A., Yen, J. (1999). Alipes: A Swift Messenger In Cyberspace. In: Proc. AAAI Spring Symposium Workshop on Intelligent Agents in Cyberspace, Stanford, March 22-24 (1999) 62-67, Available at: http://citeseer.ist.psu.edu/widyantoro99alipes.html (достъпен на 06 юни, 2016)

[103] Xu, J., Croft, W.B. (1996). Query expansion using local and global document analysis. In: 19th Annual International ACMSIGIR Conference on Research and Development in Information Retrieval (SIGIR 1996), pp. 4–11. ACM, Zurich

[104] Димитров, Д., Никовски, Д. (1999). Изкуствен интелект, Второ преработено издание, Издателски комплекс на Технически университет – София

[105] Павлов, Р., Панева, Д., Павлова-Драганова, Л., Драганов, Л. (2006). Дигитални библиотеки с мултимедийно съдържание и приложения в българското културно наследство (Аналитично изследване по договор ИД 8/21.07.2005 г. между ИМИ-БАН и Държавна агенция за информационни технологии и съобщения), стр. 1-100, Available at: http://mdl.cc.bas.bg/Digital_libraries_with_multimedia_content_and_applications_in_Bulgarian_cultural_heritage.pdf (достъпен на 06 юни, 2016)

АВТОРСКА СПРАВКА – ПУБЛИКАЦИИ И УЧАСТИЯ В НАУЧНИ ФОРУМИ

Списък на публикациите

- Turnina A., (2014). A Short Term User Model for Adaptive Search Based on Previous Queries. In proceedings of 18th International Conference on Circuits, Systems, Communications and Computers (CSCC 2014), Santorini Island, Greece, July 17-21, 2014

- Turnina A., (2013). A novel approach for modeling user's short-term interests, based on user queries paper number 'IJCSI-2013-10-1-4812' International Journal of Computer Science Issue, Volume 10, Issue 1, January 2013

- Turnina A., (2012). Digital libraries towards Mobile Web. In proceedings of 2nd Doctoral Symposium, 11-12 October 2012, Sofia, Bulgaria

- Turnina A., (2012). Context ontology for digital library in pervasive environment. In proceedings of 2nd Doctoral Symposium, 11-12 October 2012, Sofia, Bulgaria.

- Turnina A., (2012). ShareTec Mobile Adaptation model. In proceedings of 14th International Conference on Modern Information Technology in the Innovation Processes of Industrial Enterprises, MITIP, 24-26 October 2012, Budapest, Hungary

- Turnina A., (2012). Semantic-oriented architectures and use of ontology for organizing adaptive search in e-learning environments. In Mathematica Balkanica. Available at: http://www.mathbalkanica.info

- Turnina A., (2011). Share.Tec Metadata model. In proceedings of 1st Unite Doctoral Symposium, Bucharest

- Търнина А., (2011). Технологиите на Семантичния Уеб в услуга на електронното обучение. Доклад в сборник от статии на конференция „Електронно, дистанционно или...образованието на 21 век", 6-8 Април 2011

- Търнина А., (2011). Технологиите на Семантичния Уеб в услуга на електронното обучение. Списание на Софийски университет за електронно обучение. Брой 1/2011

- Turnina A., (2011). Metadata – key to effective searching in digital libraries.In proceeding of ED-MEDIA 2011--World Conference on Educational Multimedia, Hypermedia & Telecommunications 27.06-1.07

Списък на научни събития, на които са докладвани резултатите от дисертацията:

- 18th International Conference on Circuits, Systems, Communications and Computers (CSCC 2014), Santorini Island, Greece, July 17-21, 2014
- Blue sky conference, 17 - 19 June 2014, Budapest, Hungary
- Lab Surfing workshop @ Milan, 20th & 21st May 2014, Milano, Italy
- Lab Surfing workshop @ Thessaloniki, 25th & 26th March 2014, Thessaloniki, Greece
- 2nd Doctoral Symposium, 11-12 October 2012, Sofia, Bulgaria.
- 1st Unite Doctoral Symposium, Bucharest 2011
- Международна Конференция „Електронно, дистанционно или...образованието на 21 век“, 6-8 Април 2011
- ED-MEDIA 2011--World Conference on Educational Multimedia, Hypermedia & Telecommunications 27.06-1.07

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ 1

Софтуерен код на Бейсов модел за адаптиране и персонализиране

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<smile version="1.0" id="Network1" numsamples="1000"
discsamples="10000">
      <nodes>
            <cpt id="Stereotype1">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <noisyadder id="Need1">
                  <state id="True" />
                  <state id="False" />
                  <parents>Stereotype1</parents>
                  <dstates>1 1</dstates>
                  <weights>1 1</weights>
                  <parameters>1 0 0 1 0 1</parameters>
            </noisyadder>
            <cpt id="Stereotype2">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <cpt id="Stereotype20">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <noisyadder id="Need2">
```

```
                <state id="True" />
                <state id="False" />
                <parents>Stereotype1 Stereotype2
Stereotype20</parents>
                <dstates>1 1 1 1</dstates>
                <weights>1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <cpt id="Stereotype12">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype21">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype29">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype30">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <noisyadder id="Need3">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype2 Stereotype12 Stereotype20
Stereotype21 Stereotype29 Stereotype30</parents>
                <dstates>1 1 1 1 1 1 1</dstates>
```

```
                <weights>1 1 1 1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
1 0 0 1 0 1</parameters>
          </noisyadder>
          <cpt id="Stereotype13">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <noisyadder id="Need14">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype12 Stereotype13</parents>
                <dstates>1 1 1</dstates>
                <weights>1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <cpt id="Stereotype27">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <noisyadder id="Need15">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype12 Stereotype13
Stereotype27</parents>
                <dstates>1 1 1 1</dstates>
                <weights>1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <cpt id="Stereotype35">
                <state id="True" />
                <state id="False" />
```

```
      <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need27">
      <state id="True" />
      <state id="False" />
      <parents>Stereotype13 Stereotype35</parents>
      <dstates>1 1 1</dstates>
      <weights>1 1 1</weights>
      <parameters>1 0 0 1 1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype31">
      <state id="True" />
      <state id="False" />
      <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need24">
      <state id="True" />
      <state id="False" />
      <parents>Stereotype31</parents>
      <dstates>1 1</dstates>
      <weights>1 1</weights>
      <parameters>1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype32">
      <state id="True" />
      <state id="False" />
      <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need25">
      <state id="True" />
      <state id="False" />
      <parents>Stereotype32</parents>
      <dstates>1 1</dstates>
```

```
      <weights>1 1</weights>
      <parameters>1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype28">
      <state id="True" />
      <state id="False" />
      <probabilities>0.2 0.8</probabilities>
</cpt>
<cpt id="Stereotype34">
      <state id="True" />
      <state id="False" />
      <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need23">
      <state id="True" />
      <state id="False" />
      <parents>Stereotype28 Stereotype34</parents>
      <dstates>1 1 1</dstates>
      <weights>1 1 1</weights>
      <parameters>1 0 0 1 1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype44">
      <state id="True" />
      <state id="False" />
      <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need30">
      <state id="True" />
      <state id="False" />
      <parents>Stereotype44</parents>
      <dstates>1 1</dstates>
      <weights>1 1</weights>
      <parameters>1 0 0 1 0 1</parameters>
```

```
</noisyadder>
<cpt id="Stereotype45">
     <state id="True" />
     1<state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need31">
     <state id="True" />
     <state id="False" />
     <parents>Stereotype45</parents>
     <dstates>1 1</dstates>
     <weights>1 1</weights>
     <parameters>1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype46">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need32">
     <state id="True" />
     <state id="False" />
     <parents>Stereotype46</parents>
     <dstates>1 1</dstates>
     <weights>1 1</weights>
     <parameters>1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype49">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need35">
```

```
                <state id="True" />
                <state id="False" />
                <parents>Stereotype49</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="ES">
                <state id="True" />
                <state id="False" />
                <parents>Need1 Need2 Need3 Need14 Need15 Need27
Need24 Need25 Need23 Need30 Need31 Need32 Need35</parents>
                <dstates>1 1 1 1 1 1 1 1 1 1 1 1 1 1</dstates>
                <weights>1 1 1 1 1 1 1 1 1 1 1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 0
1</parameters>
          </noisyadder>
          <cpt id="Stereotype3">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype4">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype14">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <noisyadder id="Need4">
```

```
                <state id="True" />
                <state id="False" />
                <parents>Stereotype3 Stereotype4
Stereotype14</parents>
                <dstates>1 1 1 1</dstates>
                <weights>1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <cpt id="Stereotype5">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype22">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype23">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype24">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype25">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
```

```
            <cpt id="Stereotype37">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <noisyadder id="Need6">
                  <state id="True" />
                  <state id="False" />
                  <parents>Stereotype5 Stereotype22 Stereotype23
Stereotype24 Stereotype25 Stereotype37</parents>
                  <dstates>1 1 1 1 1 1 1</dstates>
                  <weights>1 1 1 1 1 1 1</weights>
                  <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
1 0 0 1 0 1</parameters>
            </noisyadder>
            <cpt id="Stereotype6">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <cpt id="Stereotype8">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <cpt id="Stereotype17">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <noisyadder id="Need8">
                  <state id="True" />
                  <state id="False" />
```

