



БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ
ИНСТИТУТ ПО ИНФОРМАЦИОННИ И
КОМУНИКАЦИОННИ ТЕХНОЛОГИИ



Даниела Ананиева Орозова

Приложение на науката за данните
във виртуалното образователно пространство

ДИСЕРТАЦИЯ

за придобиване на научната степен „доктор на науките“

по професионално направление 4.6. Информатика и компютърни науки

София, 2023 г.

СЪДЪРЖАНИЕ

| | |
|--|----|
| Таблица на съответствията на използваните термини от английски език | 4 |
| УВОД | 5 |
| 1. Науката за данните в обучението | 5 |
| 2. Цел и задачи на дисертационния труд | 9 |
| ГЛАВА 1. ВИРТУАЛНО ОБРАЗОВАТЕЛНО ПРОСТРАНСТВО | 11 |
| 1.1. Архитектура на данните на виртуално образователно пространство | 13 |
| 1.2. Виртуално образователно пространство и концепцията Интернет на нещата | 16 |
| 1.3. Виртуалното образователно пространство и големите данни | 18 |
| 1.4. Виртуалното образователно пространство на нововъзникващите технологии и рискове | 23 |
| 1.5. Модели за анализ на предметната област | 29 |
| 1.5.1. Модели, основани на свойства | 29 |
| 1.5.2. Изпълними модели | 29 |
| 1.5.3. Моделиране чрез апарата на обобщените мрежи | 31 |
| ГЛАВА 2. СРЕДСТВА НА ВИРТУАЛНОТО ОБРАЗОВАТЕЛНО ПРОСТРАНСТВО | 35 |
| 2.1. Интеграция на средства за извличане на знания в образователното пространство | 37 |
| 2.2. Семантични знания за пространството | 39 |
| 2.3. Интеграция на персонални асистенти към виртуалното образователно пространство | 40 |
| 2.4. Инструменти за подготовка и изчистване на данните в пространството | 42 |
| 2.5. Средства за коректност на данните, търсене на данни и контрол на достъпа в образователното пространство | 43 |
| 2.6. Инструменти на интелигентните обучаващи пространства | 43 |
| 2.6.1. Оценка на работата и знанията на обучаемите | 43 |
| 2.6.2. Препоръчващи системи | 44 |
| 2.6.3. Инструменти за подпомагане на работата на преподавателите | 44 |
| 2.6.4. Стандарти за съвместимост на учебно съдържание и образователни услуги | 44 |
| ГЛАВА 3. МОДЕЛИ ЗА АНАЛИЗ НА ДАННИ ВЪВ ВИРТУАЛНОТО ОБРАЗОВАТЕЛНО ПРОСТРАНСТВО | 48 |
| 3.1. Използване на електронни таблици за аналитични изследвания на данните за обучаемите | 52 |
| 3.2. Аналитични изследвания на данните в обучаващото пространство със системата „Orange“ | 54 |
| 3.2.1. Проблеми при подготовка на данните за анализ | 56 |
| 3.2.2. Оценяване и прогнозиране на знанията на обучаемите във виртуалното образователно пространство | 60 |
| 3.3. Уеб метрики за оценяване дейностите на обучаемите във виртуалното образователно пространство | 73 |
| 3.4. Моделиране на потребителя в образователното пространство | 78 |
| 3.5. Когнитивен агент в среда за електронно обучение | 90 |
| 3.6. Модели за йерархично многокомпонентно оценяване | 97 |

| | |
|---|-----|
| ГЛАВА 4. ОБОБЩЕНОМРЕЖОВИ МОДЕЛИ ВЪВ ВИРТУАЛНО ОБРАЗОВАТЕЛНО ПРОСТРАНСТВО | 105 |
| 4.1. Модел на процесите на персонализиране и използване на среда за обучение | 105 |
| 4.2. Модел на процеса на прилагане на средства за извличане на знания от данните в среди за обучение чрез апарата на обобщените мрежи | 113 |
| 4.3. Обобщеномрежов модел на процес за многокомпонентно оценяване | 116 |
| 4.4. Обобщеномрежов модел на процесите в проектно-базирано обучение | 122 |
| 4.5. Модел за игровизация на електронен курс за обучение | 130 |
| 4.6. Модели за осигуряване на качеството и акредитация на висшето образование | 137 |
| 4.6.1. Слабо-централизиран модел за акредитация | 139 |
| 4.6.2. Силно-централизиран модел за акредитация | 140 |
| 4.6.3. Обобщеномрежов модел на процесите на обработка на данни при самооценяване във висше училище | 141 |
| ГЛАВА 5. РЕШЕНИЯ НА НАУЧНОПРИЛОЖНИ ПРОБЛЕМИ В ОБУЧЕНИЕТО | 146 |
| 5.1. Учебна програма за анализ на данните | 146 |
| 5.2. Инструменти за извличане на знания чрез примери | 152 |
| 5.2.1. Класификация и прогнозиране | 153 |
| 5.2.2. Регресионен анализ | 157 |
| 5.2.3. Асоциативен анализ | 157 |
| 5.2.4. Клъстерен анализ | 159 |
| 5.2.5. Анализ на шумове | 160 |
| 5.3. Въвеждане на онтологии в обучението на студенти | 161 |
| 5.3.1. Методология за проектиране на онтология | 162 |
| 5.3.2. Прилагане на езика OWL за изграждане структурата на онтология | 162 |
| 5.4. Виртуалното образователно пространство и науката за данните | 165 |
| ЗАКЛЮЧЕНИЕ | 166 |
| Приноси на дисертационния труд | 168 |
| Декларация за оригиналност на резултатите | 170 |
| Списък на научните публикации по темата на дисертационния труд | 171 |
| БИБЛИОГРАФИЯ | 178 |

Таблица на съответствията на използваните термини от английски език

| Термин от английски език | Пояснение |
|--|--|
| Big Data | големи данни |
| Content Management System (CMS) | система за управление на съдържанието |
| Cyber-Physical Social System (CPSS) | кибер-физическа социална система |
| Cyber-Physical Social Educational System | кибер-физическа социална образователна система |
| Data Analytics | анализи на данни |
| Data Science | наука за данните |
| Data Mining | извличане на знания от данните |
| Data Cleaning | почистване на данните |
| Data Integration | интеграция на данните |
| Data Reduction | редуциране на данните |
| Data Transformation | трансформация на данните |
| E-Learning | електронно обучение |
| Educational Data Mining | извличане на образователни данни |
| Emerging technologies | нововъзникващи технологии |
| Executable models | изпълними модели |
| Fuzzy sets and systems | размити множества и системи |
| Generalized Net (GN) | обобщени мрежи (OM) |
| Internet of Things (IoT) | интернет на нещата |
| Knowledge based systems | системи, основани на знания |
| Knowledge representation | представяне на знания |
| Learning Analytics | учебни аналитики |
| Learning Management System (LMS) | система за управление на учебния процес |
| Learning System | учебна система |
| Machine Learning | машинно самообучение |
| Multi-agent systems | мултиагентни системи |
| Natural language processing | обработка на естествен език |
| Ontologies engineering | онтологично инженерство |
| Orange Data Mining system | софтуер за машинно обучение и анализ на данни |
| Outliers | отличителни стойности |
| Property-based models | модели, основани на свойства |
| Reinforcement Learning | подсилващо обучение |
| Recommendation systems | препоръчващи системи |
| Sharable Content Object Reference Model (SCORM) | спецификация за стандартизирано електронно учебно съдържание |
| Semantic Web | семантичен уеб |
| Supervised and Unsupervised learning | контролирано и неконтролирано обучение |

УВОД

1. Науката за данните в обучението

Поради специфичните характеристики на NET-поколението, обучението с помощта на виртуални образователни платформи е един от предпочитаните начини за получаване на знания. *Виртуалните образователни пространства* [49] са информационни и социални пространства, които съчетават различни технологии, средства и подходи за осигуряване на обучаващи материали и образователни услуги за различни потребителски групи. Тези данни в образователното пространство непрекъснато нарастват. Университети от различни страни изграждат инфраструктури за съхранение и обработка на такива данни. Siemens & Long [184] определят *Data analytics in Education* като "измерване, събиране, анализ и отчитане на данните за обучаваните и техните контексти с цел разбиране и оптимизиране на обучението и околната среда, в която се случва." Целите на този анализ се определят [61] като: мониторинг, прогнозиране, индивидуализация, намеса в обучаващия процес, оценка и препоръки на обучаемия и обратна връзка с него.

Термините *Big Data*, *Data Analytics* и *Data Mining* описват както самите данни, така и технологиите за събиране, обработка, управление на данните и методите за анализ [62]. *Data Mining* е процеса на търсене на взаимосвързани и предварително неизвестни данни, но практически полезни и необходими за взимане на решения в различни сфери на човешките дейности.

Big Data Analytics се явява развитие на концепцията *Data Mining* и развитие на решаваните задачи, методи и технологии на обработка, източници на данни, сфери на приложение. От появата на *Data Mining* до настъпване на ерата на *Big Data*, се изменя обема на анализираните данни, появяват се нови технологии за обработка и високопроизводителна техника, концепции за паралелна обработка на данни и техни различни програмни реализации.

През 1998 г. Чикио Хаяши [103] въвежда термина *Data science* като: "нова, интердисциплинарна концепция, с три аспекта: проектиране на данни, събиране и анализ". Появява се науката за данните, която използва различни методи, алгоритми, инструменти и техники за анализ и извлечение на знания от данни и изисква нов поглед и съвместно прилагане на знания от областта на математиката, статистиката, програмирането, бизнес анализа и редица други утвърдени технологии, фигура 1.

Науката за данните съчетава множество подходи и техники, свързани с анализ на данни, откриване на знания, машинно обучение, изкуствен интелект, програмиране, комуникация др. [69]. Тя се развива като резултат от интеграцията на традиционните подходи за работа с бази от данни със съвременни области като: обектно-ориентирано програмиране, бизнес информационни системи, автоматизирано извличане на знания (*Data Mining*), хипермедия и други.

Науката за данните включва и процесите по изчистване и интеграция на данните, избор и трансформация на данни, извличане на знания, техния анализ, оценяване и представяне. Може да се каже, че *Data science* е „сплав“ от различни дисциплини, технологии и средства за анализ на данни. Тя се занимава с обработка и трансформиране на данни в ясна и полезна информация за вземане на решения. *Data Science* [74] се явява основно средство за откриване и оползотворяване на потенциала на Big Data.



Фиг.1. Възникване на понятието Data Science.

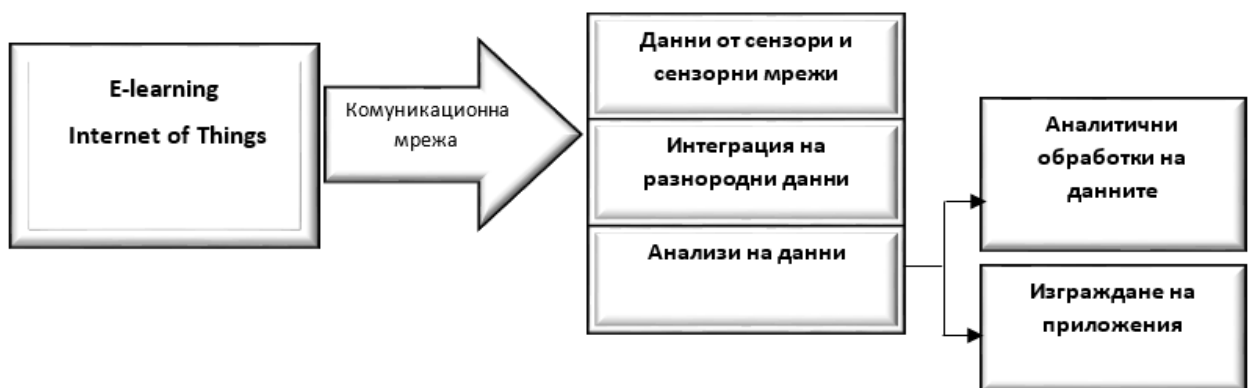
Науката за данните (*Data science*) е „концепция за обединяване на статистика, анализ на данни и свързани с тях методи, за да се разберат и анализират действителните явления с данни“ [183]. Според [58] „*Data science* е нова интердисциплинарна област, която синтезира и надгражда статистика, информатика, компютри, комуникация, управление и социология за изучаване на данни и техните среди (включително домейни и други контекстуални аспекти, като организационни и социални аспекти), за да трансформира данните в откритие, прогноза, услуга, предложение за вземане на решения, модел, парадигма, инструмент или система“. Популярността на “*Data science*” се увеличава с всеки изминал ден, според данните на Google Trends (<https://trends.google.com/trends/>). В допълнение към науката за данни, тенденциите на популярност на областите: „Анализ на данни“, „Извличане на данни“, „Големи данни“, „Машинно обучение“ се увеличават непрестанно. Терминът „*Data analysis*“ [58] се отнася до обработката на данни чрез конвенционални (класически статистически, емпирични или логически) теории, технологии и инструменти за извличане на полезна информация за практически цели. Терминът „*Data analytics*“, от друга страна, се отнася до теориите, технологиите, инструментите и процесите, които позволяват задълбочено разбиране на изследваните данни [58].

Изследванията показват, че организациите, които използват анализ на данните, основан на изкуствен интелект и машинно обучение, нови подходи за управление на информация и приложения за големи данни, генерират подобрения в своята бизнес ефективност [69]. Днес научните изследвания се движат много бързо в посока на насърчаване на изследвания, където данните, събрани от различни източници, чиито определения и формати са различни, трябва да бъдат интегрирани и анализирани.

Събирането на данни за анализ на обучението се отнася до целия процес. Включват се всички данни, които могат да бъдат събрани и анализирани в областта на образованието от всички учебни дейности. Това са много и разнообразни данни. Международната организация по стандартизация IMG Global Learning Consortium (IMG Global) класифицира данните в пет групи: данни за учебната дейност; данни за учебното съдържание; данни за профила на обучаемия; оперативни данни; данни, свързани с кариерното развитие. Това е предпоставка за появата на *Learning Analytics* [61], нов клон на анализ на данни, който включва начертани цели и методи от образователни и психологически изследвания.

В книгата "Уеб анализ 2.0" [123] се дефинира *Digital Analytics* като „непрекъснато анализиране на количествени и качествени данни от уеб пространството, което да подобри онлайн преживяването на потенциалните клиенти и така да доведе до желаните резултати“ [123]. Въведен е подход на разделяне на данните на различен принцип, който е използван по-късно при разработването на Google Analytics. Популяризирането на социалните мрежи поставя началото и на съвременния социален анализ *Social Analytics*, определян като "наблюдение, анализ, измерване и тълкуване на цифровите взаимодействия и взаимоотношения на хора, теми, идеи и съдържание" [189]. Изследват се социалните взаимодействия и тяхното влияние в процеса на учене.

В съвременния живот, данните, генерирани от IoT сензорите, се предават чрез мрежата и интегрираните данни се анализират [208], функционален изглед на процеса е показан на фигура 2.



Фигура: 2. Интернет на нещата и науката за данните

Днес сме свидетели на широко навлизане на средствата за извличане на знания в различни сфери на съвременния живот с цел анализ и прогнозиране. Извличането на образователни данни, наричано още *Educational Data Mining*, е дефинирано като „дисциплина, занимаваща се с разработването на методи за изследване на уникалните и все по-мощни данни, които идват от образователните среди, и използването на тези методи за по-добро разбиране на обучаемите и на средата, в която те учат“ [114]. Интегрирайки средствата за извличане на знания от данните в обучаващите среди се търсят тенденции в развитието на процесите на електронното и дистанционното обучение, и нови възможности за персонализирано овладяване на ключови знания, умения и компетенции.

Основни задачи и търсене на решения за всяко съвременно висше училище са:

- Планиране и подобряване на учебните програми и процеса на обучение, изпитване, дипломиране и израстване на обучаемите в обучаващи.

- Провеждане на анализи относно степента на придобиване и забравяне на знанията на обучаемите в различни интервали от време и сравняване на показателите през годините.

- Оптимизиране на техниките за тестване на знанията на обучаемите и предприемане подходящ вид намеса в дейността им.

- Идентифициране на обучаемите в риск, идентифициране на профила на студентите, които могат да продължат успешно и завършат магистърски специалности или да продължат обучението си в други курсове.

- Може да бъде предвидено, отпадане на студенти или прехвърлянето им в друга специалност и тяхното академично развитие да бъде подпомогнато чрез различни дейности като: допълнителни курсове, въвеждане на нови интерактивни занимания с цел продължаване на обучението, консултации. Според [112] критичен период е между втората и третата година от следването. Различни техники за анализ на данните могат да се приложат в търсене на важните фактори за повишаване на успеваемостта по време на обучението и факторите допринасящи за отпадане на студенти от обучението.

- На основата на класификационна функция може да се определи профила на различни групи преподаватели. По-доброто разбиране на поведението им, може да позволи на ръководителите на организацията да създаде бизнес план за повишаване на тяхната активност и удовлетвореност.

- Подходящ модел на студентите, които се обучават в едно висше училище може да подпомогне процеса на търсене на кандидат-студенти, като отговори на въпроса коя група от потенциални студенти е най-желателна за обучение в това училище. За целта могат да се прилагат техники, базирани на “регресия” и “класификация”.

Анализирайки данните от потребителските профили, е възможно да се търсят начини за установяване и измерване на различни лични умения на потребителите, които влияят на ученето, като: желанието за промяна или усъвършенстване на неща, които другите хора приемат за даденост, способността да се правят връзки между данни, които на пръв поглед нямат общо, способността да се търсят по-добри решения, въпреки че вече съществуват задоволителни решения, способността да се работи с големи обеми информация, както и други качества и лични умения. На базата на изведените знания, могат да се предприемат конкретни действия от обучаващата среда за адаптиране и персонализиране на обучението.

С навлизането на Интернет на нещата в нашия живот нараства необходимостта от анализ на данните и извличане на знания. За по-добро насочване на обучаемите в обучаващия процес системите могат да допълват своите данни от социални мрежи, браузъри, сензорни и др. На база на това те могат по-добре да предвидят потенциалната загуба на обучаеми; могат да определят кои дейности ще се предпочитат и каква ще е тяхната ефективност за съответния тип обучаван и т.н.

Един от основните проблеми на съвременните обучаващи среди, е че те трябва да се синхронизират с потребителя, за да постигнат различни образователни аспекти като: гъвкавост, повторно използване на учебно съдържание и персонализирано представяне на учебния материал. Тези среди трябва да са в състояние да предвидят нуждите, интересите

на потребителя, причините за по-нататъшни действия и да предлагат планове за постигане на желаната от потребителя цел. За да могат да действат по такъв интелигентен начин, те трябва да са способни да придобиват знания и да правят изводи на база на натрупани данни, основани на действията на потребителя и неговите интереси. С други думи, трябва да могат да усетят поведението на потребителя и да адаптират собственото си поведение, за да единоедействат с него. От друга страна интересът е насочен и към оптимизиране на достъпа до големи обеми от данни в съвременно образователно пространство. Обработката и съхранението на големите данни изисква нов поглед и съвместно прилагане на редица технологии. Новите информационни технологии допринасят за изследване, анализиране и развиване на обучаващите среди като непрекъснато наблюдават и измерват различни аспекти на средата, в която се провежда обучението, произвеждайки големи количества данни.

2. Цел и задачи на дисертационния труд

Целта на изследването са теоретични обобщения на процесите на наблюдение и анализ на данните от динамичното взаимодействие на обекти в обучаваща среда и създаване на методи и модели за решаване на научни или научноприложни проблеми във виртуалното образователно пространство.

За постигане на целта са поставени следните четири **задачи**:

- Анализ на прилагането на средства за извличане на знания от данните в обучаващите пространства и търсене на решения за персонализиране на електронното и дистанционно обучение.
- Създаване на методи за оценяване и прогнозиране на знанията, уменията и компетентностите на обучаемите във виртуалното образователно пространство.
- Създаване на модели, като резултат от теоретични обобщения на процесите на наблюдение и анализ на дейностите на обучаемите, свързани с Big Data Analytics, Data Mining, Web метрики, обобщени мрежи, машинно обучение и размита логика.
- Въвеждане на базови модули и инструменти на науката за данните за решения на научноприложни проблеми в обучението.

Дисертационният труд на тема „Приложение на науката за данните във виртуалното образователно пространство” включва 49 фигури, библиографията обхваща 220 източника, от които 197 на английски език.

Текстът е организиран в увод, пет глави и заключение. В **увода** се прави преглед на основните понятия и концепции на изследваната област. Дефинира се целта на дисертационния труд и свързаните с нейното постигане задачи. В **първа глава** се разглежда областта на виртуалното образователно пространство, основните характеристики и възможности на такъв тип инфраструктура и мотивацията за проведената научноизследователска работа. **Втора глава** представя състоянието, предлагани решения, обща характеристика и възможности за интеграция на средства и инструменти към виртуалното образователно пространство. В **трета глава** са предложени методи и техники за моделиране на процесите в образователното пространство, с приложение на инструменти за извличане на данни в образователен контекст (Educational

Data Mining). В **четвърта глава** се разглеждат работни процеси във виртуалното образователно пространство и техни формални модели чрез апарата на обобщените мрежи. **Пета глава** представя практики и тенденции при навлизането на науката за данните (DataScience) във висшето образование. **Заключението** обобщава резултатите и дава насоки за продължаване на изследванията по темата на дисертационния труд. Приложени са: Декларация за оригиналност на резултатите, Приноси на дисертационния труд, Списък на публикациите по темата на дисертационния труд.

Благодарности

Изказвам своите благодарности на всички колеги и сътрудници в програми, проекти и договори, без които резултатите в дисертационния труд не биха били възможни.

Благодаря на семейството си за постоянната им подкрепа и всеотдайност, без които малките трудности се превръщат в непреодолими пречки.

ГЛАВА 1. ВИРТУАЛНО ОБРАЗОВАТЕЛНО ПРОСТРАНСТВО

Виртуалното образователно пространство се дефинира като среда, която интегрира хетерогенни информационни технологии и различни педагогически подходи за доставка на учебни материали и образователни услуги за различни целеви групи, независимо от времето и пространството. Обхваща различни образователни инструменти и ресурси, които се използват в дигитална форма, за да се подобри образователният процес.

Съдържанието, което предлагат образователните пространства е структурирано спрямо избраната цел. Ресурсите са разнообразни по вид, могат да се импортират и експортират и многократно да се използват в различен образователен контекст. Обикновено предоставят възможност за обучение на неограничен брой потребители, както и обособяването им в групи за сътрудничество и комуникация. Важна функционалност е проследяването на образователния напредък на обучаемите, тяхното оценяване и сертифициране. Елемент от съществено значение за успеха на пространствата е интерактивността.

Виртуално образователно пространство [130], може да следи непрекъснато случващото се в него, да предоставя възможности за общуване на разположените в него компоненти, да прави изводи и взема решения и да оперира съгласно тях.

Виртуално образователно пространство е контекстно-зависимо [17]. Според [72] контекст е цялата информация, която може да се използва за определяне на състоянието на човек, място или предмет, които могат да взаимодействат помежду си. В [73] се твърди, че една система е контекстно-зависима, когато използва контексти за доставяне на услуги и информация, като значимостта се определя от целта на потребителя. В разглеждания случай контекст-зависимостта е способност на обучаващата система да идентифицира и интерпретира промени (събития) в нейната среда и да предприема ответни действия.

Основните действия са персонализация и адаптация. *Персонализацията* е способността на системата да се адаптира към индивидуалните особености, желания, намерения и цели на потребителя. *Адаптацията* е способността на системата да се адаптира към другите характеристики на контекста като например област на познание, учебен предмет, видове устройства, използвани от крайните потребители и т.н..

Виртуалните образователните пространства варират от системи, предоставящи базови възможности, до комплексни системи с голям набор от функции и с възможности за интеграция с други софтуерни компоненти. Основни елементи на тези среди са системите за управление на учебния процес (Learning Management System, LMS) и системите за управление на съдържанието (Learning Content Management System, LCMS). Обикновено учебните пространства поддържат стандарти за достъпност (W3C, ISO и др.) и оперативна съвместимост (SCORM, IMS и др.). Има няколко организации, чиято цел е дефиниране на спецификации и стандарти като IMS Global Learning Consortium, Advanced Distributed Learning (ADL), ARIADNE, IEEE, ISO и други, с които да предоставят **рамка за архитектури на системи** за електронно обучение, да подпомогнат съвместимостта на електронно учебно съдържание и услуги между различните системи, пакетиране и управление на съдържанието, предоставяне на метаданни за различни абстрактни обекти в

процеса на обучение, управление на курсове и много други. Значителна роля има стандартът Sharable Content Object Reference Model 2004 (SCORM 2004) [185].

Виртуалното образователно пространство съчетава традиционни образователни методи с модерни технологии и интерактивни инструменти, като: уеб базирани платформи, онлайн курсове, виртуални класни стаи, видео лекции, тестове, симулации, дискуссионни форуми и др. Ученето във виртуално образователно пространство насърчава участниците да се запознаят и работят с различни интерактивни инструменти, което е важно умение в дигиталния свят.

Въпреки многобройните предимства, виртуалното образователно пространство също така среща и предизвикателства, като липса на физическо присъствие, потенциално по-ниско ниво на мотивация, необходимост от добра интернет свързаност и подходящи технически устройства и др. Затова е важно да се установи подходящ баланс между виртуалните и традиционните образователни методи за постигане на най-добри резултати в образованието.

1.1. Архитектура на данните на виртуално образователно пространство

Архитектурата на няколко нива на данните във виртуалното образователно пространство, позволява на потребителите и разработчиците да имат независим достъп до данни на различните нива, като интегрира различни средства и технологии за съхранение и обработка на данни и знания.