```
                <parents>Stereotype6 Stereotype8
Stereotype17</parents>
                <dstates>1 1 1 1</dstates>
                <weights>1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <cpt id="Stereotype7">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype18">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <noisyadder id="Need10">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype7 Stereotype18</parents>
                <dstates>1 1 1</dstates>
                <weights>1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <cpt id="Stereotype9">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
          </cpt>
          <cpt id="Stereotype10">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
```

```
            </cpt>
            <cpt id="Stereotype11">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <cpt id="Stereotype26">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <noisyadder id="Need11">
                  <state id="True" />
                  <state id="False" />
                  <parents>Stereotype9 Stereotype10 Stereotype11
Stereotype26</parents>
                  <dstates>1 1 1 1 1</dstates>
                  <weights>1 1 1 1 1</weights>
                  <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 0
1</parameters>
            </noisyadder>
            <cpt id="Stereotype15">
                  <state id="True" />
                  <state id="False" />
                  <probabilities>0.2 0.8</probabilities>
            </cpt>
            <noisyadder id="Need38">
                  <state id="True" />
                  <state id="False" />
                  <parents>Stereotype15</parents>
                  <dstates>1 1</dstates>
                  <weights>1 1</weights>
                  <parameters>1 0 0 1 0 1</parameters>
            </noisyadder>
```

```
<cpt id="Stereotype16">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need18">
     <state id="True" />
     <state id="False" />
     <parents>Stereotype16</parents>
     <dstates>1 1</dstates>
     <weights>1 1</weights>
     <parameters>1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype19">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need39">
     <state id="True" />
     <state id="False" />
     <parents>Stereotype19</parents>
     <dstates>1 1</dstates>
     <weights>1 1</weights>
     <parameters>1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype43">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need29">
     <state id="True" />
```

```
      <state id="False" />
      <parents>Stereotype43</parents>
      <dstates>1 1</dstates>
      <weights>1 1</weights>
      <parameters>1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype51">
      <state id="True" />
      <state id="False" />
      <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need37">
      <state id="True" />
      <state id="False" />
      <parents>Stereotype51</parents>
      <dstates>1 1</dstates>
      <weights>1 1</weights>
      <parameters>1 0 0 1 0 1</parameters>
</noisyadder>
<cpt id="Stereotype42">
      <state id="True" />
      <state id="False" />
      <probabilities>0.2 0.8</probabilities>
</cpt>
<noisyadder id="Need9">
      <state id="True" />
      <state id="False" />
      <parents>Stereotype42</parents>
      <dstates>1 1</dstates>
      <weights>20 1</weights>
      <parameters>1 0 0 1 0 1</parameters>
</noisyadder>
<noisyadder id="LessonPlan">
```

```
                <state id="State0" />
                <state id="State1" />
                <parents>Need1 Need3 Need4 Need6 Need8 Need10 Need11
Need27 Need38 Need18 Need39 Need24 Need25 Need29 Need37
Need9</parents>
                <dstates>1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</dstates>
                <weights>1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0
0 1 1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need5">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype3 Stereotype4</parents>
                <dstates>1 1 1</dstates>
                <weights>1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need7">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype5 Stereotype6
Stereotype8</parents>
                <dstates>1 1 1 1</dstates>
                <weights>1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="PedagogicalDesignPattern">
                <state id="State0" />
                <state id="State1" />
                <parents>Need2 Need5 Need7</parents>
                <dstates>1 1 1 1</dstates>
                <weights>1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
```

```
</noisyadder>
<cpt id="Stereotype33">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<cpt id="Stereotype36">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<cpt id="Stereotype38">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<cpt id="Stereotype39">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<cpt id="Stereotype40">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<cpt id="Stereotype41">
     <state id="True" />
     <state id="False" />
     <probabilities>0.2 0.8</probabilities>
</cpt>
<cpt id="Stereotype47">
     <state id="True" />
```

```
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
            </cpt>
            <cpt id="Stereotype48">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
            </cpt>
            <cpt id="Stereotype50">
                <state id="True" />
                <state id="False" />
                <probabilities>0.2 0.8</probabilities>
            </cpt>
            <noisyadder id="Need12">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype9 Stereotype10 Stereotype11
Stereotype19</parents>
                <dstates>1 1 1 1 1</dstates>
                <weights>1 1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 0
1</parameters>
            </noisyadder>
            <noisyadder id="Need13">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype9 Stereotype10
Stereotype11</parents>
                <dstates>1 1 1 1</dstates>
                <weights>1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
            </noisyadder>
            <noisyadder id="Need16">
                <state id="True" />
```

```
                <state id="False" />
                <parents>Stereotype14</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need17">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype15</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need20">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype17 Stereotype18
Stereotype25</parents>
                <dstates>1 1 1 1</dstates>
                <weights>1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need19">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype16</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need21">
                <state id="True" />
```

```
                <state id="False" />
                <parents>Stereotype17</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need22">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype18</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need36">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype50</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need0">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype4 Stereotype14 Stereotype15
Stereotype16</parents>
                <dstates>1 1 1 1 1</dstates>
                <weights>1 1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 0
1</parameters>
          </noisyadder>
          <noisyadder id="Need34">
                <state id="True" />
```

```
                <state id="False" />
                <parents>Stereotype48</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
           </noisyadder>
           <noisyadder id="EN">
                <state id="True" />
                <state id="False" />
                <parents>Need4 Need5 Need0 Need7 Need13 Need15
Need27 Need16 Need38 Need17 Need18 Need19 Need21 Need22 Need29
Need30 Need31 Need32 Need34 Need37</parents>
                <dstates>1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1</dstates>
                <weights>1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0
0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
           </noisyadder>
           <noisyadder id="Need28">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype36 Stereotype38 Stereotype39
Stereotype40 Stereotype41</parents>
                <dstates>1 1 1 1 1 1</dstates>
                <weights>1 1 1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
0 1</parameters>
           </noisyadder>
           <noisyadder id="IT">
                <state id="True" />
                <state id="False" />
                <parents>Need6 Need7 Need8 Need10 Need20 Need21
Need22 Need28 Need29 Need30 Need31 Need32 Need36 Need9
Need27</parents>
                <dstates>1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</dstates>
```

```
                <weights>1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0
0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need26">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype33</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Need33">
                <state id="True" />
                <state id="False" />
                <parents>Stereotype47</parents>
                <dstates>1 1</dstates>
                <weights>1 1</weights>
                <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="BG">
                <state id="True" />
                <state id="False" />
                <parents>Need11 Need12 Need13 Need39 Need26 Need30
Need31 Need32 Need33</parents>
                <dstates>1 1 1 1 1 1 1 1 1</dstates>
                <weights>2 2 2 2 2 2 2 2 1</weights>
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="NonPedagogicallyStructured">
                <state id="State0" />
                <state id="State1" />
```

```
                <parents>Need2 Need5 Need7 Need13 Need15 Need16
Need17 Need19 Need21 Need22 Need26 Need23 Need28 Need30 Need31
Need32 Need37</parents>

                <dstates>1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1</dstates>

                <weights>1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1</weights>

                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0
0 1 1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>

           </noisyadder>

           <noisyadder id="PedagogicalSciences">

                <state id="True" />

                <state id="False" />

                <parents>Need1 Need7 Need8 Need21</parents>

                <dstates>1 1 1 1 1</dstates>

                <weights>1 1 1 1 1</weights>

                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 0
1</parameters>

           </noisyadder>

           <noisyadder id="LearningDesignUnit">

                <state id="State0" />

                <state id="State1" />

                <parents>Need0 Need12 Need14 Need20 Need32</parents>

                <dstates>1 1 1 1 1 1</dstates>

                <weights>1 1 1 1 1 1</weights>

                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
0 1</parameters>

           </noisyadder>

           <noisyadder id="ResourcesForLearners">

                <state id="State0" />

                <state id="State1" />

                <parents>Need30 Need33 Need34 Need35
Need36</parents>

                <dstates>1 1 1 1 1 1</dstates>

                <weights>1 1 1 1 1 1</weights>
```

```
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1
0 1</parameters>
          </noisyadder>
          <noisyadder id="EducationalManagement">
               <state id="True" />
               <state id="False" />
               <parents>Need26 Need28</parents>
               <dstates>1 1 1</dstates>
               <weights>1 1 1</weights>
               <parameters>1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="ComputerScience">
               <state id="True" />
               <state id="False" />
               <parents>Need39</parents>
               <dstates>1 1</dstates>
               <weights>1 1</weights>
               <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Religion">
               <state id="True" />
               <state id="False" />
               <parents>Need18 Need19</parents>
               <dstates>1 1 1</dstates>
               <weights>20 20 1</weights>
               <parameters>1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="TeacherTrainingInMusic">
               <state id="True" />
               <state id="False" />
               <parents>Need38 Need17 Need24 Need9</parents>
               <dstates>1 1 1 1 1</dstates>
               <weights>1 1 1 1 1</weights>
```

```
                <parameters>1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 0
1</parameters>
          </noisyadder>
          <noisyadder id="TeacherTrainingPhisicalTraining">
               <state id="True" />
               <state id="False" />
               <parents>Need10 Need22 Need25</parents>
               <dstates>1 1 1 1</dstates>
               <weights>1 1 1 1</weights>
               <parameters>1 0 0 1 1 0 0 1 1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="CareerAdvising">
               <state id="True" />
               <state id="False" />
               <parents>Need23</parents>
               <dstates>1 1</dstates>
               <weights>1 1</weights>
               <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="Arts">
               <state id="True" />
               <state id="False" />
               <parents>Need29</parents>
               <dstates>1 1</dstates>
               <weights>1 1</weights>
               <parameters>1 0 0 1 0 1</parameters>
          </noisyadder>
          <noisyadder id="PublishingDesign">
               <state id="True" />
               <state id="False" />
               <parents>Need31</parents>
               <dstates>1 1</dstates>
               <weights>1 1</weights>
```

```
            <parameters>1 0 0 1 0 1</parameters>
        </noisyadder>
        <noisyadder id="LibraryInformationArchive">
            <state id="True" />
            <state id="False" />
            <parents>Need30</parents>
            <dstates>1 1</dstates>
            <weights>1 1</weights>
            <parameters>1 0 0 1 0 1</parameters>
        </noisyadder>
        <noisyadder id="WebDesign">
            <state id="True" />
            <state id="False" />
            <parents>Need32</parents>
            <dstates>1 1</dstates>
            <weights>1 1</weights>
            <parameters>1 0 0 1 0 1</parameters>
        </noisyadder>
    </nodes>
    <extensions>
        <genie version="1.0" app="GeNIe 2.1.380.0"
name="Network1" faultnameformat="nodestate">
            <node id="Stereotype1">
                <name>Stereotype1</name>
                <interior color="e5f6f7" />
                <outline color="000080" />
                <font color="000000" name="Arial" size="8" />
                <position>102 9 184 39</position>
            </node>
            <node id="Need1">
                <name>Need1</name>
                <interior color="e5f6f7" />
                <outline color="000080" />
```

```
      <font color="000000" name="Arial" size="8" />
      <position>257 226 305 256</position>
</node>
<node id="Stereotype2">
      <name>Stereotype2</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>192 12 274 42</position>
</node>
<node id="Stereotype20">
      <name>Stereotype20</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>561 50 643 80</position>
</node>
<node id="Need2">
      <name>Need2</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>311 229 359 259</position>
</node>
<node id="Stereotype12">
      <name>Stereotype12</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1092 12 1174 42</position>
</node>
<node id="Stereotype21">
      <name>Stereotype21</name>
```

```
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>650 50 732 80</position>
</node>
<node id="Stereotype29">
      <name>Stereotype29</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>144 87 226 117</position>
</node>
<node id="Stereotype30">
      <name>Stereotype30</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>236 87 318 117</position>
</node>
<node id="Need3">
      <name>Need3</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>366 227 414 257</position>
</node>
<node id="Stereotype13">
      <name>Stereotype13</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1182 12 1264 42</position>
</node>
```

```
<node id="Need14">
      <name>Need14</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>970 227 1021 259</position>
</node>
<node id="Stereotype27">
      <name>Stereotype27</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1186 51 1268 81</position>
</node>
<node id="Need15">
      <name>Need15</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1024 227 1077 259</position>
</node>
<node id="Stereotype35">
      <name>Stereotype35</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>686 86 768 116</position>
</node>
<node id="Need27">
      <name>Need27</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
```

```
      <position>669 277 722 309</position>
</node>
<node id="Stereotype31">
      <name>Stereotype31</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>326 87 408 117</position>
</node>
<node id="Need24">
      <name>Need24</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>491 276 544 308</position>
</node>
<node id="Stereotype32">
      <name>Stereotype32</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>416 87 498 117</position>
</node>
<node id="Need25">
      <name>Need25</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>547 275 600 308</position>
</node>
<node id="Stereotype28">
      <name>Stereotype28</name>
      <interior color="e5f6f7" />
```

```
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1274 50 1356 80</position>
</node>
<node id="Stereotype34">
      <name>Stereotype34</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>596 86 678 116</position>
</node>
<node id="Need23">
      <name>Need23</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>438 276 488 307</position>
</node>
<node id="Stereotype44">
      <name>Stereotype44</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>374 124 456 154</position>
</node>
<node id="Need30">
      <name>Need30</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>828 275 882 306</position>
</node>
<node id="Stereotype45">
```

```
      <name>Stereotype45</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>464 124 546 154</position>
</node>
<node id="Need31">
      <name>Need31</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>883 275 935 306</position>
</node>
<node id="Stereotype46">
      <name>Stereotype46</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>554 124 636 154</position>
</node>
<node id="Need32">
      <name>Need32</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>946 274 996 305</position>
</node>
<node id="Stereotype49">
      <name>Stereotype49</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>823 124 905 154</position>
```

```
</node>
<node id="Need35">
      <name>Need35</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1104 275 1152 305</position>
</node>
<node id="ES">
      <name>ES</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>62 540 110 570</position>
</node>
<node id="Stereotype3">
      <name>Stereotype3</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>282 12 364 42</position>
</node>
<node id="Stereotype4">
      <name>Stereotype4</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>368 11 450 41</position>
</node>
<node id="Stereotype14">
      <name>Stereotype14</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
```

```
      <font color="000000" name="Arial" size="8" />
      <position>1273 12 1355 42</position>
</node>
<node id="Need4">
      <name>Need4</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>421 227 469 257</position>
</node>
<node id="Stereotype5">
      <name>Stereotype5</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>462 12 544 42</position>
</node>
<node id="Stereotype22">
      <name>Stereotype22</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>739 50 821 80</position>
</node>
<node id="Stereotype23">
      <name>Stereotype23</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>828 50 910 80</position>
</node>
<node id="Stereotype24">
      <name>Stereotype24</name>
```

```
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>917 50 999 80</position>
</node>
<node id="Stereotype25">
      <name>Stereotype25</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1007 51 1089 81</position>
</node>
<node id="Stereotype37">
      <name>Stereotype37</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>865 87 947 117</position>
</node>
<node id="Need6">
      <name>Need6</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>531 227 579 257</position>
</node>
<node id="Stereotype6">
      <name>Stereotype6</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>552 12 634 42</position>
</node>
```

```
<node id="Stereotype8">
      <name>Stereotype8</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>732 12 814 42</position>
</node>
<node id="Stereotype17">
      <name>Stereotype17</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>294 50 376 80</position>
</node>
<node id="Need8">
      <name>Need8</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>641 227 689 257</position>
</node>
<node id="Stereotype7">
      <name>Stereotype7</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>642 12 724 42</position>
</node>
<node id="Stereotype18">
      <name>Stereotype18</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
```

```
      <position>383 50 465 80</position>
</node>
<node id="Need10">
      <name>Need10</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>748 227 803 259</position>
</node>
<node id="Stereotype9">
      <name>Stereotype9</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>822 11 904 41</position>
</node>
<node id="Stereotype10">
      <name>Stereotype10</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>912 12 994 42</position>
</node>
<node id="Stereotype11">
      <name>Stereotype11</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1002 12 1084 42</position>
</node>
<node id="Stereotype26">
      <name>Stereotype26</name>
      <interior color="e5f6f7" />
```

```
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1099 52 1181 82</position>
</node>
<node id="Need11">
      <name>Need11</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>805 230 860 261</position>
</node>
<node id="Stereotype15">
      <name>Stereotype15</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>118 49 200 79</position>
</node>
<node id="Need38">
      <name>Need38</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1260 272 1308 302</position>
</node>
<node id="Stereotype16">
      <name>Stereotype16</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>205 50 287 80</position>
</node>
<node id="Need18">
```

```
      <name>Need18</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1189 227 1242 259</position>
</node>
<node id="Stereotype19">
      <name>Stereotype19</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>474 50 556 80</position>
</node>
<node id="Need39">
      <name>Need39</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1315 272 1368 304</position>
</node>
<node id="Stereotype43">
      <name>Stereotype43</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>284 124 366 154</position>
</node>
<node id="Need29">
      <name>Need29</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>776 274 826 306</position>
```

```
</node>
<node id="Stereotype51">
      <name>Stereotype51</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1003 124 1085 154</position>
</node>
<node id="Need37">
      <name>Need37</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1209 275 1257 305</position>
</node>
<node id="LessonPlan">
      <name>LessonPlan</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>350 526 439 589</position>
</node>
<node id="Need5">
      <name>Need5</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>476 227 524 257</position>
</node>
<node id="Need7">
      <name>Need7</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
```

```
      <font color="000000" name="Arial" size="8" />
      <position>586 227 634 257</position>
</node>
<node id="PedagogicalDesignPattern">
      <name>PedagogicalDesignPattern</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>452 527 552 591</position>
</node>
<node id="Stereotype33">
      <name>Stereotype33</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>503 87 585 117</position>
</node>
<node id="Stereotype36">
      <name>Stereotype36</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>775 87 857 117</position>
</node>
<node id="Stereotype38">
      <name>Stereotype38</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>955 87 1037 117</position>
</node>
<node id="Stereotype39">
      <name>Stereotype39</name>
```