Разглеждайки архитектурата на виртуалното образователно пространство от гледна точка на данните, акцентът се поставя върху различните нива за представяне на данни и свързаните с тях механизми за събиране и анализ на данни, за извличане на информация и прогнозиране поведението на средата и обучаемите. На фигура 3 е направено обобщение в четири базови нива на представяне на данните, но могат да бъдат разглеждани и други нива в зависимост от прилаганите критерии за съхранение и достъп.



Фигура 3. Функционални нива във виртуалното образователно пространство

Първо ниво: Данни от виртуалното образователно пространство и физическия свят. Източниците на данни обхващат широко поле, обединявайки данни получавани от сензорни датчици, данни от социалните мрежи, база от данни за областта на обучението и много други.

Основната функция на сензорното ниво е събиране, регистриране, трансформиране и пренасяне на различни данни, необходими за оперирането и управлението на пространството. Най-общо, във пространството могат да се поддържат два типа сензори – физически и виртуални. За пространството, физическият свят представлява множество от физически сензори. За разлика от физическите, виртуалните сензори са абстракции. Типични източници на виртуални сензорни данни в пространството са SCORM 2004

Engine и QTI 2. Engine, интегрирани в образователното пространство.

Второ ниво: *Хранилища на данните.* Събирането и съхранението на данни се отнася до целия обучаващ процес. Това ниво предоставя всички данни за състоянието на физическия свят, пренесени във виртуалната среда на пространството.

Хранилището трябва да съхранява освен големи обеми от данни за обучаващия процес и редица други данни като географски координати и използвани технологии, което ще даде възможност да се подготвят различни доклади, статистически отчети, резюмета и прегледи на състоянието и процесите в наблюдаваната среда. Информацията, добита чрез анализ, може да се комбинира с данни, получени от сензорни мрежи или да предостави семантичен контекст на редица свързани събития. Най-общо този слой включва използването на технологии за събиране, съхраняване, обработка на големи данни и инструменти за анализ.

Трето ниво: *Хранилища с аналитични подпространства на данните.*

Данните във виртуалните образователни пространства могат да бъдат разглеждани от различни гледни точки и подлагани на различни видове анализи като [208]:

Predictive Analytics – тази група от технологии за анализ на данни, дава възможности за задълбочено разбиране и изследване на данните, прогнозиране и предвиждане на резултати. Тези анализи в областта на обучението могат да доведат до промяна на различни елементи в процеса на обучение, до смяна на стила на преподаване и да подпомогнат откриването на „*студент в риск*“ за продължаване на обучението.

Adaptive Analytics - технологията осигурява подходящото ниво на преподаване и взаимодействие с обекта на обучение. Тези анализи насочват обучаемия в процеса на учене и могат да доведат до повишаване на резултатите му. Например: определяне на нивото на трудност, отчитайки резултатите и поведението на обучаемия, особено в критични ситуации, анализ на данни, свързани с броя часове, през които лицето е извършвало дейности по конкретен предмет, брой задавани въпроси, изпълнявани текущи дейности и др.

Social Network Analytics - анализът на данните от социалните мрежи, взаимодействието на обучаемите чрез използваната среда за обучение, участие в групи, чатове, електронна поща и дискусии може да окаже влияние на ефективността на обучението, успехи при работа в екипи и съвместно обучение. Колаборативното обучение за голяма част от студентите води до повишаване на разбирането на учебното съдържание.

Discourse Analytics - анализ на данните от *log*-файловете на системата могат да дадат информация кога и колко дълго потребителят е прекарал в средата за обучение, какви са изпълняваните задачи, неговите участия в дискусии и различни други изпълнявани от него дейности. Анализ на съдържанието на използваните документи, коментари в дискусиите и участия във форуми, изисква техники за анализ на текстове на естествен език.

Техники от областта на *Text mining* могат да доставят допълнителна информация за анализ на съдържанието на използваните документи и ресурси, анализ на дискусиите в онлайн текстов чат, както и да се даде оценка на степента на взаимодействие между участниците [65].

Извличане на *достъпваното уеб съдържание и свързани данни*, събрани или генерирани в резултат на взаимодействието на потребителите с уеб ресурси на един или повече уебсайтове и прилагане на техники за извличане на данни като класификация или размита клъстеризация се използват за анализ на поведението на посетителите на уеб сайтове [77].

Четвърто ниво: *Приложения на данни*. Приложният слой предоставя на потребителите набор от инструменти и приложно-програмни интерфейси (API) за управление, последваща обработка и визуализиране на данни, идващи от предходните слоеве. Потребителите могат да отправят специфични заявки, да изготвят доклади и отчети и да ги представят на отговорните лица с цел взимане на управленчески решения. Освен това този слой осигурява уеб портал и мобилни приложения, които отговарят на нуждите и изискванията на различните заинтересовани страни.

Облачните технологии в съчетание с технологии за управление, съхраняване и разпределена обработка на големи данни като Hadoop/HDFS, Spark, HIVE, осигуряват достъпа до споделени данни и аналитични инструменти и могат да използват огромните компютърни ресурси, които са твърде скъпи за притежание и управление. Новите технологии за работа с големи данни и техническата мощ на съвременните компютри създават реалната възможност за иновативни приложения, базирани на интегриране на разнородни данни.

1. 2. Виртуално образователно пространство в концепцията Интернет на нещата

Интернет на нещата (IoT) е нова революция на Интернет – мрежа, изградена от разпознаваеми и откриваеми обекти, които могат да придобиват знания, като взимат или способстват за взимането на решения, базирани на контекстна информация. „Обектите в пространството на IoT са способни да обменят информация за заобикалящата ги среда, да достъпват информация, агрегирана от други обекти или да участват като компоненти от сложни услуги, изграждайки глобална инфраструктура за информационното общество, даваща възможности за създаването на услуги от по-високо ниво чрез свързването на (физически и виртуални) обекти на базата на съществуващи и развиващи се съвместими комуникационни технологии“ [211]. Но текущата технологична фрагментация и ограниченото внедряване на стандартизирани комуникационни и приложни протоколи, затруднява свързването на съществуващите системи и устройства в интелигентна среда.

Концепцията за Интернет на нещата дефинира устройствата или обектите (нещата), съставляващи екосистема, която обитават и обогатяват чрез споделяне на своите функционалности, съдържание и знания [93]. Електронното обучение може да бъде включено в условията на IoT. Всеки аспект на онлайн образованието е като нещо или като комбинация от неща: учене, оценка, интелигентна образователна помощ и др. Образователните аспекти са капсулирани в единици, които взаимодействат и се свързват в цялата среда. Те споделят знанията си, които могат да бъдат детайлизирани до конкретно потребителско ниво или представени по обобщен начин по отношение на групи потребители, изпълнение на образователни процеси, придобити знания, степен на разбиране на учебния материал. От архитектурна гледна точка, за интегриране на електронното обучение с IoT парадигмата, може да бъде възприета концепцията за архитектура на компонентните услуги (*Service Component Architecture (SCA)*) [59]. Компонентът предоставя функционалност, която няма ограничения относно технологиите, използвани за изпълнение, процеси или машини, на които компонентът ще съществува. Взаимодействията на компонентите могат да се моделират като комуникации, базирани на услуги. По този начин компонентът е софтуерен обект, предназначен да взаимодейства с други компоненти, капсулирайки определена функционалност или колекция от функционалности. Има ясно дефиниран интерфейс и съответства на предписано поведение, което е общо за всички компоненти на една архитектура [201].

Навлизането на Интернет повсеместно в нашия живот и постепенната му трансформация в Интернет на нещата (IoT) е предпоставка за създаване на “кибер-пространство” [48]. В [46] се дефинира „кибер-физическа система“ (*Cyber-Physical-System, CPS*) като инженерна система, изградена от изчислителни и физически компоненти. "Физически" са елементите на системата, заемащи физическо пространство, "кибер" са изчислителните и комуникационни елементи на системата.

Понятието „кибер-физическа система“ се използва за подчертаване тясната интеграция и координация между изчислителни и материални ресурси, където съществува компактна интеграция между изчисление, комуникация и контрол, и средата, в която те са разположени [210]. Тази система не може да бъде разглеждана като затворена самостоятелна система, защото участието на човешкия фактор в нея е активно. Новата кибер-физическа реалност е основополагаща за социалните промени в обществото [81].

Затова отчитането на социалната компонента в CPS я трансформира в кибер-физическа социална система (Cyber-Physical Social System, CPSS) [211].

Приложенията на кибер-физическата социална система са в множество области на човешкия живот – в индустрията, в селското стопанство, в бита, в околната среда, в интелигентното здравеопазване, изграждане на интелигентни домове и градове, пестене на енергия, контрол и защита на критични инфраструктури. Реализацията в сферата на образованието се явява като CPSES (Cyber-Physical Social Educational System) и представя подмножество на CPSS в домейна „образование и обучение“.

Могат да се посочат успешни резултати за интеграция на виртуалната среда с физическия свят. В лабораторията „Център за електронно обучение DeLC (Distributed eLearning Center)“ на Пловдивския университет се разработва кибер-физическа-социална система, която се изгражда като наследник на системата за електронно обучение DeLC [195, 196]. Тя е наречена VES (Virtual eLearning Space) и се реализира чрез агентно-ориентиран подход. Изгражданото образователното пространство [193, 194] се "населява" от активни автономни компоненти, наречени асистенти. Те са имплементирани като интелигентни софтуерни агенти, които поддържат планирането, организирането и изпълнението на учебния процес [162, 200, 201]. Пространство интегрира виртуалния свят на електронното обучение с интелигентни устройства, сензори за събиране на данни от физическия свят, комуникационни инфраструктури и системна архитектура.

Повечето университети в страната реализират проекти за прилагане на информационни технологии и новаторски подходи, базирани на готови системи за управление на обучението от типа Microsoft Class Server (предпочитана система е Moodle). При тези среди липсва тясна интеграция на виртуалната среда с физическия свят, в който протича реалния учебен процес.

В рамките на този дисертационен труд *виртуалното образователно пространство* се дефинира като среда, която интегрира различни информационни технологии и педагогически подходи за доставка на учебни материали и образователни услуги в абстрактен университет (без да реализира интеграция с физическия свят). Обхваща различни образователни инструменти и ресурси, които се използват в дигитална форма, за да се подобри образователния процес. Усилията са в посока разработване на методология за структуриране и моделиране на процесите на обучение, детайлизиране и формализация на процесите.

1. 3. Виртуалното образователно пространство и големите данни

Виртуалното образователно пространство ежедневно събира огромно количество данни, което изисква нов подход към обработката на данни чрез рамка за разпределена обработка и паралелна изчислителна парадигма. С термина "BigData" се определят усилията в организирането, съхранението, обработката и анализа на големи масиви от данни, които са толкова обемни и сложни, че е невъзможна тяхната обработка с традиционните приложения за работа с данни [43]. В статията [146] се предлага модел, описващ процесите свързани със съхранение, обработка и анализ на големи данни, които изискват нов поглед и съвместно прилагане на редица утвърдени технологии.

Характеристиките на големите данни

През 2012 г. *Gartner* [108] дефинира големите данни като "информационни масиви с голям обем, висока скорост и/или голямо разнообразие, които изискват рентабилни, иновативни форми на обработка на информацията, позволяващи по-добра информираност, вземане на решения и автоматизация на процесите". Дефиницията е популярна като трите „V“-та (*volume, velocity, variety*).

В различни литературни източници се определят основни характеристики, които се допълват, означавани като ключови думи с буква V [113]:

Volume (обем) – много големи данни по размер. Big Data не е свързан с конкретно количество данни, но терминът се използва за големи обеми (от порядъка на петабайти и ексабайти данни).

Velocity (скорост) - висока скорост на генериране и събиране на данни и необходимост от анализи на потоци от данни. Например съобщенията в социалните мрежи се разпространяват в целия свят за секунди. Съвременните технологии позволяват да се анализират данните в реално време, без да постъпват в бази данни.

Variety (разнообразие) - данните постъпват в различни форми и от различни източници. С технологията на големи данни е възможно да се анализират и консолидират данни от различни типове, като например съобщения, разговори в социалните мрежи, снимки, данни от сензори, видеозаписи или гласови записи.

Variability (надеждност) – отчита редовност и надеждност на постъпващите данни.

Veracity (достоверност) – проверява данните за истинност, оценява качеството на данните. Често се срещат данни с ниско качество, грешки, пропуски и неточности, които могат да окажат влияние върху точността на анализа и вземането на решения. При големия обем данни, тяхното качество и точност е трудно за проследяване и контрол (съобщения в Twitter, съкращения и грешки в разговорна реч, неточно съдържание и др.).

Value (ценност) – проверява данните от гледна точка на тяхната значимост, важност, полезност, използваемост, добавена стойност за компаниите.

Virtue (добронамереност) – етиката при използването на големите данни, в светлината на различните разпоредби за поверителност, правата на личността, защитата на личните данни и потенциална дискриминация.

Viscosity - измерва приложимост на данните, постъпващи от различни източници.

Virality – отчита бързината, с която данните се разпространяват сред потребителите.

Феноменът *Big Data* е все по-присъстващ, все по-разпространен и все по-важен. С нововъзникващата екосистема с големи данни, следват и новите роли, необходими за подпомагане на нейния растеж. С понятието *Big Data Analytics* се свързва група от технологии и методи за обработка на големите данни. Новите технологии, като облачни изчисления и разпределени системи, заедно със съвременните методи за анализ на данни позволяват едновременно използване на различни типове данни и извеждане на знания.

В масивите от големи данни могат да се крият ценни знания и информация, те са трудни за откриване поради сложността на данните. Науката за данните [74] се явява основно средство за откриване и оползотворяване на потенциала на големите данни като предоставя начини за откриване на взаимоотношения, за осмисляне на разнообразна информация, за достигане до прозрения, подобро разбиране на проблемите и възможности за прогнозиране на бъдещето.

Основните характеристики на големите данни, определят и проблемите при работа с тях. Процесите на решаване на тези проблеми се разглеждат като предизвикателства пред големите данни [7].

Предизвикателства пред големите данните:

- *Интеграция на данни*: източниците на данни обикновено са различни, генерираните данни са от различен тип и са подчинени на различни стандарти. В много случаи постъпващите в системата данни са неподходящи за директна обработка и интеграция в платформа. Това изисква допълнителни стъпки за обработка на данните, премахване на противоречиви данни и преобразуване във формат, съгласно дефинираните правила, което е нетривиална задача и обикновено изисква прилагане на специфичен софтуер [131].

Големите данни могат да бъдат с различна степен на структурираност, като програмния код е оптимизиран за работа с определени външни формати [29]:

- Хранилища за съхранение и обработка на *неструктурирани данни* (новини, документи и др) във формат (PDF, MS Word, Excel и др.) Неструктурирани данни могат да се генерират от: социални мрежи (Facebook, Tweeter, LinkedIn и др.), мобилни телефони, смартфони, таблети и др. Тези данни се различават по състав и организация, както и по съдържание - например текстови документи с общ файлов формат, но различно съдържание (доклади, анализи, заповеди и т.н.). Организацията на неструктурираните данни допуска динамични и често непредсказуеми промени. Например една преписка може да се превърне от прост линеен списък от документи “въпрос/отговор” в сложна структура, поддържаща данни от технически, юридически и друг характер.

- Хранилища за съхранение и обработка на *полуструктурирани данни*. Полуструктурирани са данните, когато се съхраняват във формат XML, JSON, YAML или други формати. Различните типове данни изискват различен подход и софтуер за обработка.

- Пример за *структурирани данни* са таблици, бази от данни, доклади генерирани от бази от данни и др.

- *Изисква се мащабна инфраструктура и скалируемост*, поради обема на големите данни. С растежа на данните е необходимо да се увеличават ресурсите и капацитета на инфраструктурата, за да се осигури способността за обработка на по-големи обеми данни. Инструментите за работа с такива данни използват разпределени системи, така че данните могат да се съхраняват и анализират в разпределени бази от данни и да се обработват паралелно. В резултат на разпределеното съхранение на данните могат да се решават мащабни изчислителни задачи като се обработват големи масиви входно/изходни данни и се извършват голям брой изчисления.

При разпределената архитектура - данните се разпределят и съхраняват на няколко сървъра. В този случай важи теоремата CAP, известна и като теорема на Брюър [106], която гласи, че в една разпределена система трябва да се избера две от следните правила:

- ✓ *Консистентност* (Consistency - C) - всички клиенти на базата от данни виждат една и съща информация, даже при конкурентно обновяване.
- ✓ *Наличност* (Availability - A) - всички клиенти на базата от данни могат да достъпват данните и да извличат информация.
- ✓ *Възможност за разделяне* (Partition tolerance - P) - базата от данни може да се разделя и локализира върху множество сървъри.

Едновременното осигуряване на трите не е възможно. В повечето случаи системите за работа с нерелационни бази от данни избират наличност и възможност за разделяне; гарантирането на консистентност при разпределена база от данни изисква кворум между отделните сървъри, което води до забавяне във времето.

Поради необходимостта от съхранение и обработка на големи данни, които се разширяват хоризонтално се използват хранилища, предоставящи механизъм за съхранение и възстановяване на данни със свободен модел. Различните хранилища използват различни технологии за съхранение на данните [7]:

- *Технология за съхранение на данни: ключ-стойност (Key-Value)*

Това е бързо, мащабируемо и високо достъпно хранилище за произволен достъп (четене/запис) до всякакви данни, асоциирани с ключ. При този подход базата от данни е съвкупност от наредени двойки <ключ, стойност>. В програмните езици тази структура е позната като: *associative array, map, dictionary* и др.

- *Колонни хранилища (Column based)*

Тези хранилища представят данните чрез фамилия от колони. Отново се използва “ключ”, който сочи към фамилия от колони. Но всеки ред може да има различни колони спрямо останалите редове и всеки ред има уникален ключ - идентификатор за реда. Всяка колона от базата от данни се третира отделно и съдържа двойка от типа <име/стойност>, заедно с времева марка (timestamp). Тя показва датата и часа на въвеждане на данните и се използва за определяне на най-новата версия на данните.

- *Графови хранилища (Graph stores)*

Графовите хранилища работят с графови структури, в които се записва семантика на данните; връзки и отношения между обекти; с цел търсене и вземане на решения и др. Данните се представят чрез обекти, наричани възли (nodes), които могат да имат свойства (properties) и връзки (edges) с други възли. Всички възли са свързани помежду си, а

връзките имат собствена посока и значение. Най-подходящи са при представяне на свързани данни и анализи в реално време, при които размерът на данните нараства бързо. Те са отличен избор при приложения за управление на достъпа, социални мрежи и други. Примерен продукт: Neo4j с отворен код, написан на Java, използва език Neo4j CQL (Cypher Query Language) [106]. Пример за употреба във Facebook – всеки профил е свързан с приятели, съобщения, харесвания на статуси, постове и много други и разнородни данни. Системи от този вид се използват за доставка на уеб съдържание, медия, поддръжка на платформи за игри и др.

- *Документни хранилища (Document stores)*

Документното хранилище служи за работа с документи в структуриран или неструктуриран формат. Хранилището е от тип *Key-Value*, като стойността е документ. Документът може да е представен отново във вида *Key-Value*. Примерен продукт от този тип е MongoDB, съхранява структурираната информация в JSON формат.

- Големите данни се генерират и събират с изключително висока скорост. Това налага *необходимостта от бърза обработка на данните*.

Map/Reduce е концепция за извършване на разпределени изчисления на големи обеми от данни. Първоначално е създадена от Google през 2004 г. [71] за индексване на уеб страници. Днес програмите, използващи този стил за автоматична паралелизация се изпълняват на клъстери от компютри, като един клъстер може да се състои от хиляди машини. Изпълнителната система осигурява разделянето на данните, управлението на изпълнението върху множество машини, управление на комуникацията на машините и обработка на откази.

Прилагайки концепцията MapReduce, работа се състои от стъпките: *Map* и *Reduce*. Основната идея е входните файлове да се разделят на M части. Тези части се разпределят автоматично от управляваща програмата между една или повече машини (или други части на програмата). Те получават входните данни и задачи за предварителна обработки. Резултатите от работата на *Map* са двойки от вида *ключ - стойност*, които се записват в междинни файлове, като данните се подреждат в групи. Получените групи се разделят на R части и се подават на *Reduce* процедури. Специална разделяща функция определя кои ключове към коя R-група да попаднат. Функцията *Reduce* обединява междинните резултати, свързани с един и същ ключ. Основният възел получава резултатите, получени от работните възли и генерира решаването на проблема.

Съществува голямо разнообразие от софтуерни реализации на MapReduce концепцията. Една от най-разпространените програмни реализации на MapReduce е Apache Hadoop [182]. Това е софтуерна платформа с отворен код, написана на Java. Поддържа множество алгоритми за съхранение и разпределена обработка на големи масиви от данни на компютърни клъстери.

В статията [147] се предлага модел, свързан с процесите на обработка и анализ на големи данни, основавайки се на *Map/Reduce* концепцията. Създаденият модел дава възможност за проследяване на процесите, свързани с разпределени изчисления при използване на различни алгоритми и инструменти за реализацията им.

- Изследователски и инженерни предизвикателства са въпросите по:

- *осигуряване на качеството* при извличането и обработката на големи данни. Често се срещат данни с грешки, пропуски и неточности. Те могат да окажат влияние върху точността на анализа и вземането на решения.

- *аспекти на сигурността* при обработка на големи данни и анализите им, устойчивостта и запазване на данните. Защита на данните от неоторизиран достъп и злоупотреби е критична задача.

- *правни и етични предизвикателства*: събирането и използването на големи данни може да повдигне въпроси относно правата на личността, защитата на личните данни и потенциалната дискриминация.

- *обработката на големи данни изисква специализирани умения и познания*, които не са толкова широко разпространени. Намирането на подходящи специалисти може да бъде предизвикателство.

Новите технологии, като облачни изчисления и разпределени системи, заедно със съвременните методи за анализ на данни помагат за преодоляване на предизвикателствата пред Big Data. Отворени проблеми в направлението са свързани с: търсене на нови техники за анализ на данните; оптимизиране на достъпа до големи данни; прилагане на агенти за извличане на знания от данните и др.

Големите данни играят важна роля и в сферата на образованието и средите за обучение. Използването на тези данни може да подобри образователния процес, предоставяйки информация и възможности за оптимизация на учебните програми и методите за обучение.

1. 4. Виртуалното образователно пространство на нововъзникващите технологии и рисковете

Концепцията „Индустрия 4.0“, наричана в книгата на Клаус Шваб „Четвърта индустриална революция“ [186] е нова парадигма, която предвижда цифровизация на всички активи и икономически агенти, включени в единната веригата на стойността и тяхната интеграция в обща цифрова екосистема. Предпоставки за нейното изграждане са трансформирането на водещите технологии, които днес е прието да се наричат нововъзникващи технологии (*emerging technologies*) или подривни технологии (*disruptive technologies*) [54]. Те се характеризират с радикална новост, бърз растеж и въздействие върху останалите технологии. В бизнес теорията, подривна е технология, която формира нова пазарна ниша и верига на стойността, трансформира или прекъсва съществуващите и измества утвърдени водещи на пазара фирми, продукти и обединения. Терминът е дефиниран от Боуър и Кристенсен в [54] и се смята за една от най-влиятелните бизнес идеи от началото на 21-ви век. В американски източници тези технологии се определят и като *брилянтни технологии* [56], които се използват и се развиват във времето и имат потенциал да повишат влиянието си върху процесите на производство на знания.

Четвъртата индустриална революция е етап в развитието на обществото, характерен с бурното развитие и напредъка на цифровите технологии и тяхното масивно навлизане във всички сфери на живота. Основава се на интелигентни и свързани помежду си машини и системи. Появяват се: Интернет на нещата (Internet of Things, IoT), Интернет на събитията, Интернет на хората (социалните мрежи - Facebook, Tweeter, LinkedIN и др.), Интернет на локацията (телефони, планшети и др.). Създават се такива мрежи, чрез които да се преустроят икономическите и обществените процеси така, че да се изключи необходимостта от участие на човека в част от действията и операциите.

Интегрирането на физическия и виртуалния свят, както и на социални общности, коренно различава четвъртата индустриална революция от останалите. Създава се потенциал за увеличение на приходите и подобряване качеството на живот. Очаква се животът на хората да се облекчава със създаване на интелигентна медицина, умни градове, умните домове, интелигентни околна среда и селско стопанство [16, 18].

В доклад на Световния икономически форум от септември 2015 г. са идентифицирани 21 конкретни технологични промени, наричани също „повратни точки“ [186], които очаквано биха формирали нашия бъдещ цифров и хиперсвързан свят. Идентифицираните повратни точки са следните: Имплантируеми технологии, Нашето цифрово присъствие, Зрението като новия интерфейс, Компютърно присъствие навсякъде, Суперкомпютър в джоба, Съхранение за всички, Интернет на нещата, Свързаният дом, Умните градове, Големи данни за решения, Автомобили без шофьор, Изкуствен интелект и вземане на решения, Изкуствен интелект и административните професии, Роботика и услуги, Биткойн и блоковата верига, Икономиката на споделянето, Правителствата и блоковите вериги, 3D печат и производство, 3D печат и човешкото здраве, 3D печат и потребителските продукти, Невротехнологии [186].

Безспорно, **изкуственият интелект е „сърцето“ на нововъзникващите технологии**, защото свързаните с него научни пробиви формират направления, чието

функциониране зависи в най-голяма степен от представянето на знанията и имитиране на способностите на човешките разсъждения.

Цифровите технологии стават все по-сложни и интегрирани и като такива те предизвикват значителни трансформации в обществото и икономиката.

Някои от очакваните *положителни въздействия* от тази промяна са [170]:

- Възможности за обработка на големи обеми от данни, изпълнение на сложни алгоритми върху данните, създаване на модели и предвиждане на събития и явления, значително по-рационални и обективни решения, основаващи се на голям обем фактология и знания, а не на субективни усещания и възприятия.