```
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1045 87 1127 117</position>
</node>
<node id="Stereotype40">
      <name>Stereotype40</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1135 87 1217 117</position>
</node>
<node id="Stereotype41">
      <name>Stereotype41</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1225 87 1307 117</position>
</node>
<node id="Stereotype42">
      <name>Stereotype42</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>188 125 270 155</position>
</node>
<node id="Stereotype47">
      <name>Stereotype47</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>643 124 725 154</position>
</node>
```

```
<node id="Stereotype48">
      <name>Stereotype48</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>733 124 815 154</position>
</node>
<node id="Stereotype50">
      <name>Stereotype50</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>915 122 997 152</position>
</node>
<node id="Need9">
      <name>Need9</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>699 227 747 257</position>
</node>
<node id="Need12">
      <name>Need12</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>858 227 913 259</position>
</node>
<node id="Need13">
      <name>Need13</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
```

```
      <position>912 227 969 257</position>
</node>
<node id="Need16">
      <name>Need16</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1079 227 1131 259</position>
</node>
<node id="Need17">
      <name>Need17</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1135 227 1186 257</position>
</node>
<node id="Need20">
      <name>Need20</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>275 274 325 306</position>
</node>
<node id="Need19">
      <name>Need19</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1246 227 1297 259</position>
</node>
<node id="Need21">
      <name>Need21</name>
      <interior color="e5f6f7" />
```

```
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>327 275 381 306</position>
</node>
<node id="Need22">
      <name>Need22</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>382 275 434 306</position>
</node>
<node id="Need36">
      <name>Need36</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1162 273 1210 303</position>
</node>
<node id="Need0">
      <name>Need0</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>199 227 247 257</position>
</node>
<node id="Need34">
      <name>Need34</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1051 276 1104 309</position>
</node>
<node id="EN">
```

```
      <name>EN</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>117 540 165 570</position>
</node>
<node id="Need28">
      <name>Need28</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>724 276 775 308</position>
</node>
<node id="IT">
      <name>IT</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>185 547 233 577</position>
</node>
<node id="Need26">
      <name>Need26</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>607 276 658 306</position>
</node>
<node id="Need33">
      <name>Need33</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>993 275 1046 307</position>
```

```
</node>
<node id="BG">
      <name>BG</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>255 545 303 575</position>
</node>
<node id="NonPedagogicallyStructured">
      <name>NonPedagogicallyStructured</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>559 528 661 591</position>
</node>
<node id="PedagogicalSciences">
      <name>PedagogicalSciences </name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>907 530 996 579</position>
</node>
<node id="LearningDesignUnit">
      <name>LearningDesignUnit </name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>789 524 899 588</position>
</node>
<node id="ResourcesForLearners">
      <name>ResourcesForLearners</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
```

```
      <font color="000000" name="Arial" size="8" />
      <position>669 526 773 590</position>
</node>
<node id="EducationalManagement">
      <name>EducationalManagement</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1281 538 1332 570</position>
</node>
<node id="ComputerScience">
      <name>ComputerScience</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1226 536 1277 568</position>
</node>
<node id="Religion">
      <name>Religion</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1177 541 1228 573</position>
</node>
<node id="TeacherTrainingInMusic">
      <name>TeacherTrainingInMusic</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1101 541 1176 579</position>
</node>
<node id="TeacherTrainingPhisicalTraining">
      <name>TeacherTrainingPhisicalTraining</name>
```

```
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1001 528 1087 579</position>
</node>
<node id="CareerAdvising">
      <name>CareerAdvising</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1342 534 1394 565</position>
</node>
<node id="Arts">
      <name>Arts</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1194 599 1246 630</position>
</node>
<node id="PublishingDesign">
      <name>PublishingDesign</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1337 593 1390 626</position>
</node>
<node id="LibraryInformationArchive">
      <name>LibraryInformationArchive</name>
      <interior color="e5f6f7" />
      <outline color="000080" />
      <font color="000000" name="Arial" size="8" />
      <position>1263 605 1316 638</position>
</node>
```

```
                <node id="WebDesign">
                    <name>WebDesign</name>
                    <interior color="e5f6f7" />
                    <outline color="000080" />
                    <font color="000000" name="Arial" size="8" />
                    <position>1118 596 1171 629</position>
                </node>
            </genie>
        </extensions>
</smile>
```

ПРИЛОЖЕНИЕ 2

Списък на променливите в Бейсов модел за адаптиране и персонализиране

- Променливи на стереотипите

Stereotype1 = „Испански Обучител на обучители във Facultad de Educación (Máster y Doctorado)“

Stereotype2 = „Испански Обучител на обучители във Факултет за подготовка на учители (Facultad de Formación del Profesorado)“

Stereotype3: = „Ирландски Обучител на обучители на учители от средни училища“

Stereotype4 = „Ирландски Обучител на обучители зает с он-лайн обучение на учители“

Stereotype5 = „Италиански Обучител на обучители в Школа за специализация на учители от основни училища (Scuola di Specializzazione per l'Insegnamento Secondario)“

Stereotype6 = „Италиански Обучител на обучители по Образователни науки (Scienze della Formazione)“

Stereotype7 = „Италиански Обучител на обучители в област Физическо възпитание и спорт (Scienze motorie)“

Stereotype8 = „Италиански Обучител на обучители в Докторантска Школа за проблеми на образованието и психо-Педагогическа науката (Scuola di Dottorato riguardante tematiche didattiche e psicopedagogiche)“

Stereotype9 = „Български университетски преподавател в Софийски университет“

Stereotype10 = „Български университетски преподавател в Шуменски университет“

Stereotype11 = „Български университетски преподавател във Великотърновски университет“

Stereotype12 = „Испански университетски преподавател в Институт по образование (Facultad de Educación)“

Stereotype13 = „Испански университетски преподавател в Университет за дистанционно обучение (Universidad de Educación a distancia)“

Stereotype14 = „Ирландски университетски преподавател на про-гимназиални учители“

Stereotype15 = „Ирландски университетски преподавател на гимназиални учители по музика“

Stereotype16 = „Ирландски университетски преподавател на гимназиални учители по религия“

Stereotype17 = „Италиански университетски преподавател по образователни науки (Scienze della Formazione)“

Stereotype18 = „Италиански преподавател по спортни науки (Scienze motorie)“

Stereotype19 = „Български Инструктор от CISCO мрежова академия“

Stereotype20 = „Испански обучител на обучители в Институт за професионално обучение (Instituto de formación Profesional (Grado Medio y Grado Superior))“

Stereotype21 = „Испански обучител в Центрове за начална професионална квалификация (Centros de Cualificación Profesional Inicia)“

Stereotype22 = „Италиански обучител в Институт за начално професионално обучение (Istituto di formazione professionale iniziale)“

Stereotype23 = „Италиански обучител в Организация за професионално обучение (Ente di formazione professionale)“

Stereotype24 = „Италиански обучител в Институт на началното и професионалното обучение (Istituto di formazione iniziale e professionale)“

Stereotype25 = „Италиански обучител от Звено за дистанционно обучение (Ente di formazione a distanza)“

Stereotype26 = „Български Обучител на обучители-специализанти от Департамент за информация и усъвършенстване на учителите“

Stereotype27 = „Испански Координатор в Министерство на образованието и науката (Ministerio de Educación, Política Social y Deporte)“

Stereotype28 = „Испански Координатор в Национален институт по заетостта (Instituto Nacional de Empleo)“

Stereotype29 = „Испански Обучител, зает с фирмено обучение (Empresas de formación)“

Stereotype30 = „Испански Обучител в Центрове за обучение на възрастни (Centros de formación de adultos)“

Stereotype31 = „Испански Обучител в Консерваторията (Conservatorio (Enseñanzas de MÜsica y Danza nivel: Avanzado))“

Stereotype32 = „Испански Обучител в Центрове за обучение по спортното образование (Centros de Formación de Enseñanzas deportivas)“

Stereotype33 = „Български Координатор в Отдел “Докторанти и следдипломна квалификация" към СУ“

Stereotype34 = „Испански Координатор в център за професионално ориентиране (Centro Nacional de Recursos para la Orientación Profesional)“

Stereotype35 = „Испански Обучител в Езиково училище (Escuela Oficial de Idiomas)“

Stereotype36 = „Италиански Координатор в Национална агенция за развитие на образованието (Agenzia Nazionale Sviluppo Autonomia Scolastica)“

Stereotype37 = „Италиански Обучител в Териториални центрове за образование и обучение в зряла възраст (Centro Territoriale Permanente per l'Istruzione e la Formazione in età adulta)“

Stereotype38 = „Италиански Координатор в Асоциация на италианските обучители (Associazione Italiana Formatori)”

Stereotype39 = „Италиански Координатор в Център на инициативните и демократични Учители (Centro Iniziativa Democratica Insegnanti)“

Stereotype40 = „Италиански Координатор в Регионален офис на образованието (Ufficio Scolastico Regionale)“

Stereotype41 = „Италиански Координатор в Институт за развитие на професионалното обучение на работниците (Istituto per lo sviluppo della formazione professionale dei lavoratori)“

Stereotype42 = „Италиански Обучител в Консерваторията (Conservatorio)“

Stereotype43 = „Италиански Обучител в Академия за изящни изкуства (Accademia delle belle arti)“

Stereotype44 = „Библиотекар“

Stereotype45 = „Издател“

Stereotype46 = „Разработчик на съдържание“

Stereotype47 = „Български само-обучаващ се учител/обучител“

Stereotype48 = „Ирландски само-обучаващ се учител/обучител“

Stereotype49 = „Испански само-обучаващ се учител/обучител“

Stereotype50 = „Италиански само-обучаващ учител/обучител“

Stereotype51 = „Стереотип по подразбиране“

- Променливи на нуждите

Need0 = „Управление на курс в платформа за електронно обучение на английски език“.

Need1 = „Подготвяне на присъствен урок за испански учители в област Педагогическа наука“.

Need2 = „Подготвяне на онлайн активност за испански обучаеми“

Need3 = „Подготвяне на присъствено занятие за испански обучаеми“

Need4 = „Подготвяне на присъствено занятие за ирландски обучаеми“

Need5 = „Подготвяне на онлайн активност за ирландски обучаеми“

Need6 = „Подготвяне на присъствен урок за италиански обучаеми“

Need7 = „Подготвяне на онлайн активност на италиански и английски език“

Need8 = „Подготвяне на присъствено занятие за италиански обучаеми в област Педагогическа наука“

Need9 = „Подготвяне на присъствен урок на италиански език в област музика“

Need10 = „Подготвяне на присъствено занятие за италиански обучаеми в област физическо възпитание и спорт“

Need11 = „Подготвяне на присъствено занятие за български обучаеми“

Need12 = „Управление на курс в платформата за електронно обучение на български език“

Need13 = „Подготвяне на доклад, статия или учебен материал за публикуване на български или английски език“

Need14 = „Управление на курс в платформата за електронно обучение на испански език“

Need15 = „Подготвяне на доклад, статия или учебен материал за публикуване на испански и английски език“

Need16 = „Подготвяне на доклад, статия или учебен материал за публикуване на английски език"

Need17 = „Подготвяне на доклад, статия или учебен материал за публикуване на английски в област музика“

Need18 = „Подготвяне на присъствена лекция на английски език в област религия“

Need19 = „Подготвяне на доклад, статия или учебен материал на английски език в област религия“

Need20 = „Управление на курс в платформа за електронно обучение на италиански език“

Need21 = „Подготвяне на доклад, статия или учебен материал за публикуване на английски и италиански език в област Педагогическа наука“

Need22 = „Подготвяне на доклад, статия или учебен материал за публикуване на английски и италиански език в област физическо възпитание и спорт“

Need23 = „Подготвяне за материали за кариерно ориентиране на испански език“

Need24 = „Подготвяне на присъствен урок на испански език в област музика“

Need25 = „Подготвяне на присъствен урок на испански език в област физическо възпитание и спорт“

Need26 = „Подготвяне на документи на български език, свързани с обучителната дейност на докторанти и специализанти“

Need27 = „Подготвяне на присъствен урок за чуждоезиково обучение“

Need28 = „Подготвяне на документи на италиански, свързани с управление на обучението“

Need29 = „Подготвяне на присъствен урок по изкуства на италиански и английски език“

Need30 = „Справка за информационни ресурси и намиране на библиотечна информация на различен език“

Need31 = „Подготвяне за издаване на произведение на различен език“

Need32 = „Разработване на курсове за електронно обучение на различен език“

Need33 = „Изучаване на дадена тема на български език“

Need34 = „Изучаване на дадена тема на английски език“

Need 35 = „Изучаване на дадена тема на испански език“

Need36 = „Изучаване на дадена тема на италиански език“

Need37 = „Преглед на наличните материали“

Need38 = „Подготвяне на присъствена лекция по музика на английски език“

Need39 = „Подготвяне на присъствен урок за български курсисти в област компютърни мрежи“

- Променливи на атрибутите

ES = „Испански“.

EN = „Английски“.

IT = „Италиански“.

BG = „Български“.

LearningDesignUnit = „Учебна единица“.

LessonPlan = „LessonPlan“

PedagogicalDesignPattern = „Педагогически модел на структурата“

Non-Pedagogically Structured = „Не-структуриран педагогически“

ResourceForLearners = „Ресурси за обучаеми“

PedagogicalSciences = „Педагогическа наука“

TeacherTraining-PhisicalTraining = „Обучение на учители - физическо възпитание“

TeacherTrainingInMusic = „Обучение на учители по музика“

Religion = „Религия“

ComputerScience = „Компютърни науки“

EducationalManagement = „Образователен мениджмънт“

CareerAdvising = „Кариерно ориентиране“

WebDesign = „Уеб дизайн“

Arts = „Изкуства“

LibraryInformationArchive = „Библиотечна информация“

PublishingDesign = „Печатно оформление“

ПРИЛОЖЕНИЕ 3

Вектори на стереотипите

Име на стереотип	Описание на стереотип	Вектор $v_i = (I_{il}(w_{il}), I_{ig}(w_{ig}), I_{ir}(w_{ir}), I_{ii}(w_{ii}), I_{ik}(w_{ik}))$
Stereotype1: ES UCM TE	Испански Обучител на обучители във Facultad de Educación (Máster y Doctorado)	ES(1), TeachingGoal (1), TeacherEducator (1), PostgraduateES_1(1),
Stereotype2: ES USC TE	Испански Обучител на обучители във Факултет за подготовка на учители (Facultad de Formación del Profesorado)	ES(1), TeachingGoal(1), TeacherEducator(1), PostgraduateES_2(1),
Stereotype3: IR SST TE	Ирландски Обучител на обучители на учители от средни училища	EN(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIE_1(1),
Stereotype4: IR OTT TE	Ирландски Обучител на обучители зает с он-лайн обучение на учители	EN(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIE_2(1),
Stereotype5: IT SSIS TE	Италиански Обучител на обучители в Школа за специализация на учители от основни училища (Scuola di Specializzazione per l'Insegnamento Secondario)	IT(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIT_1(1), PedagogicalScience(1)
Stereotype6: IT SF TE	Италиански Обучител на обучители по Образователни науки (Scienze della Formazione)	IT(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIT_2(1), PedagogicalScience(1)
Stereotype7: IT SM TE	Италиански Обучител на обучители в област Физическо	IT(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIT_3(1), TeacherTraining-PhisicalTraining(1)

	възпитание и спорт (Scienze motorie)	
Stereotype8: IT SD TE	Италиански Обучител на обучители в Докторантска Школа за проблеми на образованието и психо-Педагогическа науката (Scuola di Dottorato riguardante tematiche didattiche e psicopedagogiche)	IT(1), TeachingGoal(1), TeacherEducator(1), PostgraduateIT_4(1), PedagogicalScience(1)
Stereotype9: BG SU ST	Български университетски преподавател в Софийски университет	BG(1), TeachingGoal(1), StudentTeacher(1), UndergraduateBG_1(1),
Stereotype10: BG ShU ST	Български университетски преподавател в Шуменски университет	BG(1), TeachingGoal(1), StudentTeacher(1), UndergraduateBG_3(1),
Stereotype11: BG VTU ST	Български университетски преподавател във Великотърновски университет	BG(1), TeachingGoal(1), StudentTeacher(1), UndergraduateBG_2(1),
Stereotype12: ES FD ST	Испански университетски преподавател в Институт по образование (Facultad de Educación)	ES(1), TeachingGoal(1), StudentTeacher(1), UndergraduateES_1(1),
Stereotype13: ES UNED ST	Испански университетски преподавател в Университет за дистанционно обучение (Universidad de Educación a distancia)	ES(1), TeachingGoal(1), StudentTeacher(1), UndergraduateES_2(1),

Stereotype14: IR PTT ST	Ирландски университетски преподавател на про-гимназиални учители	EN(1), TeachingGoal(1), StudentTeacher(1), UndergraduateIE_1(1),
Stereotype15: IR SMTT ST	Ирландски университетски преподавател на гимназиални учители по музика	EN(1), TeachingGoal(1), StudentTeacher(1), UndergraduateIE_2(1), TeacherTrainingInMusic(1)
Stereotype16: IR SRTT ST	Ирландски университетски преподавател на гимназиални учители по религия	EN(1), TeachingGoal(1), StudentTeacher(1), UndergraduateIE_3(1), Religion(1)
Stereotype17: IT SF ST	Италиански университетски преподавател по образователни науки (Scienze della Formazione)	IT(1), TeachingGoal(1), StudentTeacher(1), UndergraduateIT _1(1), PedagogicalScience(1)
Stereotype18: IT SM ST	Италиански преподавател по спортни науки (Scienze motorie)	IT(1), TeachingGoal(1), StudentTeacher(1), UndergraduateIT _2(1), TeacherTraining-PhisicalTraining(1)
Stereotype19: BG CNA Tr	Български Инструктор от CISCO мрежова академия	BG(1), TeachingGoal(1), Trainer(1), TrainingInstitutionBG_1(1), ComputerScience(1)
Stereotype20: ES IFP TT	Испански обучител на обучители в Институт за професионално обучение (Instituto de formación Profesional (Grado Medio y Grado Superior))	ES(1), TeachingGoal(1), TeacherTrainer(1), TrainingInstitutionES_1(1),
Stereotype21: ES CCPI Tr	Испански обучител в Центрове за начална професионална квалификация (Centros de Cualificación Profesional Inicia)	ES(1), TeachingGoal(1), Trainer(1), TrainingInstitutionES_2(1),
Stereotype22: IT IFPI Tr	Италиански обучител в Институт за начално	IT(1), TeachingGoal(1), Trainer(1), TrainingInstitutionIT_1(1),

	професионално обучение (Istituto di formazione professionale iniziale)	
Stereotype23: IT EFP Tr	Италиански обучител в Организация за професионално обучение (Ente di formazione professionale)	IT(1), TeachingGoal(1), TeacherTrainer(1), TrainingInstitutionIT_2(1),
Stereotype24: IT IFIP Tr	Италиански обучител в Институт на началното и професионалното обучение (Istituto di formazione iniziale e professionale)	IT(1), TeachingGoal(1), TeacherTrainer(1), TrainingInstitutionIT_3(1),
Stereotype25: IT EFD TT	Италиански обучител от Звено за дистанционно обучение (Ente di formazione a distanza)	IT(1), TeachingGoal(1), TeacherTrainer(1), TrainingInstitutionIT_4(1),
Stereotype26: BG SDK TT	Български Обучител на обучители-специализанти от Департамент за информация и усъвършенстване на учителите	BG(1), TeachingGoal(1), TeacherTrainer, FurtherEducationAgencyBG_1(1),
Stereotype27: ES MEPSYD C	Испански Координатор в Министерство на образованието и науката (Ministerio de Educación, Política Social y Deporte)	ES(1), ObservingGoal(1), Coordinator(1), FurtherEducationAgencyES_1(1),
Stereotype28: ES INEM C	Испански Координатор в Национален институт по заетостта (Instituto Nacional de Empleo)	ES(1), ObservingGoal(1), Coordinator(1), FurtherEducationAgencyES_2(1), CareerAdvising(1)
Stereotype29: ES EdF Tr	Испански Обучител, зает с фирмено обучение (Empresas de formación)	ES(1), TeachingGoal(1), Trainer(1), FurtherEducationAgencyES_3(1),
Stereotype30: ES CFA Tr	Испански Обучител в Центрове за обучение на възрастни (Centros de formación de	ES(1), TeachingGoal(1), Trainer(1), FurtherEducationAgencyES_4(1),

	adultos)	
Stereotype31: ES Cons Tr	Испански Обучител в Консерваторията (Conservatorio (Enseñanzas de MÜsica y Danza nivel: Avanzado))	ES(1), TeachingGoal(1), Trainer(1), FurtherEducationAgencyES_5(1), TeacherTrainingInMusic(1)
Stereotype32: ES CFED Tr	Испански Обучител в Центрове за обучение по спортното образование (Centros de Formación de Enseñanzas deportivas)	ES(1), TeachingGoal(1), Trainer(1), FurtherEducationAgencyES_6(1), TeacherTraining-PhisicalTraining(1)
Stereotype33: BG SU C	Български Координатор в Отдел “Докторанти и следдипломна квалификация" към СУ	ES(1), ObservingGoal(1), Coordinator(1), FurtherEducationAgencyES_7(1), EducationalManagement(1)
Stereotype34: ES CNROP C	Испански Координатор в център за професионално ориентиране (Centro Nacional de Recursos para la Orientación Profesional)	ES(1), ObservingGoal(1), Coordinator(1), FurtherEducationAgencyES_8(1), CareerAdvising(1)
Stereotype35: ES EOI Tr	Испански Обучител в Езиково училище (Escuela Oficial de Idiomas)	IT(1), TeachingGoal(1), Trainer(1), FurtherEducationAgencyES_8(1), Languages-Foreign(1)
Stereotype36: IT ANSAS C	Италиански Координатор в Национална агенция за развитие на образованието (Agenzia Nazionale Sviluppo Autonomia Scolastica)	IT(1), TeachingGoal(1), Trainer(1), FurtherEducationAgencyIT_2(1), EducationalManagement(1)
Stereotype37: IT CTP Tr	Италиански Обучител в Териториални центрове за образование и обучение в зряла възраст (Centro Territoriale Permanente per l'Istruzione e la Formazione in età adulta)	IT(1), TeachingGoal(1), Trainer(1), FurtherEducationAgencyIT_3(1),
Stereotype38: IT	Италиански	IT(1), ObservingGoal(1), Coordinator(1),

AIF C	Координатор в Асоциация на италианските обучители (Associazione Italiana Formatori)	FurtherEducationAgencyIT_4(1), EducationalManagement(1)
Stereotype39: IT CIDI C	Италиански Координатор в Център на инициативните и демократични Учители (Centro Iniziativa Democratica Insegnanti)	IT(1), ObservingGoal(1), Coordinator(1), FurtherEducationAgencyIT_5(1), EducationalManagement(1)
Stereotype40: IT USR C	Италиански Координатор в Регионален офис на образованието (Ufficio Scolastico Regionale)	IT(1), ObservingGoal(1), Coordinator(1), FurtherEducationAgencyIT_6(1), EducationalManagement(1)
Stereotype41: IT ISFOL C	Италиански Координатор в Институт за развитие на професионалното обучение на работниците (Istituto per lo sviluppo della formazione professionale dei lavoratori)	IT(1), ObservingGoal(1), Coordinator(1), FurtherEducationAgencyIT_6(1), EducationalManagement(1)
Stereotype42: IT Cons Tr	Италиански Обучител в Консерваторията (Conservatorio)	IT(1), TeachingGoal(1), Trainer(1), FurtherEducationAgencyIT_7(1), TeacherTrainingInMusic(1)
Stereotype43: IT ADBATr	Италиански Обучител в Академия за изящни изкуства (Accademia delle belle arti)	IT(1), TeachingGoal(1), Trainer(1), FurtherEducationAgencyIT_7(1), Arts(1)
Stereotype44: Librarian	Библиотекар	, ObservingGoal(1), Librarian(1), Library(1), 0
Stereotype45: Publisher	Издател	, ObservingGoal(1), Publisher(1), PublishingHouse(1), PublishingDesign(1)
Stereotype46: ContDev	Разработчик на съдържание	, ObservingGoal(1), ContentDeveloper(1), WebAgency(1), WebDesign(1)
Stereotype47: BG Self-l	Български само-обучаващ се учител/обучител	BG(1), LearningGoal(1), , ,

Stereotype48: IR Self-l	Ирландски само-обучаващ се учител/обучител	EN(1), LearningGoal(1), , ,
Stereotype49: ES Self-l	Испански само-обучаващ се учител/обучител	ES(1), LearningGoal(1), , ,
Stereotype50: IT Self-l	Италиански само-обучаващ учител/обучител	IT(1), LearningGoal(1), , ,
Stereotype51: Default	Стереотип по подразбиране	EN(1), TeachingGoal(1), , ,

ПРИЛОЖЕНИЕ 4

Софтуерен код, представящ реализация на избрани операции над Бейсов модел за адаптиране и персонализиране