- В полза за гражданите - възможност за по-добро здравеопазване, безопасни коли, персонализирани продукти. Улеснява се достъпа до информация и обучение, работните места стават по-безопасни.

- В полза на бизнеса - нови продукти и услуги, зелена икономика, прецизно селско стопанство, здравеопазване, мода, туризъм. Възможности за увеличаване на производството, подобряване на качеството и обслужването на клиенти, разкриване на нови работни места и иновативни професии.

- Нови възможности за администриране и значително намаляване на бюрокрацията.

- Енергийна независимост, спестяване на енергия, значителен напредък в много области и производства.

За съжаление, трябва да отчитаме и *отрицателните въздействия* на тази промяна, като например: загуба на работни места в традиционни професии; киберпрестъпност и хакерски атаки; алгоритмични грешки и неразбираемост; необходимост от нови концепции за отговорност, отчетност и управление; повишаване на неравенството между хората. За субектите на пазара на труда организацията на задачата за адаптиране към цифровата среда се усложнява от развиващия се процес на автоматизиране на неквалифицираните и нискоквалифицираните работни места, чрез работи и алгоритми, които обработват големи количества данни. Технологиите, които ускоряват и разширяват обработката на информацията могат и да съкратят драстично работните места. Социалните мрежи отварят вратите и минимизират преградите в свързването и общуването, но пораждаат и нови форми на социално безпокойство. Цифровизацията улеснява разплащанията и търговията, но и стимулира нови видове измами [12].

Изучаването на риска, управлението на риска или, както често се пише, **риск мениджмънта** е в предмета на различни науки, в конструкциите на много теории, в огромното разнообразие от методи, модели, алгоритми, програми, устройства и системи на техниката, технологиите и човека.

„**Рискът** е събитие или група случайни събития, нанасящи ущърб на обект. Под обект се разбира материален обект или имуществен интерес или някакво свойство на обект, а ущърб (щета) е влошаване или загуба на свойство на обекта“ [2]. Той често се приема като несигурност и опасност, съмнение по отношение на възможността да предвижда кои възможни изходи ще се сбъднат. Несигурността е личното усещане за риска. **Управление на риска** представлява координирани дейности за ръководене и насочване на дадена организация по отношение на риска [2, Guide 73]. **Система за**

управление е рамка от политики, процедури, указания и свързаните с тях ресурси за постигане на целите на организацията. Идентифицираните до момента основни рискови фактори, които потенциално засягат отделните нововъзникващи технологии са анализирани в [13]. Формулирани са осем категории рискови фактори:

(1) **Поверителност и защита на данните** - фактор, който включва рискове от злоупотреба с поверителност, потенциален надзор, наблюдение, on-line тормоз, преследване, наблюдение, събиране, защита и сигурност на лични данни, хакерство, заплаха за сигурност, престъпност, повишена уязвимост към кибер-атаки, кибер-риск, възможност за краткосрочна злоупотреба с доверие и кражба на самоличност. Приложения, интегрирани в тялото на човека могат да станат заплаха, попадайки под нежелан чужд контрол и липса на регулации.

(2) **Промяна на работни места/промяна в пазара на труда** - фактор, който включва рискове от загуба на работни места, намалена устойчивост след загуба на работа (поради по-малко спестявания), повече труд, основан на договор/задача (в сравнение с обикновено по-стабилната дългосрочна заетост), глобална и регионална верига за доставки и логистика: по-ниско търсене, което води до загуба на работни места, автоматизация на работата, непрекъснатата заетост в сферата на услугите (24-часови услуги).

(3) **Рискове свързани с психично разсейване/нарушения във вниманието** - включват психично разсейване, причиняващо злополуки, травми от негативни преживявания, повишена зависимост и бягство от реалността, повишено разсейване (дефицит на внимание), пристрастяване.

(4) **Манипулация и ехо камера** – потенциално рисковете могат да са предизвикани от разпространяване на неточна информация (необходимост от управление на репутацията), ехо камери, липса на прозрачност, при която потребителите не са запознати с информационните алгоритми (за новини/информация), по-висока сложност и загуба на контрол, манипулации с доверие, нечестни алгоритми, неразбиране. Може да доведе до поляризация на обществото, до манипулиране на избори и други събития.

(5) **Фрагментация** – възможно е да се прояви поради стимулиране на групово мислене в рамките на групи по интереси и повишена поляризация, неравенство, лобиране, правно регламентирани структури за управление, битки за алгоритми, наличие и толериране на ограничени и ограничителни за достъп среди, региони или страни. В някои случаи автоматизираните системи: при отпускане на кредити, при разглеждане на кандидатури за работа, при лечение, при правни въпроси могат да доведат до засилване и утвърждаване на социални разделения.

(6) **Отговорност и отчетност** - рисков фактор, който се основава на неформална, формална, регламентирана или нерегламентирана собственост върху алгоритмите и техните приложения, отговорността, доверителните права, основанията и обосноваване на законността или промените в законодателството, финансовото оповестяване, риска, намалената възможност за оценка на потенциално сивата икономика. Предизвикателство е

определянето чия е отговорността за щети. Твърде стриктните регулации ограничават тласъка към иновации.



Фигура 4. Технологии, изложени на най-голям брой рискове [11].

(7) **Екология, екосистема и етика** - фактори, които имат всестранно влияние и провокират рискове, свързани с нарастване обемите и необходимостта за утилизация на отпадъците, допълнителна тежест върху околната среда, въздействието на 3D печат технологиите за „печатани храни“ върху икономиката и екологията на селското стопанство, манипулиращи фактори за здравето, екзистенциална заплаха за човечеството, отваряне на възможности за „отпечатване“ на обекти с високи нива на потенциална злоупотреба (например оръжия), качество на марките и продуктите, значителни нарушения в контрола на производството, потребителските разпоредби, търговските бариери, патенти, данъци и други правителствени ограничения, борба за адаптиране, етични дебати, произтичащи от „печатане“ на органи, отделни части или цели тела и контрол върху тяхното производство, отговорност за гаранциите на качеството, неконтролирано или нерегулирано производство на части от човешкото тяло, медицинско оборудване или храна, производството на анизотропни части, които могат да ограничат функционалността, опасност от срив (изчерпване) на енергийната система.

(8) **Промяна в структурата на приходи/разходи и собствеността на активите** - фактори засягащи цялата икономическа и социална система и преразпределителните

механизми, свързани с проява на риск от не конкурентно и не пазарно превъзходство от отношение на интелектуална собственост като източник на стойност в производството, по-малко инвестиционен капитал в системата, намаляване на приходите от трафик нарушения и възстановяване на застраховка и пътна помощ („плащайте повече за шофиране сами“), изменения в структурата на собственост на МПС, лобиране срещу автоматизацията и др.

Изложените на най-голям брой рискове (фигура 4) са технологиите на „Изкуствен интелект и вземане на решения“, „Големи данни и решения“, „Икономика на споделянето“, „3D печат и производство, потребителски продукти и здравеопазване“, „Нашето цифрово присъствие“ и „Автономни автомобили“ [170].

Безспорно изкуственият интелект, блокчейн технологията, интернет на нещата, 3D принтирането и редица други технологии оказват значително влияние в съвременния човешки живот и до голяма степен ще определят бъдещето. Ключовите технологии, които са движещи сили на Четвъртата индустриална революция са многобройни. Те обаче имат обща доминираща характеристика – използват цифровата и всепроникващата мощ на информационните и комуникационните технологии. Нарастващата сложност и взаимозависимост на технологиите се придружава от значително увеличаване на рисковите фактори, съчетани с не винаги адекватни средства за осигуряване на информационна сигурност. Всеки риск има негативни, а често и неизвестни, недефинирани предварително въздействия. Това изисква изучаване и вземане на решение за мониторинг, контрол и оценка на риска в условията на мултикритериално взимане на решения, с акцент върху модели с размита логика, алгоритми и приложения. За тази цел може да се предложи следната схема от пет фази [11]:

Фаза 1: идентификация на риска;

Фаза 2: количествена и качествена оценка на риска и неговите характеристики;

Фаза 3: избор на инструмент и/или инструменти за въздействие върху риска (стандарты, норми, правила, модели, методи, алгоритми);

Фаза 4: риск мениджмънт, непосредствено въздействие върху средата или обекта с избрания инструментариум;

Фаза 5: мониторинг, контрол и оценка на риск мениджмънта, което може да е достатъчно основание за връщане отново в предходни фази.

Възможни са и други комбинации от рискове за проучването, с акцент върху различни фактори като: психологически, политически, социално-икономически, демографски, технологични аспекти. Целта на всеки риск мениджмънт е създаване, защита и нарастване на богатството и ценностите чрез управление на факторите на неопределеност, които могат да влияят както отрицателно, така и положително върху достигането на целите на предприятието.

Стандартите за управление на риска са основата за мениджърите и за практикуващите, и те се прилагат и интегрират във вътрешните документи и стратегии в предприятията. В различни страни стандартите се приемат или по начина, по който международната организация по стандартизация ги е формулирала, или те са модифицирани в съответствие с националното законодателство, правила и практика.

Времето и технологиите притискат и по думите на Клаус Шваб три са най-важните и неотложни стъпки, които чакат адекватни подходи и решения [186]:

- *първо*: повишаване на осведомеността и насърчаване на познанията във всички обществени групи;

- *второ*: създаване на позитивни, общоприети и всеобхватни послания относно възможностите за възползване от четвъртата индустриална революция от сегашните и следващите поколения;

- *трето*, на база на повишената информираност и споделени послания стартиране на процес по реструктуриране на икономическите, социалните и политическите системи, за да могат те да се възползват пълноценно от отворените възможности на технологичния пробив.

В книгата „*Индустриите на бъдещето*“ Алек Рос [15] обобщава основните рисове и опасности: силите, които „отприщват безпрецедентния напредък“ в начина на живот и натрупването на богатство имат и възможността да допускат кражба на личността или намеса в личното жизнено пространство. Технологиите, които ускоряват и разширяват обработката на информацията могат и да съкратят драстично работните места. Социалните мрежи отварят вратите и минимизират преградите в свързването и общуването, но пораждат и нови форми на социално безпокойство.

Четвъртата индустриална революция е тясно свързана със следващото поколение Интернет, което трябва да гарантира, че огромният потенциал на изкуствения интелект, виртуалната реалност, връзката с физическия свят, машинното обучение, повсеместните мрежи от хора и машини се използват пълноценно за всеобщо подобряване качеството на живот и допринасят за изграждането на устойчиви общества.

За да продължи да се развива концепцията на виртуалното образователно пространство в тази динамична и хетерогенна среда, тя трябва да може да се адаптира към всички характеристики и изисквания, които средата налага. Но всички промени трябва да са подчинени на политика и инвестиции за надежден изкуствен интелект [172] и да са основани на етичен и ориентиран към човека подход [80].

1. 5. Модели за анализ на предметната област

В литературата [20] формалните модели за анализ на програмно осигуряване могат да се разделят на: модели, основани на свойства, изпълними модели и модели интегриращи двата подхода.

1.5.1. Модели, основани на свойства (property-based models)

Тези модели от своя страна се разделят на логически и алгебрични модели.

1.5.1.1. Средства за изграждане на логически модели са:

- Съждително смятане (propositional calculus) [4];
- Предикатно смятане (predicate calculus) [4];
- Предикатно смятане от по-висок ред (higher-order predicate calculus) [4]
- Ламбда смятане (lambda calculus) [1];
- Ламбда смятане от по-висок ред (higher-order lambda calculus) [1];
- Модални логики (modal logics) [21];
- Темпорални логики (temporal logics) [207];
- Линейни темпорални логики (linear temporal logic, LTL) [129];
- μ -смятане (или смятане с неподвижни точки, μ -calculus) [57];
- Логики с явно време (timed temporal logics) [207].

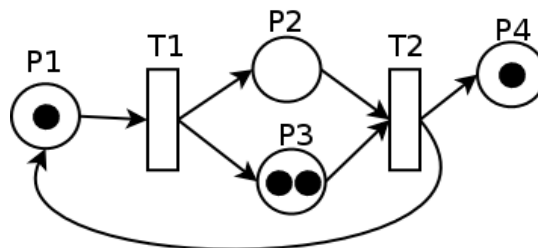
1.5.1.2. Средства за изграждане на алгебрични модели са:

- Релационни алгебри, лежащи в основата на релационните системи за управление на бази от данни [66];
- Алгебрични модели на абстрактните типове данни [97];
- Алгебри на процесите (process algebras, process calculus) [82].

1.5.2. Изпълними модели (executable models)

Основни средства за изграждане на изпълними модели са:

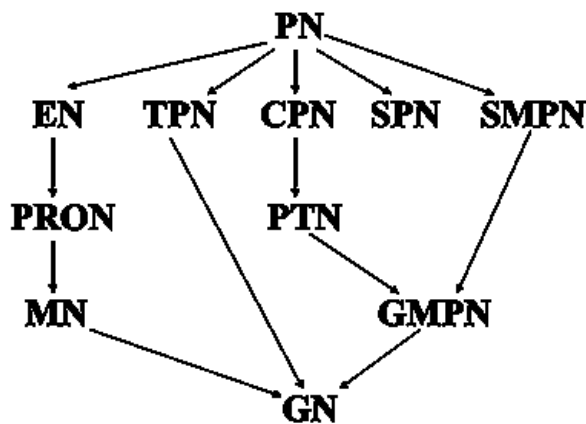
- Крайни автомати (finite state machine);
- Системи за преход (labeled transition systems);
- Взаимодействащи автомати (communicating finite state machines);
- Йерархични автомати (hierarchical state machines);
- Времеви автомати (timed automata);
- Хибридни автомати (hybrid automata);
- Абстрактни автомати (abstract state machines);
- Мрежи на Петри (PN - Petri nets) - просто и ясно средство за моделиране [161]. Те имат много разширения, добавящи редица нови свойства и възможности за моделиране. Независимо от техните спецификации, всички тези инструменти имат позиции (обозначени с P_i) и преходи (обозначени с T_i) с дъги и ядра, които се движат в мрежата.



Фигура 5. Пример на мрежа на Петри.

Основни разширения на мрежите на Петри са [31, 32]:

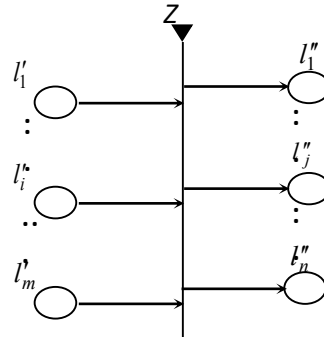
- o EN – *Evaluation nets* добавят продължителност на движението на ядрото;
- o TPN – *Temporal PN* добавят момент за активиране на прехода;
- o CPN – *Color PN*. При цветните мрежи на Петри всяко ядро има цвят и може да се движи само по дъги от същия цвят;
- o SPN – *Stochastic PN*. Изборът на дъга при стохастичните мрежи на Петри се основава на произволно генерирано число;
- o SMPN – *Self Modifying PN*. При самомодифициращите се мрежи на Петри, изборът на дъга се основава на генерирано двоично число 0 или 1, като ако се генерира 0, то преходът на ядрото не се извършва;
- o PRON – *Pro-Net* въвежда тип на прехода;
- o PTN – *Predicate/Transition Net*. Предикатно-преходните мрежи дефинират условие на прехода;
- o MN – *M Net*, при които ядрата изпълняват и допълнителни процедури;
- o GMPN – *Generalized Modifying PN* – ядрото се поглъща, когато предикатът генерира 0, а в противен случай при генериране на 1 ядрото извършва прехода.
- o GN *Generalized Net* – обединяват различните разширения в единен формализъм за описание на паралелно протичащи процеси. Различните разширения на мрежите на Петри и връзката между тях се представя на фигура 6.



Фигура 6. Разширения на мрежите на Петри

1.5.3. Моделиране чрез апарата на обобщените мрежи

Концепцията на Обобщените мрежи (ОМ), представена в [31, 32] се състои от преходи. Графично преходът се представя като съвкупност от два елемента: \circ и \Uparrow (Фиг. 7).



Фигура 7. Графично представяне на прехода Z в ОМ

Всеки преход в ОМ има поне една входна и поне една изходна позиция. Графичното означение на позициите е кръгче \circ . В позицията може да влиза не повече от една дъга и също от позицията може да излиза не повече от една дъга. Позиция, от която излиза дъга е входна за прехода, а позиция, в която влиза дъга е изходна за прехода.

Всеки преход може да има m входни и n изходни позиции, където $m \geq 1$ и $n \geq 1$. На фиг.7 входните позиции за прехода Z са означени с $l'_1, \dots, l'_i, \dots, l'_m$, а изходните с $l''_1, \dots, l''_j, \dots, l''_n$. Входна позиция, в която не влиза дъга се нарича вход на мрежата, а изходна позиция, от която не излиза дъга – изход на мрежата. В позициите в мрежата може да има ядра, които се предвижват от входните към изходните позиции на преходите. Когато във входните позиции има достатъчен брой ядра и настъпи определения за прехода момент от време, ядрата от входните позиции придобиват възможност да се придвижат до изходните позиции. Този процес на преминаване на ядрата от входните в изходните позиции на прехода се нарича активиране на прехода. В началото ядрата, които постъпват в мрежата през входните ѝ позиции имат т.нар. начални характеристики. При всяко преминаване през преход в мрежата те получават нови характеристики и така всяко ядро в мрежата е уникално и има своя история. Всяка позиция има свой капацитет.

Преход се означава със символа \Uparrow . С всеки преход е свързана индексирана матрица, която се нарича условие на прехода. Тя съдържа толкова реда, колкото са входните позиции на съответния преход и толкова колони, колкото са изходните му позиции. За прехода Z от Фиг. 7 индексираната матрица е с размерност $m \times n$. Елементите ѝ са предикати, които имат логически стойности (“вярно” или “невярно”). Така всяка входна и изходна позиция са свързани с предикат. Ако предикатът има вярна стойност (“*true*”), ядрата, които се намират в съответната входна за прехода позиция могат да се придвижат до съответната му изходна. Ако предикатът има невярна стойност (“*false*”), ядрата остават във входната позиция.

1.5.3.1. Формално описание на преход

Преходът Z в ОМ (Фиг. 7) се представя като наредена последователност от седем компоненти:

$$Z = \langle L', L'', t_1, t_2, r, M, \square \rangle,$$

където:

- 1) $L' = \{l'_1, \dots, l'_i, \dots, l'_m\}$ – крайно непразно множество от входните позиции на прехода;
- 2) $L'' = \{l''_1, \dots, l''_j, \dots, l''_n\}$ – крайно непразно множество от изходните позиции на прехода.
- 3) t_1 – момент време на активиране на прехода.
- 4) t_2 – задава продължителността на активното състояние на прехода.
- 5) r – условие на прехода.

Представява индексирана матрица от вида:

| | l''_1 | ... | l''_j | ... | l''_n | |
|--------|-----------|-----|-----------|-----|-----------|---------------------------------|
| l'_1 | $r_{1,1}$ | ... | $r_{1,j}$ | ... | $r_{1,n}$ | за $i \in [1, m], j \in [1, n]$ |
| ... | ... | ... | ... | ... | ... | |
| . | $r_{i,1}$ | ... | $r_{i,j}$ | ... | $r_{i,n}$ | |
| r | ... | ... | ... | ... | ... | |
| = | $r_{m,1}$ | ... | $r_{m,j}$ | ... | $r_{m,n}$ | |
| l'_i | | | | | | |
| . | | | | | | |
| . | | | | | | |
| . | | | | | | |
| l'_m | | | | | | |

$r_{i,j}$ е предикат, съответстващ на $i^{\text{та}}$ входна позиция на прехода и $j^{\text{та}}$ изходна позиция на прехода. Ако предикатът е верен (има стойност “true”) е възможно преминаване на ядро от $i^{\text{та}}$ към $j^{\text{та}}$ позиция. Предикатите не могат да зависят от бъдещи събития.

- 6) M – задава капацитетите на дъгите.

Представява индексирана матрица от вида:

| | | | | | | | |
|-------|--------|-----------|--------|-----------|--------|-----------|--------------------------|
| | l'_1 | ... | l'_j | ... | l'_n | | |
| $M =$ | l'_1 | $m_{1,1}$ | ... | $m_{1,j}$ | ... | $m_{1,n}$ | |
| | | ... | ... | ... | ... | ... | |
| | | $m_{i,1}$ | ... | $m_{i,j}$ | ... | $m_{i,n}$ | за $i \in [1, m], j \in$ |
| | | ... | ... | ... | ... | ... | $[1, n]$ |
| | | $m_{m,1}$ | ... | $m_{m,j}$ | ... | $m_{m,n}$ | |
| | l'_i | | | | | | |
| | . | | | | | | |
| | . | | | | | | |
| | . | | | | | | |
| | l'_m | | | | | | |

$m_{i,j}$ са естествени числа, задаващи капацитета на дъгата от $i^{\text{та}}$ входна позиция на прехода към $j^{\text{та}}$ изходна позиция на прехода.

7) \square – булев израз. Ако стойността е “true” съответният преход може да се активира, ако е “false” – не. В него участват идентификаторите на всички входни позиции на прехода, свързани с логическите операции и (\wedge) и или (\vee).

Ако булевият израз е от типа: $\wedge(l_{i_1}, l_{i_2}, \dots, l_{i_u})$, то това означава, че във всяка входна позиция $l_{i_1}, l_{i_2}, \dots, l_{i_u}$ трябва да има най-малко по едно ядро.

Ако булевият израз е от типа: $\vee(l_{i_1}, l_{i_2}, \dots, l_{i_u})$, то това означава, че най-малко в една от входните позиции $l_{i_1}, l_{i_2}, \dots, l_{i_u}$ трябва да има най-малко едно ядро.

След като е дефинирано понятието преход като основен градивен елемент на обобщената мрежа, може да се дефинира и понятието ОМ.

1.5.3.2. Формално описание на обобщена мрежа

Наредената четворка:

$$E = \langle\langle A, \pi_A, \pi_L, c, f, \theta_1, \theta_2 \rangle, \langle K, \pi_K, \theta_K \rangle, \langle T, t^\circ, t^* \rangle, \langle X, \Phi, b \rangle\rangle,$$

се нарича **обобщена мрежа (ОМ)**, ако:

- 1) A – множеството от всички преходи в мрежата;
- 2) π_A – функция, задаваща приоритетите на преходите - $\pi_A: A \rightarrow N$, за $N = \{0, 1, 2, \dots\} \cup \{\infty\}$
- 3) π_L – функция, задаваща приоритетите на позициите, като L е множеството от всички входни позиции в ОМ.

 $\pi_L: L \rightarrow N$, където $L = \text{pr}_1 A \cup \text{pr}_2 A$ и с $\text{pr}_k X$ е означена $i^{\text{та}}$ проекция на n -мерното множество X , $n \in N$, $n \geq 1$, $1 \leq k \leq n$.
- 4) c – функция, задаваща капацитетите на позициите, $c: L \rightarrow N$.

- 5) f – функция, определяща вярностната стойност на предикатите.
- 6) θ_1 – задава следващия момент, в който може да се активира прехода. Стойността на тази функция се преизчислява в момента, в който завършва активното състояние на прехода.
 $\theta_1(t) = t'$, където $\text{pr}_3 Z = t$, $t' \in [T, T + t^*]$ и $t' \geq t$.
- 7) θ_2 – функция, която задава продължителността на активното състояние на даден преход Z и стойността ѝ се изчислява в момента, в който се активира прехода.
 $\theta_2(t) = t'$, където $\text{pr}_4 Z = t \in [T, T + t^*]$ и $t' \geq 0$.
- 8) K – множество на ядрата в ОМ
- 9) π_K – функция, която задава приоритетите на ядрата т.е. $\pi_K: K \rightarrow N$
- 10) θ_K – функция, която задава момента от време, в който определено ядро може да влезе в ОМ, т.е. $\theta_K(\alpha) = t$, където $\alpha \in K$, $t \in [T, T + t^*]$
- 11) T – начален момент от време, в който ОМ започва функционирането си. Моментът T се определя по фиксирана времева скала.
- 12) t° – времева стъпка на фиксирана времева скала.
- 13) t^* - продължителност на функционирането на ОМ.
- 14) X – Множество на началните характеристики, с които ядрата влизат в мрежата.
- 15) Φ – характеристична функция. Тя определя новата характеристика на ядрото при преместването му от входната позиция на даден преход в изходната.
- 16) $b: K \rightarrow N$ – функция, задаваща максималния брой характеристики, които едно ядро може да получи по време на движението си в ОМ. Началните характеристики на ядрата, които постъпват в ОМ винаги се пазят.

В описанието на дадена обобщена мрежа може да не се съдържат всичките и компоненти, в този случай мрежата се нарича редуцирана.

В четвърта глава на дисертационния труд са представени формални модели на процеси и компоненти на виртуалното образователно пространство чрез апарата на обобщените мрежи. Те представляват обобщение на създадена серия от модели, свързани с наблюдение и анализ на дейностите на обучаемите в образователна среда, представени в публикациите [99, 100, 143, 146, 147, 152, 153, 154, 155, 156, 158, 202, 204].

ГЛАВА 2. СРЕДСТВА НА ВИРТУАЛНОТО ОБРАЗОВАТЕЛНО ПРОСТРАНСТВО

Виртуалното образователно пространство се разглежда като среда, която интегрира различни информационни технологии, педагогически подходи, образователни инструменти и ресурси, които се използват в дигитална форма, за доставка на учебни материали и образователни услуги в абстрактен университет.

Науката за данните може да даде множество ползи в управлението на университета и да помогне за подобряване на ефективността и вземането на решения в образователната институция. Някои конкретни области, в които науката може да се приложи и да бъде особено полезна в управлението:

Управление на данни за успеваемост на обучаемите и прогнозиране. Науката за данните може да помогне за събирането, анализа и визуализацията на данни за успеваемостта и активността на обучаемите: данни за оценки, резултати от тестове, присъствие и други показатели, за да се определят тенденции, да се идентифицират слаби места и да се предложи подходяща подкрепа и ресурси за тяхното подобрене.

Оптимизиране на програмите и учебните планове. Науката за данните може да помогне за анализ на данни за индивидуалните потребности, предпочитанията на обучаемите, успеха им по различни дисциплини, предишни успехи и избор на учебни пътища. Това може да доведе до по-добро съответствие между програмите и интересите на обучаемите, както и до по-ефективно използване на ресурсите на образователната институция.

Управление на учебните материали. Образователните институции могат да анализират ефективността на различни учебни материали и обучаващи методи. Това позволява да се откриват добри и слаби обучаващи ресурси, с цел подобряване на образователния процес.

Управление на преподавателския състав. Анализът на данни за преподавателите, включително данни за професионалното развитие, резултати от отдела по качеството и други показатели на ефективност може да допринесе за по-добро разпределение на ресурсите на персонала, подобрене на професионалното развитие и оценяване на преподавателите.

Управление на материални ресурси и финанси. Науката за данните може да бъде полезна при оптимизиране на използването на ресурси, (например формиране на групи, графици на преподавателите и др.), с анализ на данни за бюджети, разходи, приходи и други финансови показатели, за да се оптимизират бюджетните процеси и да се вземат по-добри финансови решения.

Управление на комуникацията с обучаемите. Анализът на данни за комуникацията с обучаемите, включително обратна връзка, проблеми и нужди може да допринесе за по-добро управление на отношенията със заинтересованите страни, подобрене на комуникацията и предоставяне на персонализирана подкрепа.

Важно е да се имат предвид конкретните нужди и цели на образователната институция и да се изберат подходящи инструменти и методики за анализ на данните.

Съвременното виртуално образователно пространство изисква възможности за адаптация и персонализация на предлаганите услуги за различни групи потребители и при

различни условия, независимо от времето и мястото. Това предполага интеграция на подходи и инструменти, които осигуряват възможности за съхранение и достъп до данните на различни нива. От друга страна, създава възможности да се интегрират средствата, които обучаемите и обучаващите използват в ежедневно си виртуално пространство.

Пространството се разглежда като активна система, отворена за компоненти с общ подход за достъп и използване на информацията за анализ и вземане на решения [3]. Една съвременна среда за обучение трябва да осигурява базови компоненти като:

- Средства за събиране, пречистване и ефективно съхранение на данните. Съхраняваната информация да бъде достъпна на различни нива на достъп и от различни приложения за извличане, показване и вземане на решения, както и контрол на достъпа до данните. Възможности за свързване на информацията, получена от различни източници.

- Средства за управление на знания с възможности за интелигентно търсене и четене, контрол за цялостност и коректност на данните, автоматизирано извличане на информация и др. Тук се включат инструменти за управление на качеството на данните, автоматизирано откриване на модели в данните и възможности за самообновяване на данните.

- Оптимизиране на процеса на изпълнение на заявки за търсене и достъп до информацията. Средства за преобразуване и визуализиране на информацията, за по-лесно възприемане и използване от потребителя.

- Възможности за интегриране на компоненти с разнородни технологии и педагогически методи, посредством възприемане на общи архитектурни подходи за системата.

Интегрирането на средства в средите за обучение може значително да подобри образователния процес и да предостави ценна информация на обучаемите, преподавателите и администраторите, но от своя страна изисква ефективно управление на образователните данни. Това включва съхранение, обработка и защита на данните, като се осигурява тяхната сигурност и поверителност. Те трябва да бъдат използвани съобразно етични принципи и да зачитат правата на личността и поверителността.

В тази глава от дисертационния труд са обобщени резултати от проведени изследвания и анализи на основни характеристики на виртуалното образователно пространство на Университет, представени в публикациите, свързани с темата на дисертационния труд [145, 165, 168, 197, 200, 201]. Обосновани са предимства на стандартизираното електронно учебно съдържание при доставка на образователни услуги. Мотивирана е необходимостта от прилагане на средства за извличане на знания от данните в обучаващите пространства и търсене на решения за персонализиране на електронното и дистанционно обучение.

2.1. Интеграция на средства за извличане на знания в образователното пространство

Интеграция на техники за извличане на знания от данните в образователното пространство, може да осигури редица възможности за анализи, например: как обучаемите реагират на курсовете, оценяване ефективността на метода на преподаване, кои стратегии за учене работят добре и кои не са полезни за постигане на целите на курса и др. [109].

Съществуват различни подходи за изследване и анализиране на разнородни и големи обеми от данни. Най-често използваните техники за анализ на данни са [208]:

- *Извличане на асоциативни правила* - тази техника открива връзки между огромните набори от данни.

- *Класификация* - техника за идентифициране на категорията на входящите нови данни въз основа на съществуващия набор от данни, които вече са категоризирани въз основа на характеристиките им.

- *Клъстерен анализ* - техника, използвана за групиране на набор от обекти в една и съща група въз основа на общи черти помежду им и различията от другите групи.

- *Data Mining* средства - инструменти за извличане на закономерности от данни, позволяват да се идентифицират скрити модели с помощта на: правило за асоцииране, класификация, регресия или клъстерен анализ на големи набори от данни. Тези средства представят дял от технологията на интелигентните бази от данни и позволяват на потребителя автоматично да открива скрити, а понякога неочаквани връзки между данните.

- *Генетични алгоритми* – техника, основана на процеса на естествената еволюция.

- *Машинно самообучение (Machine Learning)* – техника за обучение чрез предоставяне на знания на системата. Знанието се основава на алгоритмите, които позволяват на системите да се обучават. Целта е да се идентифицира модел и да се вземе подходящо решение въз основа на знанията на системата.

- *Невронни мрежи* - изчислителни техники, които имитират структурата на биологичните невронни мрежи. Използва се както при обучение под надзор (контролирано обучение), така и при обучение без надзор (неконтролирано обучение).

- *Контролирано обучение* - техники, които могат да изведат връзки въз основа на предишни знания (данни за обучение).

- *Неконтролирано обучение* - техники, използващи машинно обучение, при които могат да идентифицират скритите модели от данните без предварителни познания. (например: клъстерен анализ).

- *Predictive Modeling* – изграждане на модел, с помощта на статистически и математически техники за прогнозиране на най-добрия възможен резултат.

- *Регресия* - статистическа техника, използвана за прогнозиране, която ще определи ковариацията между зависимите и независимите променливи.

- *Пространствен анализ* - използва набор от статистически модели, които могат да се прилагат за изследване на географски данни, като например географски информационни системи (GIS) за установяване на местоположение и др.

- *Симулация* - имитиране на поведението на някои сложни системи за прогнозиране и планиране на резултата и измерване на резултатите. Анимираните симулации са особено

подходящи за разкриване на динамични взаимодействия между променливи.

- *Визуализация* - използва се за предоставяне на същността на информацията по опростен начин: графика, диаграма, изображение или друго визуално представяне за опростяване на разбирането.

- *Средства за управление формата на данните* - средствата от този тип позволяват да се трансформират данните между различните формати. Такива средства обикновено са необходими за приложения, извършващи анализи на реални данни.

- *Описателния анализ* на понятията е основа за реализиране на *обобщаване на данни*. Ако се разполага с огромно количество данни е полезно да се направят обобщения като опишат обектите на по-високо абстрактно ниво.

- *Сравнителния анализ* на понятията или класовете осигурява описанието на характеристиките на две или повече колекции от данни.

Хранилището с данни във виртуалното образователно пространство е комбинация от структурирани, полуструктурирани и неструктурирани набори от данни, поради което е важно е да се обмислят различни аналитични модели на данните и средства за тяхното анализиране.

2.2. Семантични знания за пространството

Семантичните хранилища позволяват съхранение и управление на семантично структурирани данни, с възможности за автоматизиране на процесите на взимане на решения. Тези хранилища имат потенциала да предложат по-лесна интеграция на разнородни данни и повече аналитични възможности.

Средствата за семантичен анализ на данните позволяват на разработчиците и потребителите да комбинират различна по форма информация: текстове, изображения, звуци и др. Тук се включват средства за извличане на знание, основани на търсене по контекст и търсене по структура. При контекстното търсене се използват знания за намирането на информацията на основата на съдържанието ѝ, а при търсенето по структура - на формата ѝ. Технологиите за осигуряване на семантични знания са основани на няколко основни подхода за представяне на знанията [91]: концептуални графи, онтологии и невронни мрежи.

- *Концептуални графи* - разработени от Джон Соуа [190] описват език за представяне на знания, първоначално предназначен за разпознаване на семантиката на естествените езици. Основна идея е смисълът на дадено понятие да се разкрива от начините, по които то е свързано с други понятия. От друга страна те са ефективни алгоритми за извеждане на свойства на един обект на основата на принадлежността му към определена категория. Представената в [218] мултиагентна системата, базирана на концептуален граф добавя интелигентно търсене, автоматично известяване, упътвания и управление на персонализирана информация.

- *Онтологиите* са мощен инструмент за представяне на семантично структурирани данни [127]. Те служат за отделяне на знанията за домейна от оперативните знания, с цел многократно използване и анализ. Чрез онтологиите формално се представят знания за областта на основата на някаква концептуализация. Тя предполага описание на множество от обекти и понятия, знания за тях и отношенията между тях (роли). Така в структурата на онтологията явно се отделят два компонента: имена на съществуващи концепции и връзки в домейна.

- *Семантични невронни мрежи* са специфично разширение на изкуствените невронни мрежи. Най-общо невронната мрежа е свързана група от симулационни неврони, които представят изчислителния модел на информационен процес. За да решава конкретен проблем невронната мрежа трябва да бъде обучена, чрез постепенна промяна на теглата на връзките между невроните. Габриела Цибула в [68] представя агент, който използва невронна мрежа за самообучение чрез наблюдение и подпомагане на потребители при изпълнението на специфични задачи.

Представянето на знанията за предметната област е един от най-важните проблеми, защото върху тях се гради способността за разсъждения и експертната дейност на системата. Включването на семантични знания към обучаваща система може да повиши нивото на реализъм при взаимодействие с потребителите, доближавайки го до това на реалното общуване между хората.

2.3. Интеграция на персонални асистенти към виртуалното образователно пространство

С развитието на технологиите се налага изграждане на интелигентни компоненти, които да осигуряват по-голяма гъвкавост и проактивно поведение от страна на системите за електронно обучение. За целта в образователните пространства все повече се използват софтуерните агенти. Всеки агент оперира в определена околна среда за постигане на набелязаните цели [193]. Поведението на агента зависи от неговите способности за разсъждения, вариращи от прост рефлекс до интелигентни, целенасочени процедури за вземане на решения.

Агентът е автономна единица, която възприема своята среда с помощта на сензори и може да повлияе върху нея от своите ефектори. *Интелигентните агенти* са компютърни системи, които могат да оперират гъвкаво и автономно в някаква околна среда за постигане на набелязаните си цели [68]. Интелигентността на тези агенти се обуславя от няколко атрибути: автономност, реактивност, проактивност, социалност и възможност за самообучение. Самообучението позволява на интелигентните агенти да се адаптират към динамичната среда, в която оперират, като по този начин се постига оптимизиране на функционалния им процес и двупосочния информационен поток.

Интелигентните персонални асистенти (ИПА) са софтуерни агенти, които подпомагат потребителите при изпълнение на специфични задачи. Тяхното взаимодействие с хората включва комуникиране, сътрудничество, обсъждане и напътстване [156]. Една от основните разлики между Интелигентните агенти и Интелигентните персонални асистенти е, че последните взаимодействат с потребителите като скриват сложността на поставените задачи, изпълнявайки ги от тяхно име. Една от основните задачи при разработката на интелигентните персонални асистенти е постоянното подобряване на тяхното поведение, основано на предишен опит, компетентност и взаимоотношения с потребителите.

Едно от най-разпространените приложения на интелигентните персонални асистенти е в областта на виртуалното обучение. В [6, 200] е представено контекстно-зависимо виртуално образователно пространство (ВОП) за доставка на електронни услуги и електронно учебно съдържание, удовлетворяващо изискванията на e-Learning обучителния модел. За да се изгради учебно съдържание, като се осигури възможност за интеграция и повторно използване, то трябва да се базира на определен стандарт. За конкретните цели на изследването се разглеждат основните концепции и идеология на SCORM [185]. За дефиниране на тестовите елементи, които могат да бъдат включени в тест се прилага QTI Standard [115]. Характеристиките на различните типове въпроси съответстват на елементите на стандарта и могат да бъдат използвани за управление на обучението и натрупване на статистики за обучаващия процес.

В [93] се разглежда подход за пълна интеграция на компоненти чрез прилагане на микроуслуги. Микроуслуга BDI Rational Agents предоставя ресурси за осъществяване на интелигентно поведение, чрез REST API. В програмната реализация това се постига чрез интегриране на платформата Jadex [116]. Тя е разширение на платформата Jade, като осигурява инструменти за натрупване и споделяне на знания между BDI агенти и

интегрира модела Active Components. BDI (Belief-Desire-Intention) [51] е класическа агентна архитектура, която използва модел на човешката дейност за представяне на ограничена рационалност. Съществено за "ограничената рационалност" е намирането на смислени решения за ефективно използване на наличните ресурси. Обхватът на действие на един рационален агент може да бъде определен като оптимален успех според знанията и уменията, с ограничени ресурси. Тази архитектура се базира на: вяра (модел на околната среда), желания (задачи) и намерения на агента. Разсъжденията на агента са практически и са насочени в две действия: обмисляне какво е състоянието и изготвяне на план за постигане на избраното състояние. Практическите разсъждения са изчислителни процеси, които в реална среда се извършват с ограничени изчислителни ресурси и време. Агентите в пространството са специализирани за различни дейности - могат да доставят образователни ресурси и услуги на потребителите, да извличат и записват ресурси в информационната среда, да обработват справки и анализи и да осигуряват адаптацията в пространството [200].

2.4. Инструменти за подготовка и изчистване на данните в пространството

Когато данните постъпват от различни източници, те се нуждаят от интегриране и трансформация в подходяща форма. Всички източници на данни могат да генерират грешки, които да доведат до грешни модели и лош анализ, и по този начин предполагат неточни заключения. Коригирането на тези проблеми е известно като почистване на данни. Артър Чапман представя основните принципи на почистването на данни в [60]: планиране; организиране на данните ефективно; предотвратяване на грешки; изяснена отговорност; приоритизиране; проверка; обратна връзка; отчетност, прозрачност и възможност за одит; документация.

Основните дейности при почистване на данни са: *валидиране*, *редактиране* и *отстраняване* на данни. Методите за валидиране на данните използват инструменти за проверка, механизми за откриване на конфликти между стойностите на обработваните данни, графики и различни статистически средства. На този етап на диагностика се установяват естеството и причините за грешките. Редактиране и отстраняване на данни се използва за корекция на грешки. Редактирането се занимава с това как да се процедира с проблемните данни: коригиране на стойности, добавяне, преформатиране на данните. Понякога се налага обработка на огромно количество данни. Редукцията има за цел да ограничи представеното множество данни, без това да промени резултата при анализа.

В таксономията на проблеми с качеството на данните, предложена в [60], могат да бъдат дефинирани два общи подхода за почистване на данни: *качествен* и *количествен*. Докато качественият подход включва правила, ограничения и модели за идентифициране на грешки, количественият подход се основава на статистически техники за идентифициране и коригиране на съществуващи грешки в данните. Основни методи са:

- статистически метод - даден атрибут се счита за случайна променлива и ако стойността не е в предвидения интервал, значи е грешна;
- метод за разпознаване на връзките – алгоритми за извличане на данни и машинно обучение за намиране на полета с данни, които не отговарят на съответния модел;
- метод на групиране - данните се групират в класове или клъстери и стойностите, които не попадат в нито един клас, се считат за невалидни;
- метод на разстоянията - използва се метрика за изчисляване на разстояния и по този начин се определя количествената структура на данните;
- класификация - разграничава "валидните" от "невалидните" данни;
- правила за свързаност на данните - даден атрибут се свързва с област на променливата и ако данни не отговарят на правило за принадлежност, те са необичайни.

Съществуват различни софтуерни решения за почистване на данни. Excel е предпочитан инструмент поради потенциала да манипулира редове и колони и по този начин да извършва операции за почистване на данни. Системите за управление на релационни бази от данни са широко използвани, поради тяхната гъвкавост при филтриране и намаляване на липсващи данни и извършване на сложни заявки с данните. Скриптовите езици също могат да се използват в случай на обработка на голямо количество данни (R, Python и Perl). Други софтуерни системи за почистване на данни са: IBM Information Server, Talend Open Studio, Oracle Data Integrator, SQL Server Integration Services, SAS Data Integration Studio, Pentaho Data Integration, Clover ETL и редица други.

2.5. Средства за коректност на данните, търсене на данни и контрол на достъпа в образователното пространство

Тези средства са необходими, поради неочакваните странични ефекти при нарастване на обема на данните и служат за автоматично отчитане и сигнализиране при определени състояния, грешки в данните или неоторизиран достъп до съхраняваните данни. Две подпространства с данни доставят ресурси и механизми за работа на образователното пространство [194]: дигитални библиотеки и аналитично подпространство. Дигиталните библиотеки са хранилища с обучаващи материали, а аналитичното подпространство включва средства за анализ и статистика, способни да подобряват обучаващите процеси. Към дигиталните библиотеки се прилагат строги средства за контрол на достъпа. Могат да се включват и редица инструменти за синтезиране на текст, генератори на основно значение, статистически програми за анализ на данни и т.н.

Към аналитичното подпространство успешно се интегрират механизми за извод като следствие от прилагане на базираните на знания модели на данните. Тези средства позволяват да бъде извършено дедуктивното извличане на нови данни в среда, в която са интегрирани информационно търсене и механизми за извод. Други възможности са свързани с поддържане на активни обекти в хранилищата на данните, които наблюдават определени условия, дефинирани върху състояния на аналитичното подпространство и ако тези условия бъдат удовлетворени, да се изпълнят определени действия.

2.6. Инструменти на интелигентните обучаващи пространства

Съвременното образователно пространство трябва да осигурява контекстно-зависима, адаптивна и персонализирана доставка на образователни услуги и учебно съдържание [200]. То се изгражда от различни информационни ресурси, необходими за организацията и провеждането на обучение в електронна среда и включва хранилища на електронно учебно съдържание, бази от данни за обучаемите и различни средства за подпомагане и управление на обучаващия процес.

Прилагане на методите на изкуствения интелект и машинното обучение в контекста на онлайн системите за обучение е в няколко основни направления:

2.6.1. Оценка на работата и знанията на обучаемите.

За анализ и оценка на учебната работа и за предоставяне на обратна връзка към обучаващите се използват методи, основани на теория на графите, контекстен анализ [92], правила за асоцииране и последователно извличане на данни и анализ на времеви серии [178], класификация и клъстеризация [136] и др. За предсказване успеваемостта на студента се прилагат два модела - когнитивен и статистически. Когнитивният модел се базира на разсъжденията, които студентът използва за решаване на даден проблем. Статистическият модел отчита отделни фактори, които обясняват наблюдаваните данни.

2.6.2. Препоръчващи системи

Препоръки на подходящи обучаващи материали и дейности, съобразени с нуждите и интересите на обучаемия, могат да се базират на различни техники:

- *Knowledge-based recommendation* - препоръката се реализира с помощта на семантични знания или онтологии за предметната област и се основава на контекста и предпочитанията на обучаемия [198].
- *Content-based recommendation* - на обучаемия се препоръчва съдържание, подобно на такова, което е посетено или оценено положително от него в миналото [53].
- *Collaborative filtering* - препоръката е въз основа на сходство с предпочитанията на други обучаеми.
- *Social-network recommendation* - препоръката се базира на профила на обучаемия в социалните мрежи [107].
- *Hybrid recommendation* - препоръките се осъществяват чрез комбинация на няколко метода [199].

2.6.3. Инструменти за подпомагане на работата на преподавателите

Прилагат се различни интелигентни инструменти за подпомагане на преподавателите, за да се адаптират към нуждите на обучаемите и за улеснение на оценяването. Тези инструменти предоставят статистика и информация за усвоени умения [216], автоматична оценка на ефективността на студентите [63] и др. Инструментите за ефективно преподаване се базират на модели за компетентности и аналитични методи за обучение, за да осигурят придобиването на необходимите умения за преподаване.

Съвременните интелигентни пространства доставят разнообразни услуги, реализирани с комбинация от различни методи и технологии. В [138] изкуствена невронна мрежа предвижда резултатите на студентите след приема им във висше учебно заведение на база възраст, входящ бал, разлика между оценките от дипломата и входящия кандидатстудентски изпит и др. Невронната мрежа, която авторите използват, е многослоен перцептрон с един входен, два скрити и един изходен слой, като перцептронът е обучен с помощта на собствена версия на алгоритъма за обратно разпространение на грешката. Мрежата предвижда с точност над 86% потенциалните проблемни студенти.

2.6.4. Стандарти за съвместимост на учебно съдържание и образователни услуги

Хранилищата на електронно учебно съдържание и различни обучаващи ресурси като: курсови проекти, тестови въпроси, дипломни работи и редица др. осигуряват основна инфраструктура за достъп към обекти на обучение в надеждна среда.

Отворените образователни ресурси (Open Educational Resources) са материали за преподаване, учене и научни изследвания, с отворен лиценз, позволяващ безплатен достъп, използване и споделяне от всички без ограничения. Тези ресурси обикновено включват материали, учебници, курсове, модули, видеоклипове, тестове, софтуерни приложения и други инструменти, използвани за подпомагане на достъпа до знания.

Създаването на електронно учебно съдържание е процес, който изисква значителен

опит, време и ресурси. За да се осигури възможност за повторно или многократното използване на учебните ресурси те трябва да са създадени на базата на стандарти за достъпност (W3C, ISO и др.) и оперативна съвместимост (SCORM, IMS и др.). Организации, чиято цел е дефиниране на спецификации и стандарти са: IMS Global Learning Consortium, Advanced Distributed Learning (ADL), ARIADNE, IEEE, ISO и други. По този начин се предоставя рамка за архитектури за системи за електронно обучение, които да подпомогнат съвместимостта на електронно учебно съдържание и услуги между различните системи, пакетиране и управление на съдържанието, предоставяне на метаданни за различни абстрактни обекти в процеса на обучение, управление на курсове и много други.

Спецификация SCORM (Sharable Content Object Reference Model)

Спецификацията SCORM съчетава спецификациите IEEE и IMS в общ документ. Създава се съдържание за обучение с възможност за многократно използване под формата на „обучаващи обекти“ в обща технологична структура. Съществуват различни версии [185]: в SCORM 1.1 се използва XML файл, за определяне на структурата на съдържанието. Възможност за поддържане на метаданни се добавят в следващата версия SCORM 1.2, която е използвана и днес от системи за управление на обучението.

SCORM 2004 се базира на нови стандарти за API и комуникация между съдържанието. Осигурява се възможност да се определи изпълнение на дейности, като се използват обекти от съдържанието и да се записва информация за статуса на учебните цели. SCORM CAM (SCORM Content Aggregation Model) предоставя средства за обединяване на учебни ресурси при усвояване на знания. Процесът на създаване и предоставяне на учебни материали включва създаване на малки електронни части (активи) и свързването им в по-сложни учебни ресурси, които са в определени последователности. SCORM CAM [67, 185] определя техническите методи за изпълнение на тези процеси.

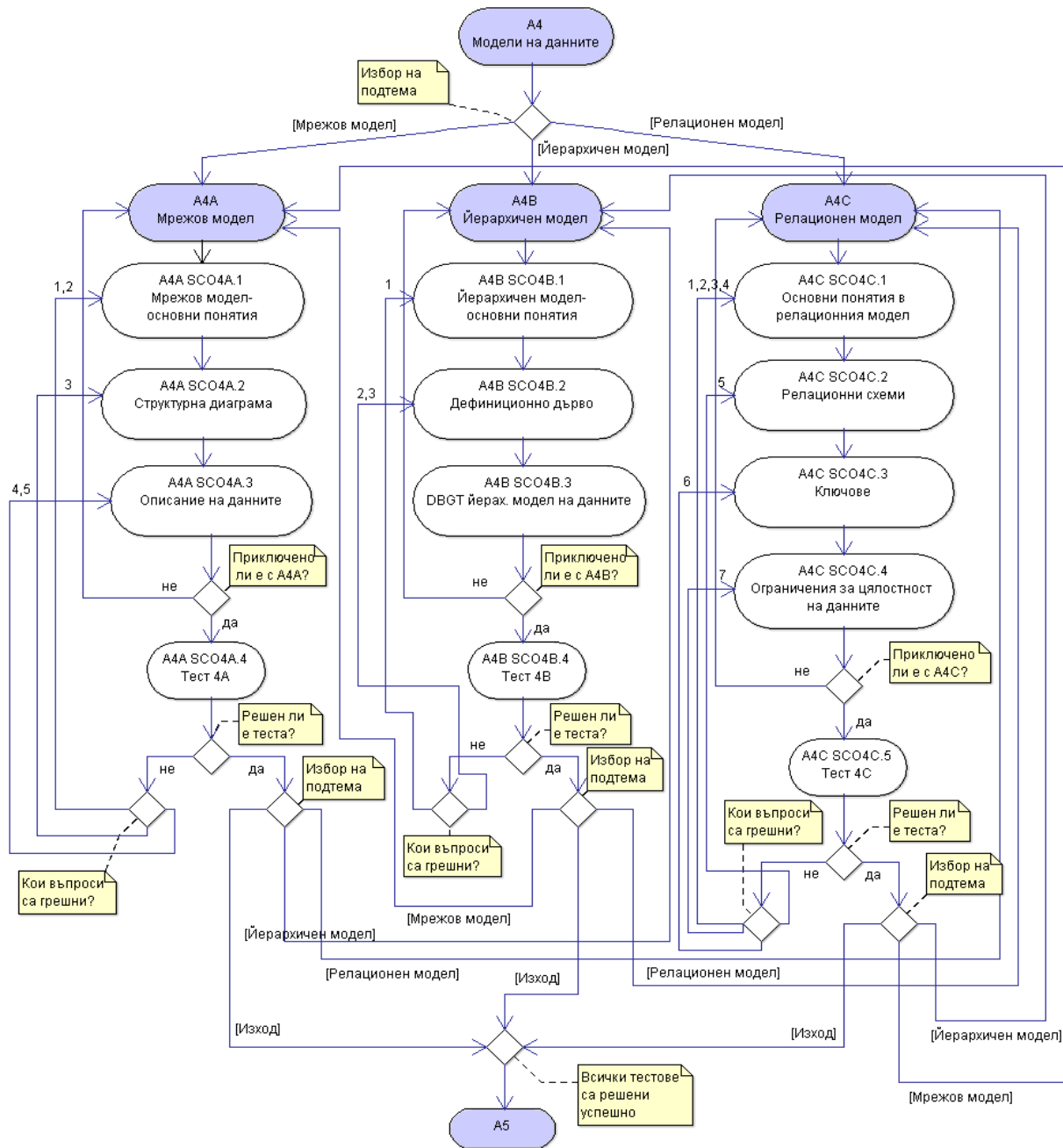
Пример на SCORM пакет за електронно обучение по дисциплината „Бази от данни”.