```
      void InfereceWithBayesNet(char *b); //Функция за извод над
мрежата

      int main()

      {

      char *s = "Stereotype44"; //Подава се даден Стереотип

      InfereceWithBayesNet(s);

      ofstream myfile; //Записва се резултата от изчислението във
файл

      return(DSL_OKAY);

      };

      void InfereceWithBayesNet(char *b) {

      DSL_network theNet;

      theNet.ReadFile("BM.xdsl"); //Прочита се мрежата на Бейс


      theNet.SetDefaultBNAlgorithm(DSL_ALG_BN_LAURITZEN); //Задаване
на алгоритъм за обновяване на мрежата (клъстериращ алгоритъм)


      int ste1 = theNet.FindNode("Stereotype1");

      theNet.GetNode(ste1)->Value()->SetEvidence(1);


      int ste2 = theNet.FindNode("Stereotype2");

      theNet.GetNode(ste2)->Value()->SetEvidence(1);


      int ste3 = theNet.FindNode("Stereotype3");
```

```
theNet.GetNode(ste3)->Value()->SetEvidence(1);

int ste4 = theNet.FindNode("Stereotype4");
theNet.GetNode(ste4)->Value()->SetEvidence(1);

int ste5 = theNet.FindNode("Stereotype5");
theNet.GetNode(ste5)->Value()->SetEvidence(1);

int ste6 = theNet.FindNode("Stereotype6");
theNet.GetNode(ste6)->Value()->SetEvidence(1);

int ste7 = theNet.FindNode("Stereotype7");
theNet.GetNode(ste7)->Value()->SetEvidence(1);

int ste8 = theNet.FindNode("Stereotype8");
theNet.GetNode(ste8)->Value()->SetEvidence(1);

int ste9 = theNet.FindNode("Stereotype9");
theNet.GetNode(ste9)->Value()->SetEvidence(1);

int ste10 = theNet.FindNode("Stereotype10");
theNet.GetNode(ste10)->Value()->SetEvidence(1);

int ste11 = theNet.FindNode("Stereotype11");
theNet.GetNode(ste11)->Value()->SetEvidence(1);
```