За целите на това изследване е направено проектиране и реализиране на пакет за дисциплината „Бази от данни”, базиран на SCORM 2004 стандарта. За изготвяне на пакета са изпълнени дейностите:

- проучване на идеологията на SCORM;
- проучване и използване на среда за създаване и тестване на SCORM пакети;
- подготовка на електронното съдържание на преподавания материал и изграждане на отделните ресурси за дисциплината "Бази от данни";
- проектиране на връзките между създадените ресурси в определени последователности;
- създаване на тестове и възможности за обратна връзка към ресурсите;
- прилагане на пакета.

Пакетът се състои от *асети* (основна изграждаща единица на учебния ресурс: текст, елемент за оценяване, звук, изображение или други данни, които могат да бъдат предоставени от уеб клиент), *SCO* (*Sharable Content Object* - множество от асети, което

представя единен учебен ресурс), *учебни дейности (Activities* - задават единици от инструкции), *организация на съдържанието (Content Organization* - показва как дейностите са свързани помежду си в структурирани единици) и *обединения на съдържанието (Content Aggregation* - набор от функционално-зависими обекти).



Фигура 8. Примерна схема на дейности от SCORM пакета по „Бази от данни“

От своя страна пакетът предоставя стандартизиран начин за обмяна на учебно съдържание между различни системи или образователни услуги. Спецификацията *IMS Content Packaging* осигурява общ формат, който всяка система може да поддържа. Пакетът съдържа два базови компонента:

- *Манифестът* е XML документ, описващ съдържанието и свързаните ресурси.

Състои от един основен манифестен файл, който описва самия пакет и от своя страна може да съдържа един или повече манифестни файла, които описват съдържанието на нивото, на което се намират.

- *Съдържанието* описва физическите файлове, които изграждат пакета. Така пакетът съдържа цялата информация и е самостоятелна единица за обучение. Тази единица може да бъде курс, част от курс или колекция от курсове. Структурната организация може да се разглежда като структурирана карта на учебните ресурси, наричана дърво на дейностите („*activity tree*”). Тя описва дейностите и насочва учащия посредством йерархия от дейности, на базата на учебните ресурси.

В примерната схема на дейности, на фигура 8 от разработения SCORM пакет по „Бази от данни“ са означени: обобщението (*aggregation*) с "A" и "SCO" и пореден номер.

Тестовите в пакета са асоциирани с елементи от съответни теми. Чрез тях се прави оценка каква е степента на усвоеност на съответната единица знание и се осъществява навигация, към съответно съдържание. Описанието на създадения пакет е представено в [9]. В университета се използва версия на системата MOODLE (SCORM 1.2), която не поддържа навигация и по тази причина съдържанието на пакета следва последователността на материала, предварително заложен в учебната програма. Използван е продукта eXe, за разработване на различни тестове [191, 192]. В конкретния случай тестовите [67] се използват при оценяване на знания (*Post-test*) и за преход към нов материал (*Pre-test*).

Най-съществените предимства от прилагането на стандартизирано електронно съдържание могат да се определят като: ефективност (води до спестяване на време и ресурси); мобилност на знанията; възможности за споделяне на опит и разпространение на добрите практики; създаване на условия за работа в екип.

ГЛАВА 3. МОДЕЛИ ЗА АНАЛИЗ НА ДАННИ ВЪВ ВИРТУАЛНОТО ОБРАЗОВАТЕЛНО ПРОСТРАНСТВО

В тази глава от дисертационния труд са обобщени резултатите от проведените експерименти и анализи на събраните данни за обучаемите в различни форми на електронно обучение, подробно представени в авторските публикации [159, 165, 166, 168]. В проведените изследвания се търси връзка между статистически методи, машинно обучение, откриване на поведенчески модели и анализ на данни. Разработените алгоритми и софтуер са представени в [140, 141, 169]. Предложени са модели, като резултат от теоретични обобщения на процесите на наблюдение и анализ на дейностите на обучаемите, свързани с Big Data Analytics, Data Mining, веб метрики, машинно обучение и размита логика.

Интегрирането на образователните пространства със средства за извличане на знания от натрупваните данни е необходимо за процеса на персонализиране на курсове за електронно обучение. На базата на получените резултати могат да се въведат допълнителни мерки за анализ и промяна на обучаващите курсове. Това от своя страна е път към повишаване на качеството на обучението [209]. Базирайки се на натрупани данни от работата на среди и системи за електронно обучение с различни потребители, прилагайки средства от областта на *науката за данните*, чрез извличане на знания от данните могат да се взимат различни решения като:

- да се изградят оптимални учебни среди с възможности за персонализирано овладяване на ключови знания, умения и компетенции;
- да се търсят тенденции относно развитието на процесите на електронното обучение и неговото обслужване;
- да се оптимизират техниките за избор на тестови елементи и подходящ вид намеса в дейността на обучаемия;
- да се идентифицират типове обучавани, на които да се предлага подходящо продължение на обучението;
- да се предвидят обучаваните, за които има опасност да не се справят с обучението;
- да се правят анализи. относно степента на придобиване и забравяне на знанията от обучаемите за различни интервали от време и за различни типове задачи, както и сравнение на показателите през годините и т.н.

Целта на настоящото изследване е да се очертаят тенденциите в анализа на данните в сферата на обучението. Направеното проучване в [22] на състоянието в научните бази от данни, относно ключовите думи: *data science*, *data analysis* и *data science in education*, в списания и академични журналы показва че след 2000 г. се проявява устойчива тенденция на растеж на броя на публикациите в областта.

С достъпа до интернет, моделът на оценяване на обучаемите се променя. Те вече решават казуси, работят по проекти, извършват изследвания, и чрез анализиране на натрупаните данни може да се проследи как се справят с предизвикателствата в учебния модул от началото до края.

За оценяване на знанията, уменията и компетентностите на обучаемите могат да се използват различни форми и методи. Крайната оценка е комплексна и включва

компоненти, които участват във формирането ѝ с различни тегла. В зависимост от изследваните проблеми, критериите за оценка могат да бъдат изразени чрез количествен, качествен, размит или смесен модел. Различни техники за мултикритериално взимане на решение (MCDM) и техни приложения са дадени в [160]. Освен критериите за оценка, теглата на критериите също могат да бъдат изразени чрез размити числа или размити отношения, за да се изрази значението на критериите. Съществуват различни подходи за оценяване на знанията на обучаемите в зависимост от целта. Проблеми, свързани с прогнозно моделиране за набиране на ученици се обсъждат в [112]. Експерименти, свързани с разработване на модели за прогнозиране въз основа на данни за прием на студенти са представени в [120]. Идеята за MCDM при клъстери би могла да се приложи при класиране и разпределение на обучаемите в групи, с цел тяхното оценяване [175]. Могат да се взимат предвид и допълнителни коефициенти за значението на експертните мнения при определянето на крайното решение.

Определянето на обективна обобщена оценка на студентите изисква разглеждане на различни аспекти на придобитите знания и компетентности. Нека разглеждаме N подкритерии $C_1 = \{C_{1,1}, C_{1,2}, \dots, C_{1,N}\}$, свързани с различни оценъчни компоненти. Преподавателят, може да определи съответни теглови коефициенти, които изразяват относителна значимост между подкритериите. Оценката на теоретичните познания може да се направи чрез автоматично генериране и оценяване на различни тестове.

В [191] генерирането на тест се поддържа от специализирани онтологии, от два интелигентни агента - Оперативен асистент и Оценяващ асистент. Първият генерира теста като формира въпроси подбрани на случаен принцип по дадена тема, използвайки база от знания. Вторият проверява отговорите на потребителите като използва онтология за областта. В [192] се предлага специфичен подход за представяне на знания за обучение в три нива: *Domain level* (основни строителни блокове, моделирани в хранилище от свързани учебни единици); *Extractor level* (избор на строителни блокове според тестовата структура); *Generator level* (генериране на тестови въпроси).

За да се получи обобщена оценка за обучаемите, като се вземат предвид получените резултати от отделните компоненти може да се приложи функция на полезност:

$$(1) \quad \max_i \sum_{j=1}^N w_j e_{ij}, \quad \text{за } i = 1, 2, \dots, S,$$

където w_j е коефициент за относителна значимост между под-критериите за оценка, e_{ij} представлява оценъчен резултат за i -ти студент към j -ти критерий.

За оценка за работата на обучаем, отчитайки отделно придобитите знания, компетентности и умения може да се приложи моделът представен в [126], при който функцията на полезност е както следва:

$$(2) \quad \max(\alpha \sum_{j=1}^N w_j e_{ij} + \beta \sum_{k=1}^M v_k e_{ik}) \quad \text{за } i = 1, 2, \dots, S;$$

$$(3) \quad \sum_{j=1}^N w_j = 1, \quad \sum_{k=1}^M v_k = 1, \quad \alpha + \beta = 1,$$

където w_j е коефициент за относителна значимост между подкритериите за оценка по отношение на теоретичните знания, v_k е коефициент за относителна значимост на подкритериите по отношение на практическите умения, e_{ij} представлява оценъчен

резултат за i -ти студент към j -ти критерий по отношение на теорията, а e_{ik} е оценката за i -тия студент по k -ти критерий по отношение на практиката. В [45] са въведени тегловите коефициенти α и β , които показват как теоретичните знания и практически умения участват в обобщената крайна оценка. Допълнително ограничение за тези коефициенти е изразено чрез: $\alpha + \beta = 1$. Нормализацията е необходима, за да се осигурят сравними размери между коефициентите за относителна значимост на подкритериите (w_j и v_k), допълнителните теглови коефициенти (α и β) и подкритерийни оценки e_{ij} и e_{ik} . За разлика от [45], където фокусна точка е да се предложи класиране на обучаемите, моделът в провежданите тук изследвания има за цел да прогнозира „студентите в риск“ по време на обучението им на базата на получаваните оценки.

Друг подход за динамично оценяване на обучаемите с интуиционистки размити оценки е представен в [142]. Оценките за ниво на усвоеност $\mu(x,t)$ и неусвоеност $\nu(x,t)$ на единица знание от обучаем x в момент време t са реални числа от множеството $[0,1]$. Степента на несигурност $\pi = 1 - \mu - \nu$ представя случаите, когато отговорите не могат да бъдат дефинирани точно или е допусната техническа грешка. Навсякъде наредените двойки се дефинират в смисъла на теорията на темпорални интуиционистки размитите множества.

В началото, когато все още не е извлечена информация за обучавания обект x , всички оценки получават нулеви стойности. Текущата оценка $(k+1)$ -вата по ред, се пресмята на базата на предходните оценки по формулата:

$$(4) \quad \langle \mu_{k+1}, \nu_{k+1} \rangle = \left\langle \frac{\mu_k k + m}{k+1}, \frac{\nu_k k + n}{k+1} \right\rangle$$

където $\langle \mu_k, \nu_k \rangle$ е предходната оценка, а $\langle m, n \rangle$ е оценката от текущата задача, за m, n в интервал $[0,1]$ и $m+n \leq 1$. Така в оценката за всяко умение се съдържа информация както от предходните събития, така и от последно решавания проблем. Разглежданият модел на електронно оценяване, предлага не само проследяване на промените на параметрите на обучавания обект, но и отчитане на състоянието на вече възприетите знания и възможността за приложение, както и възможности за оценяване и промяна на темите за обучение и критериите за оценка.

Един подход за автоматично оценяване на тестови единици със свободен отговор, представен в [119], изчислява степента на близост на понятията от областта с думи от отговора на обучаемия. Отговорите на обучаемите се оценяват, използвайки предметна онтология. Системата сканира получените думи и извършва търсене в речника на онтологията, използвайки q -gram метрики. Q -грам метриката е мярка, базирана на символи, която изчислява степента на сходство между два низа и може да бъде описана по следния начин: нека Σ е крайна азбука, Σ^* е множеството на всички низове над Σ и Σ^q е множеството на всички низове с дължина q над Σ за $q = 1, 2, \dots$

Един q -gram е стринг $v = a_1 a_2 \dots a_q$ в Σ^q . q -gram разстояние между два стринга x и y се дефинира като:

$$(5) \quad D_q(x, y) = \sum_{v \in \Sigma^q} |G(x)[v] - G(y)[v]|$$

където $G(x)[v]$ е броя срещания на v в x .

Анализи на данни във виртуално образователно пространство

За оценяване на знанията на обучаемите се използват различни форми и методи на изпитване като: online или offline тестове, отговори на отворени въпроси, тестване и разчитане на код, решаване на задачи и др. Крайната оценка по дисциплината често е комплексна и включва няколко компонента, участващи във формирането ѝ с различни тегла. Тя трябва да отразява различните страни от подготовката на обучавания (теоретични знания или практически умения), да е съобразена със спецификата на оценяването, с възрастовите и индивидуални особености на оценяваните и др. Софийският университет реализира курс, посветен на „Умения от 21 век“, изследващ обучение, основано на компетентности, и методи на активно учене [30].

Експеримент за оценяване и аналитични изследвания на данните за обучаемите, е представен в [159]. За целите на анализа разглеждаме конкретен подход за оценяване на обучаемите, при който оценката в края на семестъра се определя като претеглена средна стойност (weighted mean \bar{x}) от всички оценки през семестъра. Всяка оценка има важност или тегло (weight). Използваната функция има вида:

$$(6) \quad \bar{x} = \frac{\sum xw}{\sum w}$$

където x е оценката, w е теглото на всяка оценка.

Оценката на студентите по предмета Математика в края на семестъра се определя въз основа на следните изпитвания: Тест 1 Множества (*Test 1 Sets*), Тест 2 Логика (*Test 2 Logic*), Тест 3 Геометрия (*Test 3 Geometry*), Тест 4 Статистика (*Test 4 Statistics*), Тест 5 Вероятности (*Test 5 Probability*), изпит в края на семестъра (*Final Exam*), контролни (*Quizzes*), домашна работа (*Homework*), проекти (*Projects*) и работа в часа (*Class Activities*). Всеки критерий има степен на важност като: Test 1 Sets, Тест 2 Logic, Test 3 Geometry, Test 4 Statistics и Test 5 Probability влияят с по 9% на оценката в края на семестъра. Final Exam, Quizzes и Homework са с 15% от оценката, докато Projects и Class Activities са с по 5% от оценката в края на семестъра. Оценките на тестовете (Tests) и оценката в края на семестъра (Final exam) имат общо тегло от 60%, което е определящо за оценката. Homework, Quizzes, Project, Class activities определят 40% от оценката. Обединяваме ги в категория наречена Activity (дейности в клас и извън клас). Функцията има вида:

$$(7) \quad \bar{x} = \frac{\sum T1*0.09 + T2*0.09 + T3*0.09 + T4*0.09 + T5*0.09 + FinEx*0.15 + Act*0.40}{\sum w}$$

където \bar{x} е оценката в края на семестъра, w е теглото на всяка оценка. Разглеждаме как получената оценка по отделните тестове влияе на оценяването и как дейностите в часа и извън часа се отразяват на оценката в края на семестъра.

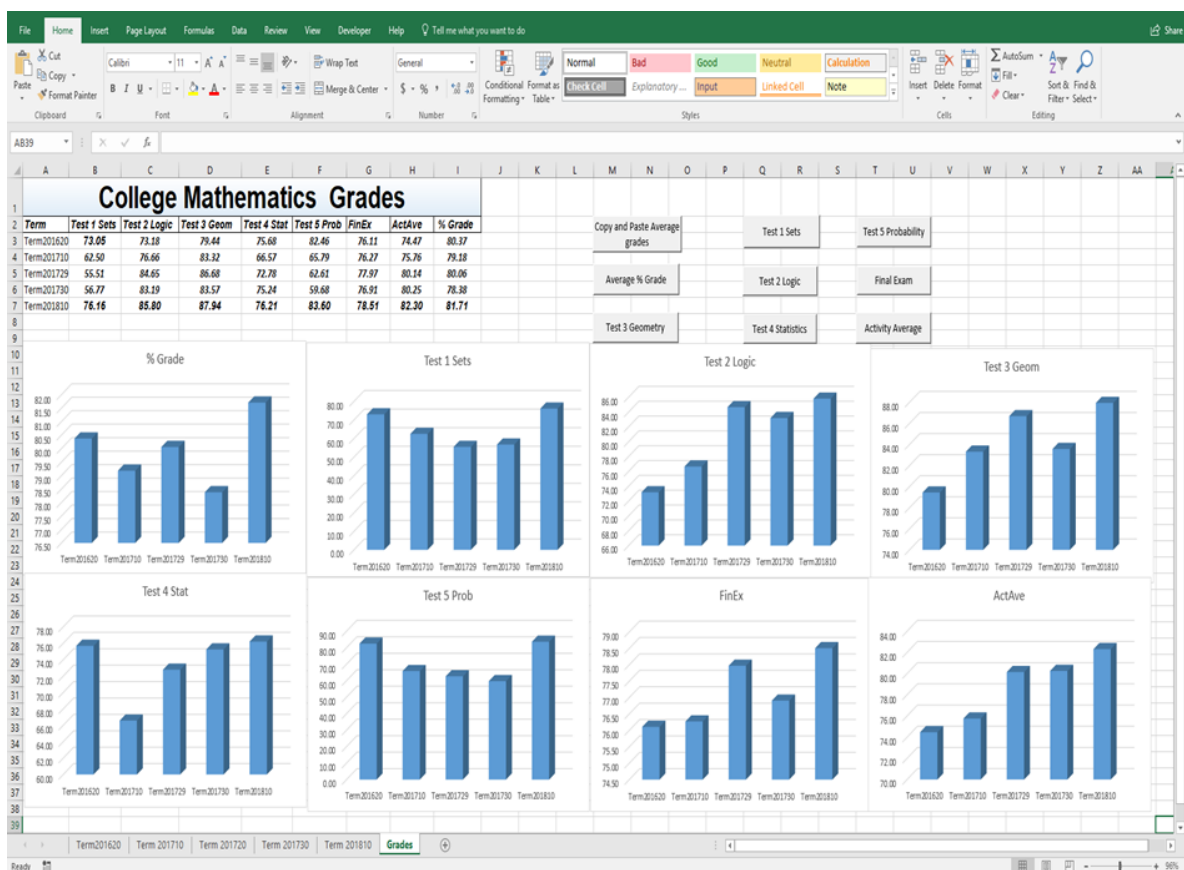
Освен точките, натрупвани по отделните компоненти, съхраняваме и данни за всеки студент: Age (години е студент), Gender (пол), New Student Experience (NSE), FullTime-PartTime (кредитите за предмета са 12 или повече), Student Program (тип на обучението), Times Taken Course (за кой път студентът изучава предмета).

3.1. Използване на електронни таблици за аналитични изследвания на данните за обучаемите

Електронните таблици (в частност Excel) предлагат ефективни функции за обобщаване на тенденциите в данните. Филтрите (*Filters*) дават възможности за подбор на записи. Условното форматиране (*Conditional formatting*) оцветява данните, които отговарят на определени критерии и помага да се откриват отклонения в данните. Графиките (*Charts*) визуално показват тенденции и отклонения. Графичните таблици (*Pivot tables*) улесняват потребителите при визуализиране на данни [212].

Голямото количество готови функции в тези системи и възможностите за създаване на формули позволяват анализи върху данните като: намиране на линейна функция, показваща тенденция и намиране на коефициент на корелация. Макросите (*Macros*) действат като набор от функции и команди, които се използват за автоматизиране на задачи и възможностите им са много големи [79].

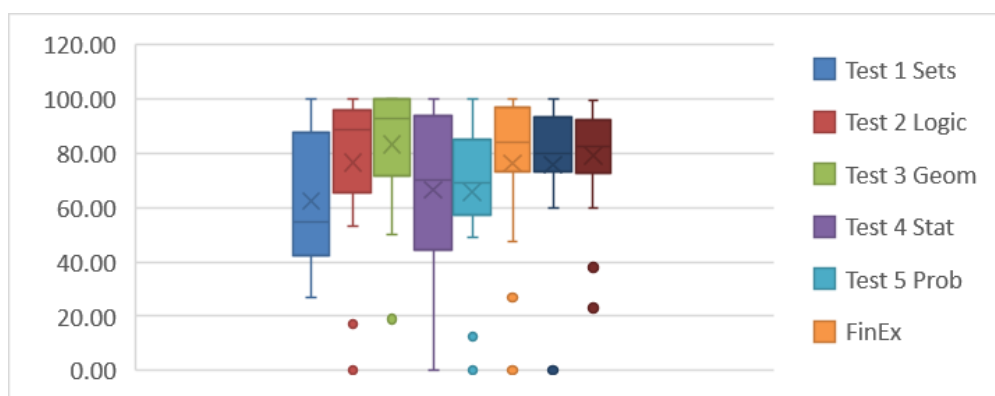
На следващите графики (фиг. 9) може да се види, че средните оценки за Test 2 Logic, Test 3 Geometry, Activity Average и Final Exam са се повишили от Term 201620 до Term 201810. Създавайки линейни диаграми за тези четири групи данни, намираме линейно уравнение за тях и стойността на R^2 , където R е корелационен коефициент. Използвайки макроси, можем да създадем линейни графики за четирите групи относно средната оценка за всеки семестър.



Фигура 9. Графики относно средните оценки за Test 2 Logic, Test 3 Geometry, Activity Average и Final Exam.

От разпределението на оценките за период (term 201730) можем да кажем, че има голяма вариация в оценките на Test 1 Sets и Test 5 Probability. На тези тестове, половината от обучаемите са се справили по-зле от 25% от обучаемите по всички останали групи. През този семестър е малък броят на обучаемите, чиито успех е много по-нисък от успеха на останалите. Вижда се подобряване на успеха по Test 2 Logic.

От следващата графика (фиг. 10) се вижда, че няма голяма разлика в оценките на обучаемите на всички тестове през периода (term 201810). Това показва добро усвояване на учебния материал. Най-ниски са оценките по Test 4 Statistics, където 75% от обучаемите имат оценки по-ниски от 50% от обучаемите по Test 1 Sets, Test 2 Logic, Test 3 Geometry и Test 5 Probability.



Фигура 10. Графика, относно разпределението на оценките по модули

Използвайки инструменти на Excel се извърши първоначален анализ на резултатите от представянето на студентите по математика в продължение на пет семестъра. Промяната на стратегиите за преподаване оказва влияние върху начина, по който обучаемите усвояват материала.

Електронните таблици се използват успешно за подготовка на данни, за визуализиране на резултатите чрез графики и за намиране на линейни функции, които показват тенденциите в разпределение на оценките по отделните категории. Статистическите възможности на електронните таблици позволяват да се определят обучаемите, чиито оценки са много по-ниски от мнозинството, каква е минималната и максималната оценка и др. Използването на макроси позволява бърза и многократна обработка на групи от данни. В резултат на изследването бе направена препоръка за промяна на последователността в изучаването на две дисциплини.

3.2. Аналитични изследвания на данните в обучаващото пространство със системата „Orange“

Днес са достъпни различни софтуерни инструменти за извличане на данни. Сред най-популярните са: RapidMiner, RapidAnalytics, WEKA, PSPP, KNIME, Orange, Apache Mahout, jHerWork, Rattle, GhostMiner, XENO, SAS Enterprise Miner, Polyanalyst и IBM SPSS modeler. Всеки софтуер разполага с методи за анализ и интерпретация на информацията от групи данни.

За целите на това изследване е разработен инструмент за идентифициране и прогнозиране на причините за изоставане или отпадане на обучаеми, представен в [164]. Използвана е системата “Orange Data Mining system” [139], софтуер за машинно обучение и извличане на данни с отворен код, създадена на Python.

Алгоритмите за машинно обучение използват изчислителни методи за „изучаване“ на данните. Прилагат се два основни типа техники: контролирано обучение, което обучава модел на известни входни и изходни данни, така че да предсказва бъдещи резултати, и неконтролирано обучение, което намира скрити модели или присъщи структури във входните данни.

Контролирано обучение

Основната задача при контролираното обучение (*Supervised learning*) е да се създаде модел от обозначени (етикирани) данни, което позволява да се правят прогнози за бъдещи данни. Основни техники са: класификация, когато етикетите на класовете са дискретни и регресия, когато резултатът е непрекъсната стойност. Този тип се нарича обучение направлявано от учител, защото системата е представена с примерни входове и съответните резултати, а целта е да се научи общо правило, което определя входовете към изходите.