```
int ste12 = theNet.FindNode("Stereotype12");
theNet.GetNode(ste12)->Value()->SetEvidence(1);

int ste13 = theNet.FindNode("Stereotype13");
theNet.GetNode(ste13)->Value()->SetEvidence(1);

int ste14 = theNet.FindNode("Stereotype14");
theNet.GetNode(ste14)->Value()->SetEvidence(1);

int ste15 = theNet.FindNode("Stereotype15");
theNet.GetNode(ste15)->Value()->SetEvidence(1);

int ste16 = theNet.FindNode("Stereotype16");
theNet.GetNode(ste16)->Value()->SetEvidence(1);

int ste17 = theNet.FindNode("Stereotype17");
theNet.GetNode(ste17)->Value()->SetEvidence(1);

int ste18 = theNet.FindNode("Stereotype18");
theNet.GetNode(ste18)->Value()->SetEvidence(1);

int ste19 = theNet.FindNode("Stereotype19");
theNet.GetNode(ste19)->Value()->SetEvidence(1);

int ste20 = theNet.FindNode("Stereotype20");
theNet.GetNode(ste20)->Value()->SetEvidence(1);
```

```
int ste21 = theNet.FindNode("Stereotype21");
theNet.GetNode(ste21)->Value()->SetEvidence(1);

int ste22 = theNet.FindNode("Stereotype22");
theNet.GetNode(ste22)->Value()->SetEvidence(1);

int ste23 = theNet.FindNode("Stereotype23");
theNet.GetNode(ste23)->Value()->SetEvidence(1);

int ste24 = theNet.FindNode("Stereotype24");
theNet.GetNode(ste24)->Value()->SetEvidence(1);

int ste25 = theNet.FindNode("Stereotype25");
theNet.GetNode(ste25)->Value()->SetEvidence(1);

int ste26 = theNet.FindNode("Stereotype26");
theNet.GetNode(ste26)->Value()->SetEvidence(1);

int ste27 = theNet.FindNode("Stereotype27");
theNet.GetNode(ste27)->Value()->SetEvidence(1);

int ste28 = theNet.FindNode("Stereotype28");
theNet.GetNode(ste28)->Value()->SetEvidence(1);

int ste29 = theNet.FindNode("Stereotype29");
```

```
theNet.GetNode(ste29)->Value()->SetEvidence(1);

int ste30 = theNet.FindNode("Stereotype30");
theNet.GetNode(ste30)->Value()->SetEvidence(1);

int ste31 = theNet.FindNode("Stereotype31");
theNet.GetNode(ste31)->Value()->SetEvidence(1);

int ste32 = theNet.FindNode("Stereotype32");
theNet.GetNode(ste32)->Value()->SetEvidence(1);

int ste33 = theNet.FindNode("Stereotype33");
theNet.GetNode(ste33)->Value()->SetEvidence(1);

int ste34 = theNet.FindNode("Stereotype34");
theNet.GetNode(ste34)->Value()->SetEvidence(1);

int ste35 = theNet.FindNode("Stereotype35");
theNet.GetNode(ste35)->Value()->SetEvidence(1);

int ste36 = theNet.FindNode("Stereotype36");
theNet.GetNode(ste36)->Value()->SetEvidence(1);

int ste37 = theNet.FindNode("Stereotype37");
theNet.GetNode(ste37)->Value()->SetEvidence(1);
```

```
int ste38 = theNet.FindNode("Stereotype38");
theNet.GetNode(ste38)->Value()->SetEvidence(1);

int ste39 = theNet.FindNode("Stereotype39");
theNet.GetNode(ste39)->Value()->SetEvidence(1);

int ste40 = theNet.FindNode("Stereotype40");
theNet.GetNode(ste40)->Value()->SetEvidence(1);

int ste41 = theNet.FindNode("Stereotype41");
theNet.GetNode(ste41)->Value()->SetEvidence(1);

int ste42 = theNet.FindNode("Stereotype42");
theNet.GetNode(ste42)->Value()->SetEvidence(1);

int ste43 = theNet.FindNode("Stereotype43");
theNet.GetNode(ste43)->Value()->SetEvidence(1);

int ste44 = theNet.FindNode("Stereotype44");
theNet.GetNode(ste44)->Value()->SetEvidence(1);

int ste45 = theNet.FindNode("Stereotype45");
theNet.GetNode(ste45)->Value()->SetEvidence(1);

int ste46 = theNet.FindNode("Stereotype46");
theNet.GetNode(ste46)->Value()->SetEvidence(1);
```

```
	int ste47 = theNet.FindNode("Stereotype47");
	theNet.GetNode(ste47)->Value()->SetEvidence(1);

	int ste48 = theNet.FindNode("Stereotype48");
	theNet.GetNode(ste48)->Value()->SetEvidence(1);

	int ste49 = theNet.FindNode("Stereotype49");
	theNet.GetNode(ste49)->Value()->SetEvidence(1);

	int ste50 = theNet.FindNode("Stereotype50");
	theNet.GetNode(ste50)->Value()->SetEvidence(1);

	int ste51 = theNet.FindNode("Stereotype51");
	theNet.GetNode(ste51)->Value()->SetEvidence(1);

	theNet.UpdateBeliefs();

	const char *a = b;
	int stereotype = theNet.FindNode(a);
	theNet.GetNode(stereotype)->Value()->SetEvidence(0);
	theNet.UpdateBeliefs();

	int attr = theNet.FindNode("IT");

	DSL_sysCoordinates theCoordinates;
theCoordinates.LinkTo(*theNet.GetNode(attr)->Value());
```

```
     theCoordinates[0] = 0; // 0 е индекс на състоянието,
съответстващ на истина

     theCoordinates.GoToCurrentPosition();



     double P = theCoordinates.UncheckedValue(); // Взима се
стойността на вероятността на възел "IT"



     int attr1 = theNet.FindNode("EN");

     theCoordinates.LinkTo(*theNet.GetNode(attr1)->Value());

     theCoordinates[0] = 0;

     theCoordinates.GoToCurrentPosition();

     double P1 = theCoordinates.UncheckedValue();



     int attr2 = theNet.FindNode("ES");

     theCoordinates.LinkTo(*theNet.GetNode(attr2)->Value());

     theCoordinates[0] = 0;

     theCoordinates.GoToCurrentPosition();

     double P2 = theCoordinates.UncheckedValue();



     int attr3 = theNet.FindNode("BG");

     theCoordinates.LinkTo(*theNet.GetNode(attr3)->Value());

     theCoordinates[0] = 0;

     theCoordinates.GoToCurrentPosition();

     double P3 = theCoordinates.UncheckedValue();

     int attr4 = theNet.FindNode("LessonPlan");

     theCoordinates.LinkTo(*theNet.GetNode(attr4)->Value());

     theCoordinates[0] = 0;
```

```
theCoordinates.GoToCurrentPosition();
double P4 = theCoordinates.UncheckedValue();

int attr5 = theNet.FindNode("PedagogicalDesignPattern");
theCoordinates.LinkTo(*theNet.GetNode(attr5)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
double P5 = theCoordinates.UncheckedValue();

int attr6 = theNet.FindNode("NonPedagogicallyStructured");
theCoordinates.LinkTo(*theNet.GetNode(attr6)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
double P6 = theCoordinates.UncheckedValue();

int attr7 = theNet.FindNode("ResourcesForLearners");
theCoordinates.LinkTo(*theNet.GetNode(attr7)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
double P7 = theCoordinates.UncheckedValue();

int attr8 = theNet.FindNode("LearningDesignUnit");
theCoordinates.LinkTo(*theNet.GetNode(attr8)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
double P8 = theCoordinates.UncheckedValue();

int attr9 = theNet.FindNode("PedagogicalSciences");
theCoordinates.LinkTo(*theNet.GetNode(attr9)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
```

```
      double P9 = theCoordinates.UncheckedValue();


      int attr10 =
theNet.FindNode("TeacherTrainingPhisicalTraining");
      theCoordinates.LinkTo(*theNet.GetNode(attr10)->Value());
      theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
      double P10 = theCoordinates.UncheckedValue();


      int attr11 = theNet.FindNode("TeacherTrainingInMusic");
      theCoordinates.LinkTo(*theNet.GetNode(attr11)->Value());
      theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
      double P11 = theCoordinates.UncheckedValue();


      int attr12 = theNet.FindNode("Religion");
      theCoordinates.LinkTo(*theNet.GetNode(attr12)->Value());
      theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
      double P12 = theCoordinates.UncheckedValue();


      int attr13 = theNet.FindNode("ComputerScience");
      theCoordinates.LinkTo(*theNet.GetNode(attr13)->Value());
      theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
      double P13 = theCoordinates.UncheckedValue();


      int attr14 = theNet.FindNode("EducationalManagement");
      theCoordinates.LinkTo(*theNet.GetNode(attr14)->Value());
      theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
      double P14 = theCoordinates.UncheckedValue();
```

```
int attr15 = theNet.FindNode("CareerAdvising");
theCoordinates.LinkTo(*theNet.GetNode(attr15)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
double P15 = theCoordinates.UncheckedValue();

int attr16 = theNet.FindNode("WebDesign");
theCoordinates.LinkTo(*theNet.GetNode(attr16)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
double P16 = theCoordinates.UncheckedValue();

int attr17 = theNet.FindNode("Arts");
theCoordinates.LinkTo(*theNet.GetNode(attr17)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
double P17 = theCoordinates.UncheckedValue();

int attr18 = theNet.FindNode("LibraryInformationArchive");
theCoordinates.LinkTo(*theNet.GetNode(attr18)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
double P18 = theCoordinates.UncheckedValue();

int attr19 = theNet.FindNode("PublishingDesign");
theCoordinates.LinkTo(*theNet.GetNode(attr19)->Value());
theCoordinates[0] = 0; theCoordinates.GoToCurrentPosition();
double P19 = theCoordinates.UncheckedValue();
```