Задачата за изграждане на класификационен модел е една от най-популярните задачи в сферата на машинното обучение, когато етикетите на класовете са дискретни стойности. Целта е да се разпределят обекти в точно определени предварително зададени класове. Когато броя на класовете е 2, разглеждаме задача за бинарна класификация, ако е 3 или повече, задачата е многокласова. Класификационните модели са приложими в практиката при решаване на различни бизнес задачи. След създаване на модела трябва да се определи точността му при постъпване на нови данни, чрез прилагане на множество тестове и преглед на различните показатели, които определят колко добре се справя модела.

Средства на системата Orange за изграждане на модел за **класификация** и **регресия** са: Logistic Regression, Naïve Bayes Classifier, Support Vector Machines (SVM), Decision Trees; Artificial Neural Networks. Към тази група принадлежат и методите: k nearest Neighbours, Linear Regression, Random Forests, Deep learning, и др.

- *Logistic Regression*. Измерва взаимовръзката между зависима променлива и една или повече независими променливи, като изчислява вероятности, използвайки логистична функция. Прогнозира вероятността от възникване на събития чрез приспособяване на данни към функцията. Най-общо, регресиите могат да бъдат използвани в приложения от типа: предсказване на приходите от

конкретен продукт; измерване на успешния процент от маркетингови кампании; изчисление на кредитни резултати и др.

- *Naïve Bayes Classifier*. Този метод е базиран на известната теорема на Bayes за вероятности. Основава се на допускането, че ефектът от стойността на атрибута на даден клас е независим от стойностите на другите атрибути, като това допускане се нарича условна независимост на класа. Алгоритъмът работи много добре в приложенията от различни области.

- *Support Vector Machines*. Бинарен класифициращ алгоритъм, при който в процеса на обучение се анализират входните данни и се идентифицират шаблони в многомерното пространство по категории. При дадено множество точки от два типа в N-мерно пространство, този алгоритъм генерира N-1 измерна хиперравнина, за да раздели тези точките в две групи. Например, ако имаме няколко точки от два типа на лист хартия, които са линейно разделени, алгоритъмът ще намери правата линия, която разделя точките от двата типа и намираща се възможно най-отдалечено от тях. SVM алгоритмите са класифицирани в две категории: линейни (данните са разделени от хиперравнина) и нелинейни.

- *Decision Trees*. Този вид алгоритми представляват минималният брой въпроси (от тип да/не), които трябва да бъдат зададени, за да се изчисли вероятността за вземане на правилно решение. Методът позволява да се подходи към проблема по структуриран и систематизиран начин и да се достигне до логическо заключение.

- *Artificial Neural Networks*. Тези алгоритми са вдъхновени от структурата и функцията на био-невронните мрежи. Най-общо се използват при проблеми свързани с регресия и класификация, но те са огромна област, състояща се от стотици алгоритми и могат да се използват при решаването на различен вид проблеми.

В системата Orange са създадени редица инструменти за изграждане на модели за класификация и регресия. В следващата подточка са представени експерименти за оценяване и прогнозиране на знанията на обучаемите в образователно пространство с прилагане на инструментите: Logistic Regression, Naïve Bayes Classifier, Support Vector Machines (SVM), Decision Trees, Artificial Neural Networks и др.

За анализ на данните в образователно пространство, основни входни данни са точките, получени от обучаемите на различни оценъчни компоненти, а изходни са съответните крайни оценки, поставени от преподавател. За решаване на класификационния проблем за оценяване и прогнозиране на знанията на обучаемите последователно се прилагат различни инструменти чрез приложението *Orange Data Mining System*.

Основни задачи за подготовка на данните за анализ са [60]: почистване на данните (Data Cleaning), интеграция на данните (Data Integration), трансформация на данните (Data Transformation), редуциране на данните (Data Reduction).

Почистването на данните се налага за гарантиране, че данните са използвани, правилни, последователни. *Структурни грешки* могат да получат по време на прехвърляне или измерване на данни. При комбинирани набори от данни от множество

источници могат да възникнат *дублиращи* се данни. Данни, които не са свързани с решавания проблем се определят като *неподходящи*. Отличителните стойности (*Outliers*) могат да доведат до проблеми с някои модели. Когато не разполагаме с пълното множество от данни за дадена характеристика имаме *липсващи данни* (*Missing Data*). Процесът на подготовка на данни, изисква изтриване, коригиране, идентифициране на грешки, подредба или друга обработка, като се запазят само потенциално полезните данни.

3.2.1. Проблеми при подготовка на данните за анализ и подходи за решаване

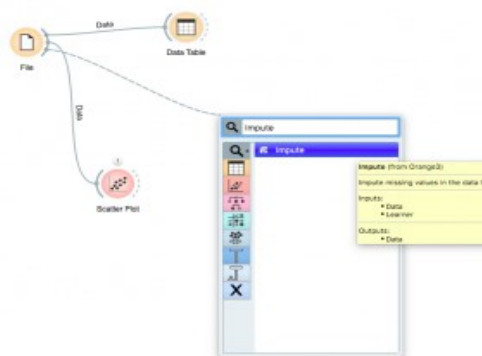
Подготовката на данните за анализ се свързва с взимане на решения за обработка в редица ситуации:

- **Проблем - липсващи стойности**

Проблеми в работата на инструментите за машинно обучение може да възникне, когато част от данните на дадена характеристика липсват. Това може да се отразява негативно върху точността на получените модели. Данните трябва да се преработят във вид, позволяващ преодоляването на проблема. В практиката се използват различни подходи като:

- *Изтриване* - радикален подход е изтриване на редовете или колоните с липсващи стойности. Това е добър подход при големи обеми от данни [208], защото ефектът от изтриване на няколко реда няма да окаже силно влияние.

При избор на инструмент „File“, чрез системата *Orange* се извършва зареждане на данните, след това се избира инструмент „Impute“ (фиг. 11).



Фигура 11. Инструменти за почистване на данните.

В случай, че липсват много стойности в колона (например над 50%) и слаба връзка с останалите стойности, тя може да бъде премахната. Но ако има корелация между колоната с липсващи стойности и останалите характеристики, то трябва да се търсят начини за заместване на стойностите.

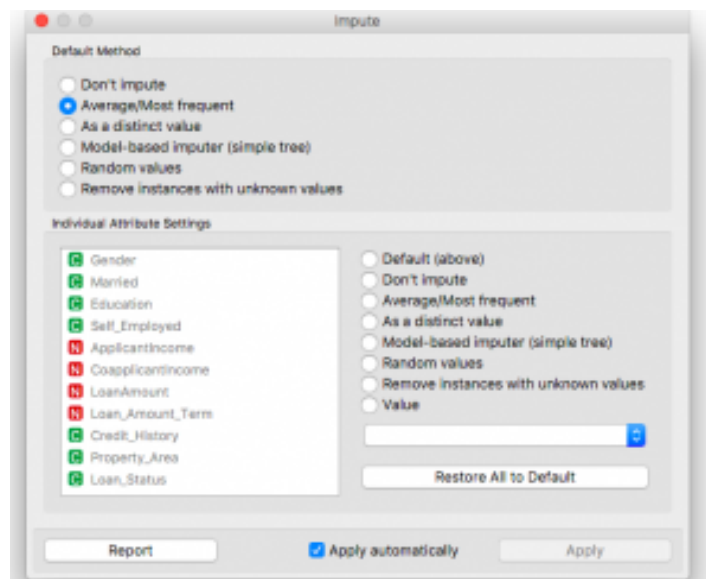
- *Заместване*. В зависимост от типа на данните се търси подход за заместване на липсващите стойности. Статистически данните се разделят на категорийни/качествени (като: семейно положение, пол, степен на образование и др.) и количествени/числови (като: доход, възраст и др.).

При *категорийните* данни обикновено се прави заместване с най-често срещаната стойност (модата). При количествени данни се използват различни методи като:

- заместване с определена константа;
- заместване с произволни стойности;
- заместване със средна стойност или медиана;
- изчисляване на стойност чрез регресия;
- определяне на стойност, базирайки се на създаден модел на данните

и други техники.

Предложени са различни методи за вмъкване, които могат да се прилагат чрез инструмент „*Impute*“ (фиг. 12). Чрез опцията „*Remove the rows with missing values*“, редовете се изтриват. Други възможности са: *Random Values*, *Distinct Value*, *Model-Based*.



Фигура 12. Подходи за заместване на липсващи данни.

- **Проблем - мащабиране (нормализация) на данните**

Нормализацията на данните трябва да промени стойностите на числовите колони в набора от данни до общ мащаб. Това трябва да стане без да се изкривяват разликите в диапазоните от стойности. Например, разглеждайки двете характеристики, възраст и доход: доходът варира от 0–20000 и повече, докато възрастта варира обикновено от 0–100. Характеристиките са в много различни граници, като доходът е около 1000 пъти по-голям от възрастта. Когато се прави анализ, като например линейна регресия - дохода ще повлияе по-същество на резултата поради по-голямата му стойност, а не би трябвало да се отчита като по-важна характеристика.

В етапа на предварителната обработка на данните (*Data preprocessing*), се прилага техниката мащабиране на характеристики (*Feature scaling*). Трябва да се извършат трансформации, така че стойностите да бъдат в един и същ порядък.

За машинно обучение не всеки набор от данни изисква нормализиране. Но при работа на определени модели с немащабиранни данни, например при такива, които изчисляват коефициенти, могат да се получават резултати с голяма грешка. За целта се прилага мащабиране като за всяка характеристика се изчислят теглови коефициенти.

Алгоритми, които изискват данните да бъдат мащабирани са например: алгоритмите, изчисляващи разстояние: k най-близки съседни (*k-Nearest Neighbors*); метод на опорните вектори (*Support Vector Machines*); k -means клъстеризация (*k-means clustering*). Алгоритми, изчисляващи коефициенти: регресионни алгоритми (линейна, нелинейна, логистична регресия и др.), невронни мрежи и др.

Прилагат се различни методи, като например [181]:

- *MinMax* – изважда се минималната стойност на характеристиката да бъде 0, а стойността се получава като се раздели на новата максимална стойност, която е разликата между първоначалните максимални и минимални стойности (max-min).
- *LogNormal* - преобразува всички стойности в логаритмична скала.
- *Zscore* - данните са трансформирани чрез изваждане на средната стойност и разделяне на стандартното отклонение.
- *Robust* - изважда се медианата (Q_2) и полученият резултат се разделя на интерквартилния размах ($Q_3 - Q_1$).
- *TanH* - всички стойности се преобразуват в хиперболична тангента.
- *MaxAbs* - мащабира данните за всяка променлива по нейната максимална абсолютна стойност.

Когато данните са мащабирани, всички характеристики са равно поставени, така липсва възможността алгоритъмът да възприема една променлива за по-важна от друга. Това подобрява финалните резултати и възможността да се направи по-точна оценка на качеството.

- **Проблем - отличителни стойности (*outliers*)**

Стойности, които значително се различават от останалите се наричат отличителни или екстремни. Те могат да бъдат необичайно високи или ниски. Има случаи, при които се интересуваме точно от стойности, които се различават много от останалите, именно те съдържат важната информация.

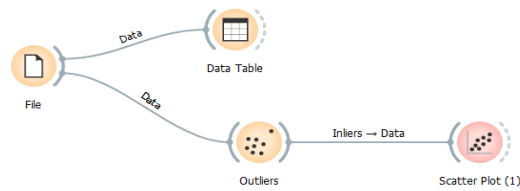
Чрез инструмент *Z-score* може да се направи анализ за такива данни. Определя се число, което показва колко се отдалечава дадена точка от средната, измерено в стандартни отклонения. В този случай, всяка стойност, която е по-ниска или по-висока от средната стойност плюс или минус три пъти стандартното отклонение се определя като отличителна стойност. Други подходи за откриване на отклонения са например линейни модели като: метод на главните компоненти, линейна регресия и др.

Наличието на такива стойности се отразява негативно върху работата на линейни модели (регресионен и дискриминантен анализ) и SVM.

Отличителните стойности могат да бъдат тълкуват като *липсващи стойности* и да бъдат обработени като такива. Друг подход е *дискретизацията*. При този процес се трансформират непрекъснатите променливи в дискретни, като стойностите се разпределят в определен брой интервали. Дискретизацията може да решава проблема с отличителните стойности, като те се поставят в най-ниския или най-високия интервал. След извършване на дискретизация тези променливи вече се третират като категорийни. Алтернативен подход е поставяне на горен и долен праг и *заместване с минимална и максимална допустима стойност*. Подходът позволява обработка на отличителните стойности без да

се премахват записи от данните, но може да изкриви разпределението и връзката между отделните променливи.

За идентификация и обработка на такива стойности на данните в системата *Orange* се използва инструмент *Outliers* (фиг. 13).



Фигура 13. Откриване на отличителни.

Идентификацията и обработката на отличителни стойности в данните позволява получаването на по-точни стойности на метриците за оценка на работата на моделите, и по-добро цялостно представяне на модела.

- **Проблем – небалансирана извадка от данни**

В някои случаи, при решаване на задачи за класификация с методите на машинното обучение попадаме на извадка, при която броят на обектите, принадлежащи към един клас, е значително по-различен от този на другите класове така представителите на класовете в целевата променлива не са равномерно разпределени.

Oversampling е техника за балансиране на извадката с данни, като се увеличава броя на обектите в по-малкото множество (*minority*), за да се изравни с броя в по-голямото множество с данни (*majority*).

Undersampling е техника, с която, се намаляват случаите в класа с повече представители докато броят им да се изравни с този в класа с по-малко представители.

При прилагане на *oversampling* за балансиране на извадка се запазват всички оригинални записи, докато при *undersampling* се премахват редове от множеството с повече представители и може да се изгубят ценни данни. Възможно е приложение на комбинация от двете техники. Изборът на техника зависи от наличните данни и от решаваната задача.

3.2.2 Оценяване и прогнозиране на знанията на обучаемите във виртуалното образователно пространство

Тук се предлага като **многостъпков процес**, метод за оценяване и прогнозиране на знанията, уменията и компетентностите на обучаемите във виртуалното образователно пространство [163]. **Методът включва пет стъпки** и може да бъде адаптиран към различни курсове на нововъзникващи технологии (*Emerging technologies*) във виртуалното образователно пространство.



Фигура 14. Многостъпков процес на анализ и прогнозиране на знанията на обучаемите

Стъпка 1. Избор на начин за оценяване. Определяне на ключови знания и компетентности за изучаваната технология. Определяне на степен на тежест (важност) на всяка анализирана компетентност.

- Основните **теоретични знания** се оценяват с помощта на компоненти като: *междинни тестове, решаване на задачи и казуси, изпити, обобщена дискусия* и др. Тези компоненти оценяват придобитите знания и способностите за разбиране на изучавания теоретичен материал. Използват се отворени тестови въпроси, въпроси от вида „избор на верен отговор“, изброяване и сравняване на обекти, даване на примери за понятия, разясняване и използване на алгоритми. Както и въпроси, свързани с обясняване, интерпретация и визуализиране на решения.

- Основните **компетентности** се оценяват от възможностите на студентите да прилагат придобитите знания за вземане на решения в: *контролни и домашни работи, курсови задачи и проекти*, в които се решават нови задачи, извършва се анализ на решения, определят се рискове, правят се самостоятелни изводи и заключения.

Стъпка 2. Процес на оценяване. По време на цялостния процес на обучение се провеждат и процедурите за оценяване. Резултатите се натрупват и съхраняват с цел профилиране на студентите, последващи анализи и прогнозиране в оценяването на знанията и уменията на нови студенти.

В получените резултати се търсят зависимости в отделните оценъчни компоненти между теоретичните и практическите знания, умения и компетентности. В някои случаи студент получава висока оценка на тестовете и слаба оценка на контролната работа или обратно. Създаването на автоматизиран алгоритъм за оценяване при налични специфични стойности на оценъчните компоненти изисква нестандартно решение.

Стъпка 3. Анализ на натрупаните данни от проведените обучения в реална среда. Има различни алгоритми за машинно обучение. Основни входни данни са точките, получени от оценъчните компоненти, а изходни са съответните крайни оценки на студентите по дисциплината. При методите за обучение, част от примерните данни се ползват за обучение на алгоритъма, а друга – за тестване. При недобри резултати от тестването, процесът на обучение може да се повтори или да се прецени, че избраният подход не е добър за решение на конкретния проблем.

Стъпка 4. Създаване на модел за предвиждане. На базата на натрупаните данни се създава класификатор, който извършва прогнози за текущи оценки по дисциплината. Системите за анализ на данни като: SPSS, Orange, Weka и други предлагат средства, на базата на алгоритми за: дърво на решенията, логистична регресия, теорема на Бейс, невронни мрежи и редица други. След приключване на обучението на алгоритмите, реализираните модели могат да бъдат прилагани върху нови входни данни.

Стъпка 5. Оценка на точността на предвиждане. След обучението на модела се извършва тестване за точност и прецизност на работата на модела.

Доколко е добър един класификатор се определя от стойностите на съответни метрики за оценка на качеството. Не е достатъчно да се гледа само една от тях. Трябва да се вземат под внимание няколко метрики, като избора на това кои са по-важни е в зависимост от задачите и съответните цели.

Confusion Matrix е един от най-популярните начини за оценка на качеството на класификацията [181]. Представява матрица с размер $N \times N$, където N е броят класове на целевата променлива. Чрез прилагане на този инструмент могат да се съпоставят определените от класификатора и действителните стойности по 4 метрики: True Positive (*TP*), False Positive (*FP*), False Negative (*FN*) и True Negative (*TN*).

- *Общата точност* е показател, който дава информация за това каква част от всички случаи са правилно класифицирани:

$$(8) \quad Accuracy (ACC) = (TP+TN) / (TP+FP+FN+TN)$$

Общият брой правилно класифицирани обекти се разделя на всички случаи.

- *Общата грешка* показва каква част от всички обекти класификаторът е разпределил в грешните класове:

$$(9) \quad ERR = (FP+FN) / (TP+FP+FN+TN) = 1 - ACC$$

Може да се изчисли като общият брой грешно класифицирани обекти се раздели на всички случаи или ако от 1 се извади оценката *Accuracy*.

Нужно е да се вземат под внимание и метриците: *Precision*, *Recall*, *Specificity* и *F1-score*. Тези метрики са най-популярни и често използвани при оценка на класификатор:

- *Precision* - метриката показва каква част от обектите, класифицирани като положителни са в действителност положителни:

$$(10) \quad Precision = TP/(TP+FP)$$

- *Recall* или *True Positive Rate* (*TPR*) - дава информация за това каква част от положителния клас е била открита от класификатора:

$$(11) \quad Recall = TP/(TP+FN)$$

- *Specificity* или *True Negative Rate* (*TNR*) - показва каква част от отрицателния клас е била намерена от класификатора:

$$(12) \quad Specificity = TN/(TN+FP)$$

- *F1-score* - обобщава *Precision* и *Recall* в единствена стойност.

Стойността на този показател е максимална, когато *Precision* и *Recall* са равни:

$$(13) \quad F1-score = 2*(Recall*Precision)/(Recall+ Precision)$$

Анализът на резултатите е свързан с решения от страна на ръководителите на обучаващия процес, как да се стимулират обучаемите и как да се помага на обучаемите в **риск**: допълнителни упражнения и задачи, допълнителна индивидуална и екипна работа, обяснения стъпка по стъпка, работа със софтуер като инструмент за моделиране и т.н.

Приложение на метода за оценяване

Изследвания и анализи на метода за оценяване и прогнозиране са извършени по дисциплината „Искусствен интелект“ с прилагане на инструментите на *Orange system* за експериментирание и изводи [163]. Представяната дисциплина (с преподаватели акад. Иван Попчев и проф. д-р Даниела Орозова) е част от виртуалното образователно пространство. Тя е структурирана в четири модула: Искусствен интелект – характеристики и проблеми; Търсене на решение в пространството на състоянията; Представяне на знания; Интелигентно вземане на решения. В средата за електронно обучение по дисциплината има предоставени материали с обем 587 МВ текстово описание, примери и 14 линка с полезни връзки. В преподаването се използват средите Moodle и Microsoft Teams. Тази дисциплина е основа на курсовете: „Анализ и проектиране на бази от данни и знания“, „Управление на знания в компютърни системи“ и др.

Стъпка 1. За оценяване на студентите по дисциплината “Искусствен интелект” са определени оценъчни компоненти, относно теоретични знания, практически умения и компетентности с различно познавателно ниво. Дефинирани са 3 теста (3 x 5 = 15 точки), Контролна работа *quiz* (15 точки), Проект - 45 точки и Обобщена дискусия - 25 точки.

В **проекта**, всеки студент сам избира тема от основните направления на изкуствения интелект, като например: Ontologies engineering, Semantic Web, Knowledge representation, Computational intelligence, Robotics, Natural language processing, Machine Learning, Deep learning, Soft computing, Pattern recognition, Multi-agent systems, Artificial neural networks, Genetic algorithms, Knowledge based systems, Decision support systems, Business intelligence, Data Science, Fuzzy sets and systems, E-learning и т.н. Проектът задължително включва: състояние, тенденции на развитие, идентификация, анализ и оценка на избрания инструментариум за въздействие върху потенциалните рискове, нерешени проблеми, изводи, заключение и библиография.

Обобщената дискусия със студента е върху темата на проекта, риск мениджмънт, мониторинг, контрол и оценка на риск мениджмънта на потенциалните рискове и възможностите за решаване на нови задачи с нестандартни решения.

Използваният примерен модел за оценяване по дисциплината е представен на фиг.15. Този модел на оценяване може да бъде динамично променян и адаптиран към конкретната дисциплина. Например в дисциплината „**Управление на знания в компютърни системи**“ се прилага **алтернативен** модел, като компонентите, формиращи оценката са: Тест 1 (10 точки) , Тест 2 (10 точки), Тест 3 (10 точки), Тест 4 (10 точки), Самостоятелна работа (15 точки), Разработка на проект (20 точки), Изпит (25 точки).

| Оценъчен компонент | Означение | Оценъчна скала |
|-----------------------------|--------------------|----------------|
| Теоретични знания модул 2 | Тест_1 | до 5 точки |
| Теоретични знания модул 3 | Тест_2 | до 5 точки |
| Теоретични знания модул 4 | Тест_3 | до 5 точки |
| Практически компетентности | Контролна работа | до 15 точки |
| Практически знания и умения | Проект | до 45 точки |
| Теоретични знания и умения | Обобщена дискусия | до 25 точки |
| Крайна оценка | Окончателна оценка | оценка [2, 6] |

Фигура 15. Примерен модел за оценяване по дисциплината

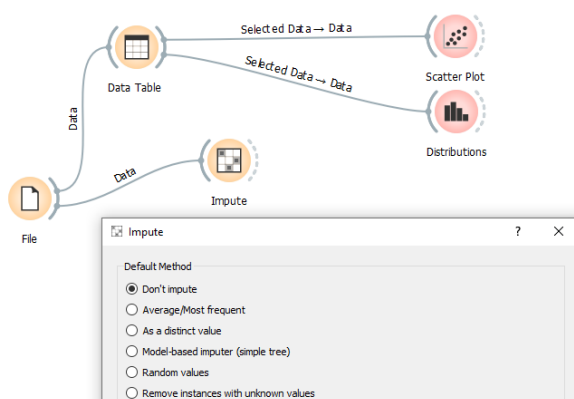
Стъпка 2. Провеждане на обучение и процес на оценяване. Задачата е свързана с намиране на общ подход за автоматизирано оценяване и прогнозиране на оценките на обучаемите. За намаляване субективността при оценяването на практическите умения се допуска оценяването да се извършва от външен оценител от фирми в областта на информационните технологии като: Технологика, Скейл Фокус и др. По всяка дисциплина студентите могат да получат до 100 точки и окончателната оценка се формира по следната скала: от 54 до 60 точки - *Среден* (3); от 61 до 70 точки - *Добър* (4); от 71 до 80 точки - *Много добър* (5); от 81 до 100 точки - *Отличен* (6).

Стъпка 3. Анализ на натрупаните данни от проведените обучения в реална среда.

В процеса на работа са проведени множество експерименти. Основната цел е да се решава класификационен проблем като се установи дали е възможно да се предвиди оценката (изходна променлива) с помощта на входните променливи (точките от отделните оценъчни компоненти), които се запазват в модела. За решаване на класификационния проблем се прилагат няколко различни техники, чрез инструментите на *Orange Data Mining System*.

В началото се създава работен процес и чрез инструмента „*File*” **зареждаме данните** за оценките на студентите по различните оценъчни компоненти в брой точки. Те могат да бъдат въведени от Excel (.xlsx), от текстов файл с раздели (.txt), файл с данни разделени със запетая (.csv) или URL адрес. За по-доброто разбиране, данните могат да се визуализират по някои колони или извадки от тях. Например, може да се свърже файла с данни с инструмента „*Scatter Plot*” и се изберат колоните, чиито стойности ще се изобразяват по осите X и Y, използвани цветове, форми, размери и други параметри. Друг популярен инструмент за визуализиране на данните е „*Distribution*”, чрез който може да се покаже разпределение в набора от данни по даден атрибут.

Следва **почистване на данните**. Могат да се приложат различни подходи: изтриване на липсващите стойности или да се заместят по подходящ начин. Чрез инструмент „*Impute*” (фиг. 16), се избира измежду различни методи за заместване. По подразбиране е опцията „*Remove the rows with missing values*”. Други възможни опции са: *Distinct Value, Random Values, Model-Based*.

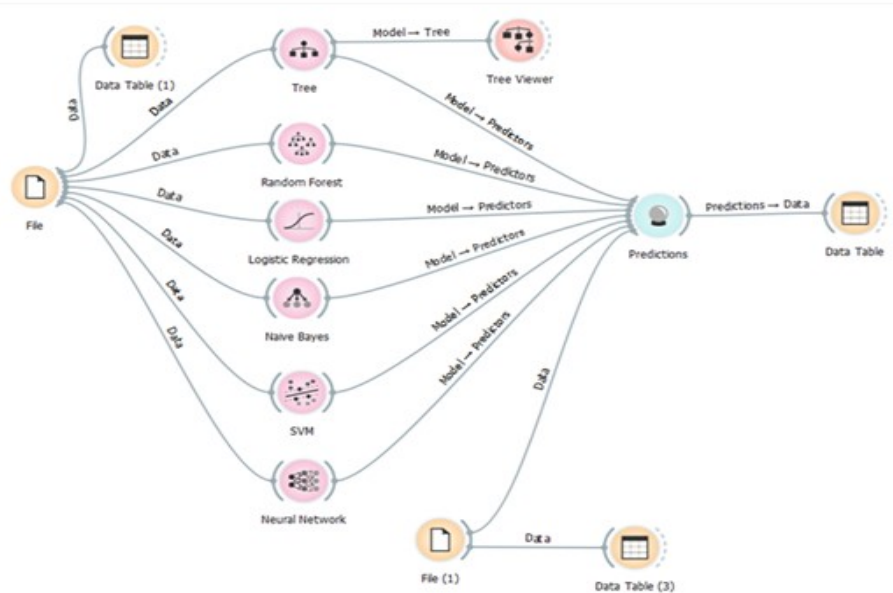


Фигура 16. Прилагане на инструменти за зареждане и почистване на данните

В зависимост от целта и вида на данните избираме конкретен инструмент за регресия или класификация и **задаваме целевата променлива**.

Стъпка 4. Създаване и трениране на модели за предвиждане.

В експеримента последователно се прилагат инструментите на системата: *Tree*, *Random Forest*, *Logistic Regression*, *Naïve Bayes*, *Support Vector Machines (SVM)* и *Neural Network*. Работния поток за създаване и трениране на моделите е показан на фиг. 17.



Фигура 17. Работен поток за трениране на моделите и предвиждане

Предвиждане на нови данни. Чрез обучените алгоритми се очаква при задаване на нова комбинация от стойности за избраните компоненти да се определя изходната променлива. На тази стъпка считаме, че моделът е готов за практическо приложение. Моделът придобива независимост и прави свои заключения въз основа на набори от данни и обучение. На фигура 17 е представен работния поток за предвиждане чрез инструмента *Prediction* на системата Orange. Новите данни се подават чрез файл *Test.xlsx*, който има същата структура като началната таблица с данни, но колоната с изходната променлива не е зададена. Инструментът *Predictions* извършва прогнозите за данните от файла *Test.xlsx* и определя предвижданата стойност на колоната.

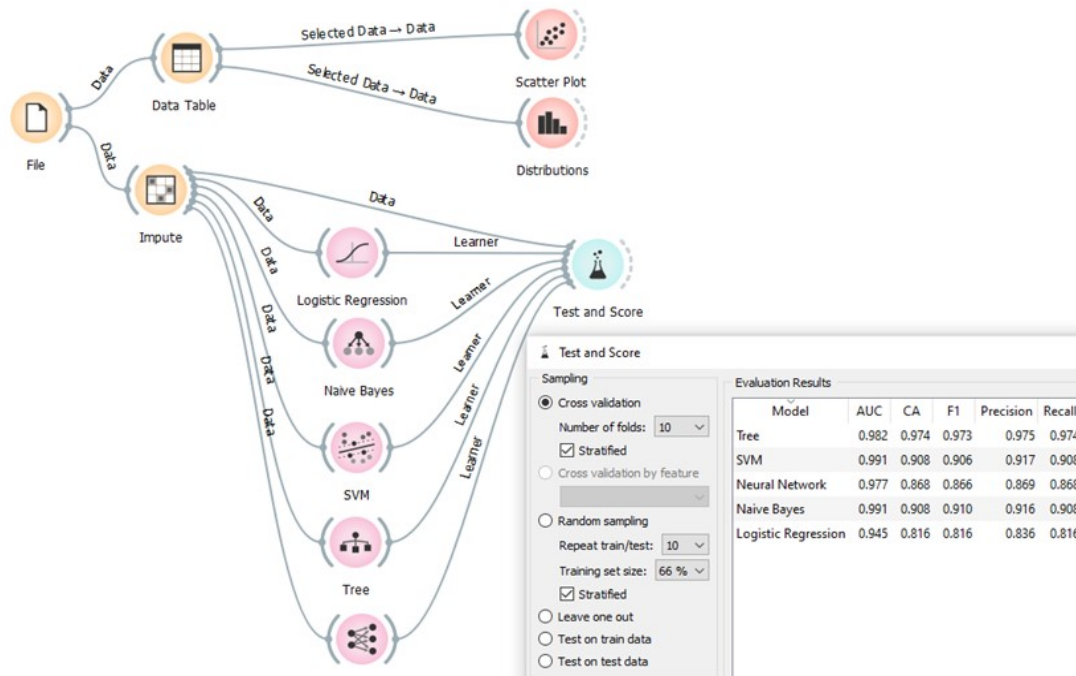
Полученият файл е с разширение (.ows) и може да бъде отворен с всяка версия на системата Orange. Общ вид на получавания резултат от предвиждането е показан на фигура 18. Това са резултатите от предвижданията за крайната оценка на обучаемите, получени от различните модели.

| Data & Predictions | | | | | |
|--------------------|--|--|--|--|----------------------|
| | Tree | Logistic Regression | Random Forest | SVM | Naive Bayes |
| 1 | 0.00 : 0.67 : 0.33 : 0.00 : 0.00 → Excellent | 0.00 : 0.49 : 0.02 : 0.06 : 0.43 → Excellent | 0.00 : 0.57 : 0.11 : 0.00 : 0.32 → Excellent | 0.01 : 0.18 : 0.03 : 0.01 : 0.77 → Very Good | 0.00 : 0.85 : 0.01 : |
| 2 | 0.25 : 0.00 : 0.50 : 0.25 : 0.00 → Good | 0.01 : 0.00 : 0.45 : 0.22 : 0.32 → Good | 0.20 : 0.10 : 0.20 : 0.50 : 0.00 → Middle | 0.22 : 0.02 : 0.23 : 0.51 : 0.03 → Middle | 0.39 : 0.00 : 0.04 : |
| 3 | 0.25 : 0.00 : 0.50 : 0.25 : 0.00 → Good | 0.00 : 0.00 : 0.20 : 0.30 : 0.50 → Very Good | 0.17 : 0.00 : 0.76 : 0.05 : 0.02 → Good | 0.03 : 0.05 : 0.42 : 0.08 : 0.42 → Very Good | 0.57 : 0.00 : 0.22 : |
| 4 | 0.00 : 0.67 : 0.33 : 0.00 : 0.00 → Excellent | 0.00 : 0.72 : 0.03 : 0.04 : 0.22 → Excellent | 0.00 : 0.60 : 0.20 : 0.00 : 0.19 → Excellent | 0.01 : 0.43 : 0.03 : 0.01 : 0.52 → Very Good | 0.00 : 0.25 : 0.04 : |
| 5 | 0.00 : 0.67 : 0.33 : 0.00 : 0.00 → Excellent | 0.00 : 0.03 : 0.31 : 0.37 : 0.29 → Middle | 0.00 : 0.20 : 0.44 : 0.02 : 0.33 → Good | 0.03 : 0.06 : 0.58 : 0.08 : 0.24 → Good | 0.01 : 0.04 : 0.86 : |
| 6 | 1.00 : 0.00 : 0.00 : 0.00 : 0.00 → Bad | 0.64 : 0.00 : 0.07 : 0.28 : 0.01 → Bad | 0.92 : 0.00 : 0.00 : 0.08 : 0.00 → Bad | 0.87 : 0.03 : 0.02 : 0.06 : 0.02 → Bad | 0.86 : 0.00 : 0.00 : |
| 7 | 0.00 : 0.00 : 0.00 : 0.00 : 1.00 → Very Good | 0.00 : 0.00 : 0.00 : 0.20 : 0.79 → Very Good | 0.10 : 0.15 : 0.00 : 0.00 : 0.75 → Very Good | 0.02 : 0.20 : 0.04 : 0.03 : 0.71 → Very Good | 0.01 : 0.60 : 0.00 : |
| 8 | 0.00 : 1.00 : 0.00 : 0.00 : 0.00 → Excellent | 0.00 : 0.80 : 0.01 : 0.01 : 0.18 → Excellent | 0.00 : 1.00 : 0.00 : 0.00 : 0.00 → Excellent | 0.02 : 0.87 : 0.03 : 0.03 : 0.06 → Excellent | 0.00 : 1.00 : 0.00 : |
| 9 | 1.00 : 0.00 : 0.00 : 0.00 : 0.00 → Bad | 0.11 : 0.00 : 0.01 : 0.08 : 0.79 → Very Good | 0.47 : 0.25 : 0.14 : 0.13 : 0.00 → Bad | 0.29 : 0.09 : 0.27 : 0.25 : 0.10 → Good | 0.46 : 0.16 : 0.33 : |

Фигура 18. Резултати от предвиждането чрез различни модели

Стъпка 5. Оценяване на работата на моделите, върху данните.

В работния поток се свързва всеки от създадените модели с “*Test and Score*” инструмент. След като се оценят моделите, трябва да се определи дали тяхната точност може да бъде подобрена, чрез настройка на параметрите, присъстващи в модела.



Фигура 19: Работен поток за оценяване с инструмент “*Test and Score*”

Резултат от работата на инструмента “*Test and Score*” е таблица с оценки за: *Accuracy*, *Precision*, *Recall* и *F1 Score* за създадените модели. Резултати относно метриците за оценка на качеството на предвиждане, за създадените моделите от проведения експеримент са показани на фиг. 20.

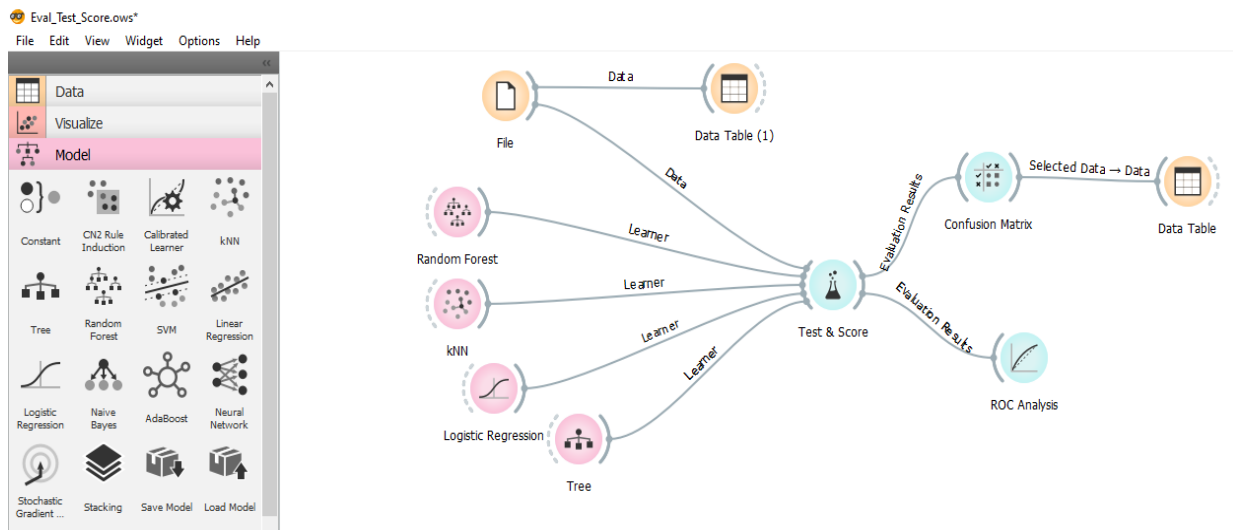
| Scores | | | | | |
|---------------------|-------|-------|-------|-----------|--------|
| Method | AUC | CA | F1 | Precision | Recall |
| Tree | 0.965 | 0.855 | 0.884 | 0.864 | 0.905 |
| Logistic Regression | 1.000 | 0.776 | 0.976 | 1.000 | 0.952 |
| Random Forest | 1.000 | 0.895 | 0.930 | 0.909 | 0.952 |
| SVM | 0.975 | 0.842 | 0.900 | 0.947 | 0.857 |
| Naive Bayes | 0.994 | 0.816 | 0.952 | 0.952 | 0.952 |
| Neural Network | 0.941 | 0.803 | 0.864 | 0.826 | 0.905 |

Фигура 20. Резултат от работата на инструмент *Test&Score* върху създадените модели

Разглеждайки прогнозната точност за всеки един от класовете оценки, може да се обобщи, че тя е най-лоша за класа оценки *Middle* (оценка Среден). Най-висока точност се постига за класовете *Bad* (Слаб) и *Excellent* (Отличен). Прогнозите за оценка *Good* (Добър) и *Very Good* (Много добър) при всички разгледани модели се представят с точност около 60-75%. Моделът на *Random Forest* е най-надежден, защото се представя с най-висока точност за всички класове оценки. Моделът *Naive Bayes* е с най-ниска оценка за точността спрямо другите модели при разглежданите данни.

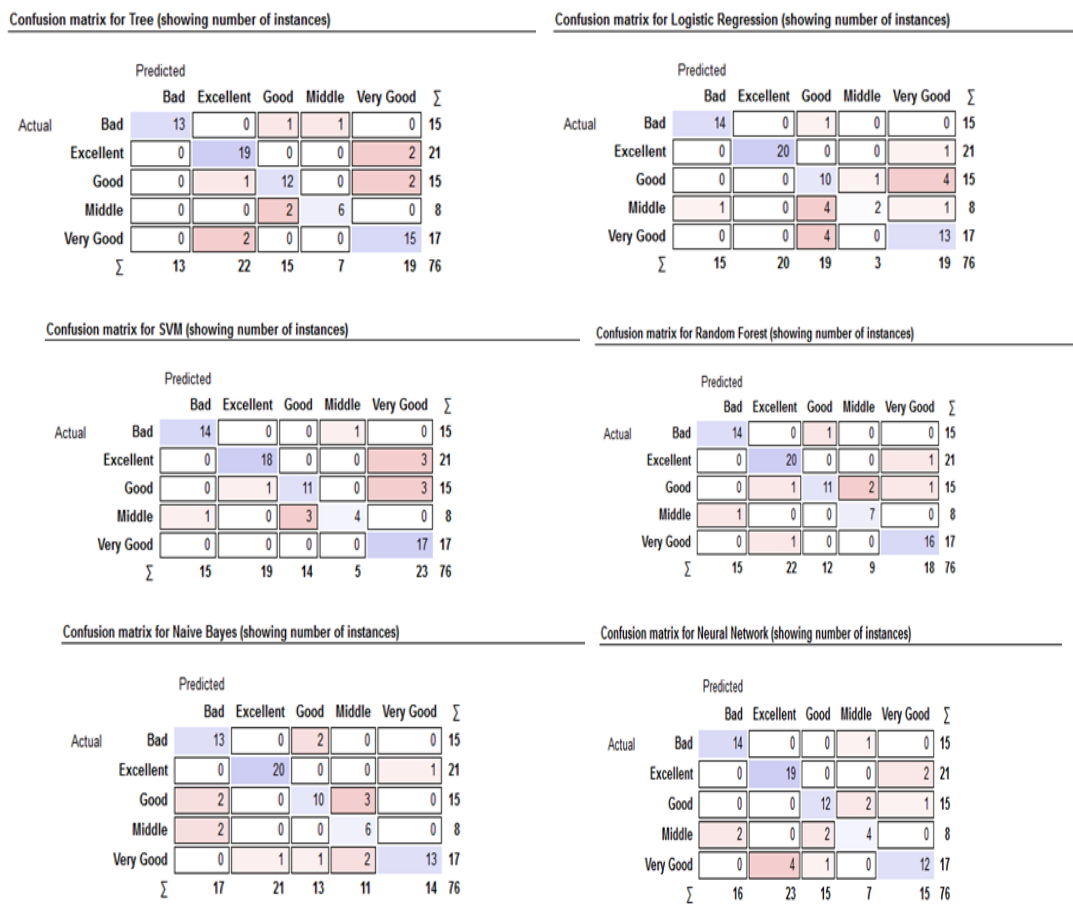
Тестване на различните модели и визуализация на точността на работата моделите може да се извърши чрез „*Confusion Matrix*”, *ROC Analysis* или друг

инструмент на системата. Работният поток е показан на фиг.21.



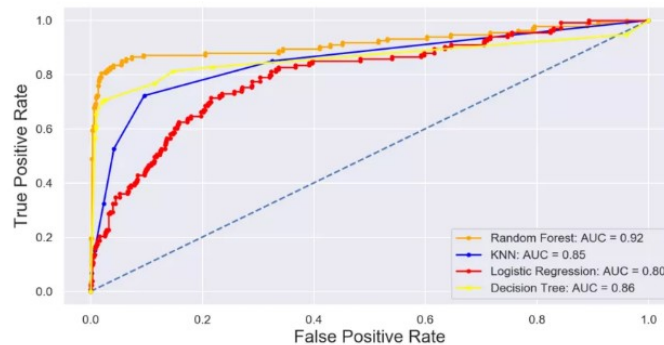
Фигура 21. Оценяване на работата на създадените модели

Резултати от работата на инструмента *Confusion matrix* са дадени на фиг. 22, за сравнение на ефективността на създадените модели.



Фигура 22. Резултати от работата на инструмента *Confusion matrix* върху създадените модели

Receiver Operating Characteristic (ROC) дава информация за това до каква степен моделът правилно разпознава съответните класове. Кривата съпоставя TPR (True Positive Rate) и FPR (False Positive Rate). Колкото по-близо до горния ляв ъгъл е ROC кривата, толкова по-високо е качеството на класификатора. На фиг. 23 е даден резултат от съпоставяне на четирите класификатора и графиката показва, че в конкретния случай за предвиждане, моделът, използващ алгоритъм *Random Forest* се справя най-добре.



Фигура 23. Примерен резултат от работата на инструмент *ROC Analysis*

Като се обобщят резултатите за средната оценка за прогнозната точност върху класовете оценки (Слаб, Среден, Добър, Много добър и Отличен) се получава: *Decision Tree* - 85.5%; *Logistic Regression* - 77.6%; *Random Forest* - 89.5%; *Support Vector Machine* - 84.2%; *Naive Bayes* - 81.6%; *Neural Network* 80.3.1%. Като подходящи за използване, при подобни задачи за оценяване, могат да бъдат класификационните алгоритми *Decision Tree* и *Random Forest*, които предвиждат с висока степен на точност елементите на *Bad* класа. Това са студентите със слаби оценки, които разглеждаме като „студенти в риск“.

В проведения експеримент инструментът обработва данните, които се събират по време на обучението на студентите в техните курсове. Освен това се използват и данни от анкетно проучване. Анкетата се изпраща по имейл през третата седмица от курса. В основата на анкетата са въпроси, които стимулират студентите да определят степента на удовлетвореност, както и въпроси, които определят мнението им за нивото на трудност на учебния предмет. Данните, получени от анкетите, добавят и редица нови характеристики, които са пряко свързани с отпадането на обучаемите, като липса на интерес, липса на време, организационни пречки по време на обучението и др. Първите експерименти са направени от малка извадка от данни, ограничена от първоначално налични реални данни за студенти. С помощта на разработената система за наблюдение и събиране на данни за обучаемите и тяхната дейност (решени задачи, преглеждане на електронни текстови и видео материали, участие във форум и/или групи по интереси и др.) от 4580 записа за обучаеми, 15 резултата от обучаемите са идентифицирани в риск. Това е многостъпкова процедура, която е пряко свързана с Big Data Analytics [208] в пространството за електронно обучение.

Ежедневно се събира огромно количество данни, което сега изисква нов подход към обработката на данни, прилагайки рамка за разпределена обработка с отворен код Hadoop, базирана на алгоритъм Map/Reduce. Hadoop [113] е разпределена и паралелна изчислителна парадигма изградена въз основа на MapReduce на Google [182]. Този подход е създаден да обработва големи набори от данни в кълъстерна среда при мащабен подход.

ИЗВОД: Проведените емпирични изследвания потвърждават, че предложеният **петстъпков метод за оценяване е перспективен** за разработване на система за ранно предупреждение за различни заинтересовани страни от обучаващия процес.

На базата на натрупани данни от работата на система за електронно обучение с различни потребители, прилагайки средства от областта на науката на данните, могат да се взимат различни решения относно обучението. Анализът на данните, натрупвани в курсовете за електронно обучение дава възможност да се промени модела на изпитване и да се проектират модули, които да отговарят на индивидуалните нужди на обучаемия, относно търсене на необходимата информация.

Подсилващо обучение (обучение с утвърждение)

Към алгоритмите за машинно обучение принадлежи и подсилващото обучение (*Reinforcement Learning*). Характерното за този тип алгоритми е, че имитира психологически модел, при който на системата се подават „награждаващи“ и „наказващи“ сигнали с цел да се максимизира вероятността за получаване на „награда“ и да се минимизират „наказанията“ [179]. Това обучение се прилага обикновено при липса на предварително зададен „правилен“ набор от обучителни данни. Такъв подход е различен от начина на действие при контролирано обучение, където целта е намаляване на отклонението, съобразно предварително зададени правилни данни (вход/изход).

Подсилващото обучение се основава на пробни действия, които постепенно водят до утвърждаване на добрите и избягване на лошите действия. Задейства се функция на кумулативно награждаване. Резултатът от обучението е оптимална стратегия за действие във всяка ситуация. Стратегията е оптимална, ако успява да максимизира сумата от всички награди, получени по време на изпълнението. Решенията се взимат на дискретни стъпки и всяко следващо решение зависи от текущото състояние на обкръжението.

Неконтролирано обучение

Алгоритмите от този тип приемат съвкупност от данни, съдържаща входни стойности и намират структура или разпределение на данните, без указания за известна променлива или функция за възнаграждане, данните не са етикетирани, няма примери за обучение [206]. Основни типове решавани задачи са: групиране (*Clustering*), намаляване на размерността (*Dimensionality reduction*), оценка на плътността (*Density estimation*).

Клъстерният анализ разпределя съвкупност от наблюдения в групи (*clusters*), така че наблюденията в един и същ клъстер да са сходни според един или няколко предварително зададени критерии, а наблюденията от различни клъстери да са различни [110]. Различните техники за клъстериране използват различни предположения за структурата на данните, които се дефинират чрез някаква метрика за сходство. Например, близост между членовете на един и същ клъстер, вътрешна компактност, разлика между клъстерите и др.

Разработените методи за групиране използват различен индукционен принцип. Farley and Raftery (1998) [83] предлагат разделянето на методите на групиране на две основни групи: йерархични и разделящи методи. Han and Kamber (2001) [105] предлагат категоризиране на методите в допълнителни три категории: базирани на плътност методи,

базиран на модели и мрежови методи. Алтернативна категоризация, основана на принципа на индукция на различни методи за групиране е представена в Estivill-Castro (2000) [78].

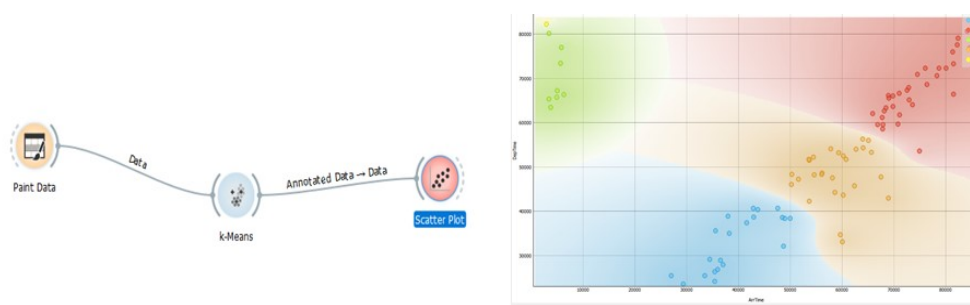
- Алгоритми за разделяне на клъстери

Тези алгоритми генерират различни дялове и след това ги оценяват по някои критерии. Те са наричани нейерархични, тъй като всеки екземпляр е поставен в точно един от k взаимно изключващи се клъстери. Изисква се предварително да се въведе желан брой клъстери.

Един от най-често използваните алгоритми за разделяне на клъстери е *k-means clustering* алгоритми. Този тип алгоритми принадлежат към *Exclusive Clustering*, защото данните са групирани, така че ако определена данни принадлежи към даден клъстер, тя не може да бъде включена в друг клъстер. Друг тип *Overlapping Clustering*, използва различни множества за групиране на данни, така че всяка точка може да принадлежи на два или повече клъстера с различна степен на членство.

Тристъпков алгоритъм: определя централната координата (центроид); определя разстоянието от всеки обект до центъра; групира обектите на базата на най-малкото разстояние. Прекратява се, когато центроидите спрат да се движат или някакъв праг е достигнат (например брой итерации). Следва примерен работен поток за прилагане на инструмент *k-means* върху набор от данни и визуализиране на резултата чрез инструмент *Scatter Plot* (фиг. 24).

За изграждане на информационния поток първо се извършва *събиране на данни*. Зарежда се работния плот и набора с инструменти и чрез инструмент „*File*” се зареждат данните за анализ. Може да бъде приложен и инструмент *Impute*” за почистване на данните. Следва *избор на инструмент k-means*. Този инструмент на системата *Orange* реализира *k-means clustering* алгоритъм. Последната стъпка е *визуализация на резултата*. Самият инструмент *k-means* не визуализира резултат, за тази цел трябва към информационния поток да бъде свързан инструмент за визуализация, например *Scatter Plot*.



Фигура 24. Работен поток с прилагане на инструменти *k-means* и *Scatter Plot*

- Йерархичната клъстериация

При йерархичните клъстеризационни процедури (*Hierarchical Cluster*) броя на клъстерите не е предварително определен. Тези алгоритми започват от деклариране на всяка точка за свой собствен клъстер и след това обединяват два най-подобни клъстера, докато не бъде удовлетворен критерий за спиране.