```
     ofstream myfile; //Файл, в който се записват атрибутите, с
които ще се адаптира заявката

     myfile.open("last.txt,");

     if (P >= 0.01)

     myfile << "IT" << endl;


     if (P1 >= 0.01)

     myfile << "EN" << endl;


     if (P2 >= 0.01)

     myfile << "ES" << endl;


     if (P3 >= 0.01)

     myfile << "BG" << endl;


     if (P4 >= 0.01)

     myfile << "LessonPlan" << endl;


     if (P5 >= 0.01)

     myfile << "PedagogicalDesignPattern" << endl;


     if (P6 >= 0.01)

     myfile << "NonPedagogicallyStructured" << endl;


     if (P7 >= 0.01)

     myfile << "ResourcesForLearners" << endl;
```

```
if (P8 >= 0.01)

myfile << "LearningDesignUnit" << endl;


if (P9 >= 0.01)

myfile << "PedagogicalSciences" << endl;


if (P10 >= 0.01)

myfile << "TeacherTrainingPhisicalTraining" << endl;


if (P11 >= 0.01)

myfile << "TeacherTrainingInMusic" << endl;


if (P12 >= 0.01)

myfile << "Religion" << endl;


if (P13 >= 0.01)

myfile << "ComputerScience" << endl;


if (P14 >= 0.01)

myfile << "EducationalManagement" << endl;


if (P15 >= 0.01)

myfile << "CareerAdvising" << endl;


if (P16 >= 0.01)

myfile << "WebDesign" << endl;
```

```
if (P17 >= 0.01)

myfile << "Arts" << endl;


if (P18 >= 0.01)

myfile << "LibraryInformationArchive" << endl;


if (P19 >= 0.01)

myfile << "PublishingDesign" << endl;


myfile.close();

theNet.GetNode(stereotype)->Value()->ClearEvidence();

system("pause");


};
```

ПРИЛОЖЕНИЕ 5

Релации между променливите и техните предшественици в Модела

Възел на Стереотип	Възел на Нужда	Възел на Атрибут
Stereotype1: ES UCM TE	Need1	ES, LessonPlan, PedagogicalSciences
	Need2	ES, PedagogicalDesignPattern, Non-PedagogicallyStructured
Stereotype2: ES USC TE	Need2	ES, PedagogicalDesignPattern, Non-PedagogicallyStructured
	Need3	ES, LessonPlan
Stereotype3: IR SST TE	Need4	EN, LessonPlan
	Need5	EN, PedagogicalDesignPattern, Non-PedagogicallyStructured
Stereotype4: IR OTT TE	Need0	EN, LearningDesignUnit
	Need5	EN, PedagogicalDesignPattern, Non-PedagogicallyStructured
Stereotype5: IT SSIS TE	Need6	IT, LessonPlan
	Need7	IT, EN, PedagogicalDesignPattern, Non-PedagogicallyStructured, PedagogicalScience

Stereotype6: IT SF TE	Need7	IT, EN, PedagogicalDesignPattern, Non-PedagogicallyStructured, PedagogicalScience
	Need8	IT, LessonPlan, PedagogicalScience
Stereotype7: IT SM TE	Need10	IT, LessonPlan, TeacherTraining-PhisicalTraining
Stereotype8: IT SD TE	Need7	IT, EN, PedagogicalDesignPattern, Non-PedagogicallyStructured, PedagogicalScience
	Need8	IT, LessonPlan, PedagogicalScience
Stereotype9: BG SU ST	Need11	BG, LessonPlan
	Need12	BG, LearningDesignUnit
	Need13	BG, EN, Non-PedagogicallyStructured
Stereotype10: BG ShU ST	Need11	BG, LessonPlan
	Need12	BG, LearningDesignUnit
	Need13	BG, EN, Non-PedagogicallyStructured
Stereotype11: BG VTU ST	Need11	BG, LessonPlan

	Need12	BG, LearningDesignUnit
	Need13	BG, EN, Non-PedagogicallyStructured
Stereotype12: ES FD ST	Need3	ES, LessonPlan
	Need14	ES, LearningDesignUnit
	Need15	ES, EN, Non-PedagogicallyStructured
Stereotype13: ES UNED ST	Need14	ES, LearningDesignUnit
	Need15	ES, EN, Non-PedagogicallyStructured
	Need27	ES, EN, IT, LessonPlan
Stereotype14: IR PTT ST	Need0	EN, LearningDesignUnit
	Need4	EN, LessonPlan
	Need16	EN, Non-PedagogicallyStructured
Stereotype15: IR SMTT ST	Need0	EN, LearningDesignUnit

	Need17	EN, Non-PedagogicallyStructured, TeacherTrainingInMusic
	Need38	EN, LessonPlan, TeacherTrainingInMusic
Stereotype16: IR SRTT ST	Need0	EN, LearningDesignUnit
	Need18	EN, LessonPlan, Religion
	Need19	EN, Non-PedagogicallyStructured, Religion
Stereotype17: IT SF ST	Need8	IT, LessonPlan, PedagogicalScience
	Need20	IT, LearningDesignUnit
	Need21	IT, EN, Non-PedagogicallyStructured, PedagogicalScience
Stereotype18: IT SM ST	Need10	IT, LessonPlan, TeacherTrainingPhisicalTraining
	Need20	IT, LearningDesignUnit
	Need22	IT, EN, Non-PedagogicallyStructured, TeacherTraining-PhisicalTraining
Stereotype19: BG CNA Tr	Need12	BG, LearningDesignUnit
	Need39	BG, LessonPlan, ComputerScience
Stereotype20: ES IFP TT	Need2	ES, PedagogicalDesignPattern, Non-PedagogicallyStructured

	Need3	ES, LessonPlan
Stereotype21: ES CCPI Tr	Need3	ES, LessonPlan
Stereotype22: IT IFPI Tr	Need6	IT, LessonPlan
Stereotype23: IT EFP Tr	Need6	IT, LessonPlan
Stereotype24: IT IFIP Tr	Need6	IT, LessonPlan
Stereotype25: IT EFD TT	Need7	IT, EN, PedagogicalDesignPattern, Non-PedagogicallyStructured, PedagogicalScience
	Need20	IT, LearningDesignUnit
Stereotype26: BG SDK TT	Need11	BG, LessonPlan
Stereotype27: ES MEPSYD C	Need15	ES, EN, Non-PedagogicallyStructured
Stereotype28: ES INEM C	Need23	ES, Non-PedagogicallyStructured, CareerAdvising
Stereotype29: ES EdF Tr	Need3	ES, LessonPlan
Stereotype30: ES CFA Tr	Need3	ES, LessonPlan
Stereotype31: ES Cons Tr	Need24	ES, LessonPlan, TeacherTrainingInMusic

Stereotype32: ES CFED Tr	Need25	ES, LessonPlan, TeacherTraining-PhisicalTraining
Stereotype33: BG SU C	Need26	BG, Non-PedagogicallyStructured, EducationalManagement
Stereotype34: ES CNROP C	Need23	ES, Non-PedagogicallyStructured, CareerAdvising
Stereotype35: ES EOI Tr	Need27	ES, EN, LessonPlan
Stereotype36: IT ANSAS C	Need28	IT, Non-PedagogicallyStructured, EducationalManagement
Stereotype37: IT CTP Tr	Need6	IT, LessonPlan
Stereotype38: IT AIF C	Need28	IT, Non-PedagogicallyStructured, EducationalManagement
Stereotype39: IT CIDI C	Need28	IT, Non-PedagogicallyStructured, EducationalManagement
Stereotype40: IT USR C	Need28	IT, Non-PedagogicallyStructured, EducationalManagement
Stereotype41: IT ISFOL C	Need28	IT, Non-PedagogicallyStructured, EducationalManagement
Stereotype42: IT Cons Tr	Need9	IT, LessonPlan, TeacherTrainingInMusic

Stereotype43: IT ADBATr	Need29	IT, EN, LessonPlan, Arts
Stereotype44: Librarian	Need30	IT, ES, EN, BG, Non-PedagogicallyStructured, ResourceForLearners, LibraryInformationArchive
Stereotype45: Publisher	Need31	IT, ES, EN, BG, Non-PedagogicallyStructured, PublishingDesign
Stereotype46: ContDev	Need32	IT, ES, EN, BG, Non-PedagogicallyStructured, LearningDesignUnit, WebDesign
Stereotype47: BG Self-l	Need33	BG, ResourceForLearners
Stereotype48: IR Self-l	Need34	EN, ResourceForLearners
Stereotype49: ES Self-l	Need35	ES, ResourceForLearners
Stereotype50: IT Self-l	Need36	IT, ResourceForLearners
Stereotype51: Default	Need37	EN, LessonPlan, Non-PedagogicallyStructured, ResourceForLearners