Голямото разнообразие на процедурите идва от прилаганата **метрика** между различните обекти. Примери за метрики за изчисляване на разстояние: Евклидово разстояние; Сума на абсолютните разлики (SAD или L1 Norm); Сума на квадратичните разлики (SSD или L2 Norm); Средна квадратна грешка (MSE) и др.

Йерархичната клъстеризация в *Orange* изчислява йерархично групиране на произволни типове обекти от матрицата на разстоянията между тях и показва съответната *дендрограма*. Започва като присвоява всеки елемент на клъстер, така при N елемента има N клъстери, всеки от които съдържа само един елемент. След това се намира най-близката (най-сходна) двойка клъстери и те се обединяват в един клъстер. Следва изчисляване на разстояния (прилики) между новия клъстер и всеки от старите клъстери. Стъпките се повтарят, докато всички елементи се групират в един клъстер с размер N .

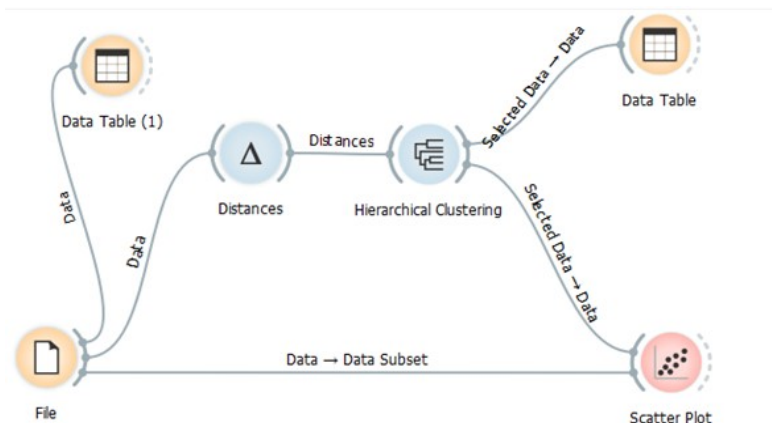
Метод за изграждане на информационен поток при йерархична клъстеризация е свързан с прилагане на следните *стъпки*:

1. **Събиране на данни.** Зарежда се работния плот и набора с инструменти и се зареждат данните за анализ. Може да бъде приложен и инструмент *Impute* за почистване на данните.

2. **Избор на инструмент *Distance*** за определяне на разстоянието между елементите. За групиране на данните могат да се използват различни показатели за изчисляване на разстояние. Мярка за разстоянието между точките, изобразени на евклидовата равнина е *Euclidean метрика*. *Manhattan метрика* е мярка, където разстоянието се изчислява като сума от абсолютната стойност на разликите между две точки, изобразени на декартовата координатна система. *Minkowski distance метрика* представлява обобщение на показателите за разстояние от предходните две метрики.

3. **Избор на инструмент *Hierarchical Clustering***. Този инструмент на системата *Orange* реализира алгоритъм за йерархичното групиране на произволни типове обекти чрез изчислените разстояния и показва съответната дендрограма. Терминът *дендрограмата* отразява граф-дърво, в което всеки възел представя стъпка от процеса на клъстеризиране и може да носи информация за разстоянието между двата клъстера.

4. **Визуализация на резултата.** Чрез различни инструменти може да се визуализира резултата по различни начини, например чрез инструментите *Data Table* и *Scatter Plot*.



Фигура 25. Работен поток при йерархично клъстеризиране

Други популярните софтуерни средствата за извличане на знания, които могат да бъдат използвани за анализ на данни са например:

WEKA (Waikato Environment for Knowledge Analysis) е разработена в Нова Зеландия, в университета на Уайкато. Софтуерът е написана на Java и е достъпен на адрес: <http://www.cs.waikato.ac.nz/ml/weka>, лицензиран под GNU General Public License. WEKA съдържа модули за визуализация на данни (49 инструмента), 76 алгоритъма за класификация и регресия, 8 алгоритъма за клъстериране и др. [213]. Не поддържа създаването на нови функции, но има поддръжка за автоматичен избор на функции. Потребителите могат да извикват алгоритмите за извличане на данни от командния ред, чрез графичен потребителски интерфейс (GUI) или чрез Java API. Интерфейсът на командния ред и API са по-мощни от GUI, защото се дава на потребителите достъп до всички разширени функции.

RapidMiner (<http://rapid-i.com/content/view/181/190/>) е пакет за извършване на анализи за извличане на данни и създаване на модели, написан на Java. Разполага с обширен набор от алгоритми за класификация и регресия, както и алгоритми за групиране, извличане на правила за асоцииране и други приложения. Графичният език за програмиране на RapidMiner е по-мощен от тези на повечето други инструменти за извличане на данни, като разполага с широка гама от налични инструменти за оценка на модела. Моделите могат да бъдат изведени в xml файлове, които могат да се използват за изпълнение на модела върху нови данни. RapidMiner включва всички налични алгоритми в Weka, освен това има широк набор от уроци, полезни при изучаване как да се използва графичния програмен език. RapidMiner се предлага безплатно за академична употреба, а търговските лицензи са достъпни чрез издателя Rapid-I. *RapidMiner* използва клиент/сървър модел, като сървъра се предлага от софтуер като услуга в облачни инфраструктури [176].

KNIME е пакет за почистване и анализ на данни, подобен на RapidMiner и Weka. Включва всички алгоритми на Weka, освен това предлага множество специализирани алгоритми в области като анализ на социалните мрежи. Мощен аспект е способността му да интегрира данни от множество източници (напр. .csv от инженерни функции, текстов документ с текстови отговори и база данни с демографски данни на обучаеми) в рамките на един и същи анализ. KNIME също предлага разширения, които му позволяват да взаимодейства с R, Python, Java и SQL. Графичен потребителски интерфейс позволява изграждане на информационен поток с включване на възли за предварителна обработка на данни. KNIME е софтуер написан на Java, като основната версия включва модули за интеграция на данни, методи за анализ на данни и визуализация [125].

KEEL (Knowledge Extraction based on Evolutionary Learning) е софтуерен инструмент с отворен код (GPLv3), разработен на Java, който може да се използва за голям брой различни задачи за извличане на данни (<http://sci2s.ugr.es/keel/>). KEEL разполага с обширен набор от алгоритми за класификация и регресия, с фокус върху еволюционните алгоритми. Но неговата поддръжка за други типове алгоритми за извличане на данни, като групиране и асоциативен анализ, е по-ограничена от другите пакети.

SPSS предлага набор от статистически инструменти и подобно на **Excel** е широко популярен. SPSS се допълва от IBM SPSS Modeler Premium пакет за анализ и извличане на данни, който интегрира пакети за анализ и извличане на текст. SPSS Modeler има функционалност за създаване на нови функции от съществуващи, инструменти за трансформиране на данни и намаляване на пространството на функции.

Spark MLlib. Spark (<http://spark.apache.org/mllib/>) е рамка за обработка на данни в множество компютърни процесори по разпределен начин. Spark може да се свърже с няколко езика за програмиране, включително Java, Python и SQL, чрез API, което позволява тези езици да се използват за разпределена обработка. Рамката за машинно обучение MLlib на Spark предоставя реализации на няколко стандартни алгоритми за машинно обучение и извличане на данни. Въпреки че функционалността на MLlib е ограничена и той е чисто програмен инструмент, неговият разпределен характер го прави ефективен избор.

3.3. Уеб метрики за оценяване дейностите на обучаемите във виртуалното образователно пространство

Анализът на веб показателите на достъпното веб съдържание може да се използва за намиране на модели в поведението на потребителите. Анализът на данните, натрупани по време на обучението позволява проучвания и обратна връзка за това как обучаемите търсят информация, какви затруднения срещат и проектиране на модули, които отговарят на индивидуалните нужди на обучавания.

На базата на веб метрики тук се предлага метод за оценка на степента на използване на веб сайтове, свързани с областта на обучението при моделиране на обучаемите, базирайки се на анализ на поведението им в обучаващата среда и веб пространството. За оценяване на посещаемостта на веб страниците се реализира система от правила, като се прилага метода на индуктивна размита класификация. Анализът на данните дава възможност да се проектират модули, които да отговарят на индивидуалните нужди на обучаемия, относно търсене и използване на информация.

Веб анализът е измерване, събиране, анализ и отчитане на интернет данни с цел по-добро разбиране на сложните взаимодействия между потребителите на уебсайтове. Процесът на веб анализ включва различни веб метрики, дефинирани в Google Analytics [65] като:

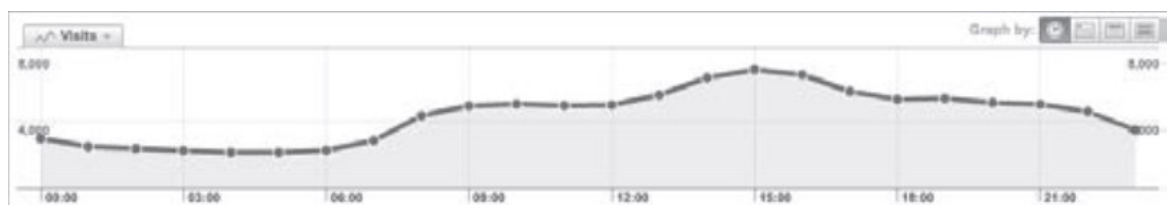
- **page views** - броя показвания за преглеждане на веб страницата, достъпна от посетител (без паяци или роботи);
- **visitors** - броя на уникалните посетители на веб сайта;
- **pages/visits** - броя на страниците, които са разгледани от посетител по време на посещение;
- **time on site** - продължителност на времето, прекарано от всички посетители на веб сайта;
- **stickiness** - способността на веб страницата да задържи посетителя на веб сайта;
- **frequency** – (честота) броят посещения, направени от посетител на сайта (индикатор за лоялност);
- **recency** - брой дни, изминали от последното посещение на посетителя на сайта;
- **length of visit** - времето на посещение, прекарано от посетител на уебсайта (в секунди);
- **depth of visit** - брой страници, посетени от посетител за едно посещение и др.

В областта на веб анализите се използват главно две техники за анализ на трафика на веб сайта: събиране на данни от страна на сървъра и от клиентска страна. Методите за събиране на данни от сървъра извличат и анализират данни основно от регистрационните файлове (*Log files*). Регистрационните файлове включват стандартна информация за регистрационни файлове като IP адрес, клеймо за час и дата, тип браузър и операционна система. Информацията за поведението включва обща информация за сърфиране, като например брой прегледани страници, езикова настройка и настройки за разделителна способност на екрана в браузъра. Когато се събират данни от клиентски сайтове или страници с отметки, данните за посетителите на страницата се изпращат до проследяващ сървър с помощта на JavaScript код (или етикет), вмъкнат в HTML страницата. С този подход могат да се проследяват точно всички действия на посетителя, както и да се събира допълнителна информация. Бисквитите могат да се използват, за да се определи колко

посетители за първи път или колко повторни посетители е получил даден сайт, колко пъти посетител се връща за всеки период и колко време минава между посещенията.

Google, WebTrends, Nedstat [23] и други компании предоставят софтуер за уеб анализ с помощта на маркиране на страници. Google Analytics е най-използваната безплатна програма [65]. От гледна точка на анализа на дейността на обучаемите, особено интересна е свързаната с клиентите информация, историята на посещенията и поведението им, потребителския профил и др. Данните от системите за управление на взаимоотношенията с клиенти (CRM) също се съхраняват в оперативни бази от данни. За да се предостави цялостен поглед върху онлайн клиентите, данните от различни източници трябва да бъдат интегрирани и анализирани. Google Analytics изготвя анонимни и статистически отчети за уебсайтовете, които го използват. Отчетите включват различни информационни изгледи и показват данни като географско местоположение (на базата на общи кодове за геолокация, базирани на IP), време на посещение и т.н. Тези доклади са анонимни и статистически.

Например: отчетът на фиг. 26 [65, глава 4] дава възможност да се проследи по часове посетителския трафик на сайта.



Фигура 26. График за почасова тенденция с Google Analytics.

За да се направи анализ на най-посещаваните от обучаемия уеб сайтове, дълбочината и честотата на посещенията, се използва *Fuzzy Classification of the Web Metric* [220]. Известно е, че в теорията за размитите множества и размитата логика, се отчитат неточност, несигурност и неяснотата на човешкото мислене и език, определяйки функцията на принадлежност.

Размито множество A в X се дефинира като [217]:

$$(14) \quad A = \{(x, \mu_A(x))\}, \text{ където } x \in X, \mu_A : X \rightarrow [0, 1]$$

е функцията на принадлежност на A и $\mu_A(x) \in [0, 1]$ е степента на принадлежност на елемента x в множество A . По този начин един елемент може да принадлежи (или не) към размития набор със степен на членство в интервал $[0, 1]$, вместо да принадлежи изцяло (стойност 1) или не принадлежи (стойност 0).

В нашите експерименти за броя на посещенията на даден уеб сайт, се въвеждат термините „low“, „medium“ и „high“ като езикови променливи. В общия случай, отчитайки стойностите за показване на страниците като true (1) или false (0) за месец, страниците се определят като „ниско посещавани“, ако посещенията са например между 0 и 25, между 25 до 55 посещения страниците са „средно посещавани“ и повече от 56 посещения са класифицирани като „високо показвани“. Въпреки това, ако посетител има 55 посещения на страница, той се определя в клас „средно“ активен, докато друг посетител с 58 посещения се определя като „много“ активен посетител. Въпреки разлика

само от 3 посещения, те са разпределени в две различни групи. Чрез дефиниране на размити множества, представени с функция за принадлежност се въвежда непрекъснат преход между класовете „low“, „medium“ и „high“. Така посетител може да принадлежи отчасти на два класа (55% на „високо активен“ и 45% на „средно активен“) едновременно. Използването на размити класове позволява по-прецизна класификация на уеб метричните стойности и рисковете от погрешните класификации могат да бъдат намалени [134].

Отчитайки това с цел а да се оцени трафика на наблюдавани уеб-ресурси, се разработва размита система от правила [77]. Измерването на броя на посещения на страница е число, което само по себе си няма голямо значение. Само контекстът на числото спрямо броя на прегледите на други страници предоставя знание, на базата на което можем да направим оценка. За определяне на броя на посещенията на уеб страници се реализират правила за размита класификация. Прилага се метода на **индуктивна размита класификация** (*Inductive Fuzzy Classification IFC*) [122], при който групирането на елементите в размит набор, се извършва с функция за членство, извеждана чрез индукция от данни - Индуктивната размита класификация по процентилен ранг (IFC-PR) [122] генерира размита функция на членство, в случая използвайки общите езикови термини „low“, „medium“ и „high“, корелиращи с броя потребителски посещения:

- емпиричният ранг на стойността x с метриката M определя принадлежността към размития клас „**high**“:

$$\mu_{\text{high}}(x) := P(M < x)$$

- емпиричният ранг на стойността x с метриката M определя като „**low**“ (отрицанието на принадлежността към класа „high“):

$$\mu_{\text{low}}(x) := 1 - \mu_{\text{high}}(x)$$

- класификация „**medium**“ се дефинира като:

$$\mu_{\text{medium}}(x) := 1 - \text{abs}(\mu_{\text{high}}(x) - 0.5) - \text{abs}(\mu_{\text{low}}(x) - 0.5)$$

Предимство при използване на методи на размита логика, е лесното интерпретиране на създадените функции за членство. Резултатите са по-разбираеми и подходящи за взимане на решения.

В провежданото изследване [168] се прави анализ на посещенията на уеб сайтовете, указани като помощни материали в курса по дисциплината Изкуствен интелект. За всяка уеб страницата се оценява посещаемостта, като например: страницата $W1$ има 135 посещения в рамките на обучаващия курс. Общо предложените и наблюдавани страници в курса са 80 и 56 от тях имат по-малко посещения от $W1$. За оценка на посещенията на $W1$, се пресмята разпределението:

$$\mu_{\text{high}}(\text{посещение } (W1)) :=$$

$$P(\text{Брой посещения} < \text{посещения } (W1)) = 56/80 = 0,7.$$

Така според размитата класификация се определя, посещаемост на страницата „висока“ над 0,7, за „ниска“ под 0,3, а за „средна“ до 0,6.

На базата на получения анализ за всяка уеб страница, предлагана в обучаващия курс, се взимат решения за актуализиране на уеб източниците, както и допълване на литературата с нови сайтове, свързани с най-търсените теми.

Представените (в предходната точка 3.1.2.2) алгоритми под формата на многостъпков процес за разработване на информационен поток с инструментите за машинно обучение на системата *Orange* тук се предлагат като веб услуги, които могат да се използват за изследване, анализ и обучение в пространството. Анализът на достъпното веб съдържание, по време на обучаващия процес може да послужи за откриване на модели в поведението на обучаемите, базирани на данните от потоци кликания и свързани данни, събрани в резултат на взаимодействието с веб ресурсите.

След като се направи анализ на най-посещаваните от обучаемия веб-сайтове, дълбочината и честотата на посещенията, следва да се анализира и определи типа на тяхното съдържание. Подход за автоматизирано определяне типа на документ и доколко е свързан с областта на обучението е представен в [166]. Анализът се базира на идеята: определени думи в документа са значими за съдържанието му. За да се определят тези думи се търси честотата на появата им, използвайки софтуера *Wordcloud* [215]. Могат се налагат филтри, относно получавания списък с думи. Изчислява се близост на всяка дума от полученото множество, с всички думи от речника на предметната онтология, свързана с областта на обучение и се взима най-малката получена стойност за всяка дума. За определяне на близост на думите и фразите могат да се приложат различни подходи, в разработеното приложение се използва подход, представен в [119] чрез q-gram метрики.

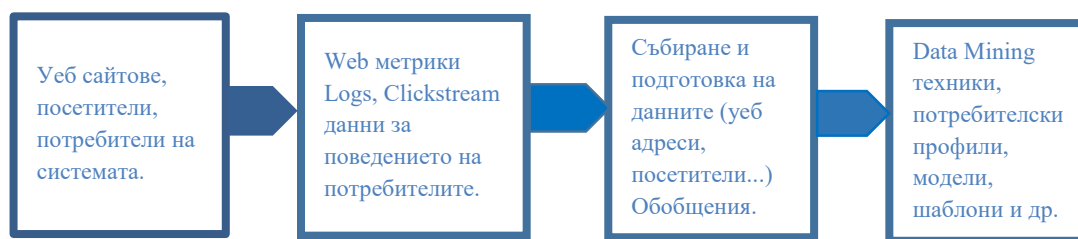
Чрез обработка на голямо множество от веб-документи се натрупва набор от данни с извлечени от тях централни думи и степента им на близост с понятията от анализираната област. Тези данни са използвани за обучение на класификационни алгоритми, представени и описани под формата на многостъпков процес, като целевия атрибут е дали документа е свързан с областта или не. Предложен е метод за анализ и прогнозиране на документи от разглеждана област на обучение, показан със схема на фиг. 27.



Фигура 27. Многостъпков процес на анализ и прогнозиране на документи от зададена предметна област

Използването на онтологии за представяне на знанията в отделните предметни области осигурява нужната независимост и гъвкавост, а тяхното описание със стандартизирани средства гарантира интелигентност при обработката на семантичната информация и прави създадените ресурси споделяеми.

От събираната информация за дейността на потребителите в веб пространството с помощта на клъстерен анализ или прилагане на асоциативен анализ, може да се извършва потребителско сегментиране и дори анализ на настроенята на потребителите. Веб съдържанието може да бъде анализирано на следните стъпки (фиг. 28).



Фигура 28. Архитектура за извличане на данни от уеб метрики

ИЗВОД: В етапа на анализ на резултатите от проведеното обучение, особено внимание се отделя на трудностите, които обучаемите срещат при изпълнение на отделните модули в дисциплината. В проведения експеримент, направените изводи са свързани с отчитане на факта, че студентите в дисциплина Изкуствен интелект срещат трудности при работа с литературните източници за изготвяне на проекта по избрана тема от *Модул 4*, в областите: „*Blockchain*“, „*Ethics and Emerging Sciences*“, „*Policy and investment recommendations for trustworthy Artificial Intelligence*“ както и при определяне на потенциалните рискове, свързани с тези технологии.

Чрез представения метод може да се проследи активността на обучаемите в интернет пространството, като се натрупват данни, относно тяхната активност в обучаващата среда. Основните метрики, които този тип анализ оценява, дават информация за посещаемостта на учебни материали и съответните уеб страници, колко често обучаемите посещават, кога и колко дълго преглеждат материалите, колко време прекарват в тяхното разглеждане, публикуват и отговарят във форуми. Анализът на този тип данни дава навременна информация за активността и успеха на обучаемите. Резултатите показват, че формулираните решения могат успешно да се използват за различни задачи и могат да бъдат адаптирани към нови технологии и приложения. Предложеният метод насърчава създаването на иновативна обучаваща среда като следващата стъпка в дигитализацията на образованието.

Различни методи за обучение и подходи за оценяване на знанията, уменията и компетентностите на обучаемите могат да бъдат реализирани във виртуално образователно пространство. Основните предизвикателства са свързани с детайлизирането на процесите и формализацията на модела.

3.4. Моделиране на потребителя в образователното пространство

С цел персонализиране на системата към знанията и уменията на обучаемия и прилагане на адекватни действия по време на обучаващия процес се създава и поддържа негов модел,

Моделът на потребителя се изгражда като структури от данни, представляваща индивидуални характеристики за потребителя, на дадена програмна среда. Моделът на потребителя в обучаваща среда има за задача да представи основно познавателните процеси и разбираня на обучаемия от областта на обучение. Целият процес на създаване, поддържане и обновяване на модела, въз основа на наблюдаваното поведение, се определя като диагностициране [28]. В процеса на проектиране на модела на потребителя трябва да се изяснят редица въпроси като:

- Каква е целта на моделиране на потребителя на съответната среда, на кои модули на системата са необходими данните, събирани и съхранявани в модела на потребителя.
- Кой е обекта на моделиране – какви и колко са потребителите, които се моделират, индивидуални ли са моделите или се прави модел на типове потребители или базов модел на потребител.
- Какво се моделира – какви са основните характеристики на потребителя, които ще се моделират и как се прилага модела. Често моделирани аспекти на потребителя са: ниво на знания, убеждения, грешни предположения, експертност и др.
- Каква е методологията за моделиране и кои са източниците на данни за модела. Използват се общи методи като: абстракция, класификация, прагматичен модел. Информацията в модела може да се получава от отговори на потребителя на въпроси или да се извежда индиректно от дейностите му. Информация може да се получава от предишни сесии на системата или от други източници.

”Моделът на потребителя съдържа в явен вид информация за основни и специфични характеристики на потребителя, свързани с взаимодействието му със системата” [37]. Обикновено в интелигентните системи се изгражда отделен модул, който има задачата са строи модела на потребителя и да предоставя на другите модули на системата данни за неговото състояние.

Данните, събирани и съхранявани в модела не са статични. Те се променят по време на действията на потребителя. Този модел съществува и има смисъл само, ако информацията от него се използва активно от системата за генериране на различни действия, относно различните потребители.

Могат да се прилагат различни техники и средства за следене и отчитане на дейността на потребителя и подходи за моделирането му. Моделирането на потребителя се разглежда в два основни аспекта:

- **Моделиране на знанията и уменията на клас от потребители.**

Изграждат се групи потребители с общи характеристики, на които се предлагат общи функционалности и ресурси. Целта на системата е да задоволява техните общи потребности. Базира се на стереотипният подход, при който потребител се класифицира

към някоя предварително дефинирана група и по този начин му се приписват общите за групата характеристики. Стереотипите могат да се представят във вид на йерархия и могат да разширяват тези, които са над тях. Стереотипните модели не са подходящи в случаите, когато се изисква индивидуална адаптация или предоставяне на специфична помощ.

- **Моделиране на знанията и уменията на отделен потребител**

Индивидуалният модел на потребителя служи за съхраняване на историята на дейностите, за фиксиране на текущото състояние на знанията, за прогнозиране развитието на уменията и компетентностите на потребителя и при откриване на грешни знания или заключения.

Barr и Feigenbaum [37] дефинират два **базови подхода** за моделиране на потребител:

- *Моделиране с припокриване*

При моделите с припокриване (overlay approach) знанието на потребителя се разглежда като подмножество на знанието на системата за областта. Знанието на системата може да бъде определено от предметното знание за областта, която се изучава, знанието на експерта в областта или очакваното знание за обучаемия. Знанието на системата е декомпозирано на независими компоненти и е покрито със система от означения, отразяващи нивото на усвояване на отделните компоненти. Обикновено тези компоненти са описани с йерархия или онтология, свързани пряко с понятията от областта. За оценяване на знанията се използват логически или числови стойности. Диференциалните модели са разновидност на този вид моделиране, при което системата поддържа множество от знания, които трябва да бъдат усвоени и друго множество от знания, които не е задължително да бъдат усвоени. Моделите с припокриване могат да се прилагат за оценяване на знанието на потребител в различни предметни области, което ги прави гъвкави и адаптивни.

Приемайки това становище тук се изгражда модел за оценяване на базовите знания на обучаемите относно основните понятия и зависимости между тях. В преподаваната предметна област е изградена йерархия на понятията, дефинирана в използваната домейн-онтология. След всеки тест на понятията се приписва относителна числова стойност (в проценти), която показва степента на сигурност на системата относно познаването на това понятие от конкретния обучаем. Оценката за всяко понятие може да се формира динамично като осреднена оценка на дъщерните му понятия по формулата:

$$(15) \quad \text{Mark_term} = \frac{1}{k} \sum_{i=1}^k \text{Mark_Subterm}[i],$$

където $\text{Mark_Subterm}[i]$ е оценката на i -тото дъщерно понятие, а k е броят на дъщерните (наследените) понятия на оценяваното основно понятие. Всяко дъщерно понятие от своя страна може да се разглежда като родителско на своите дъщерни понятия и да получава оценката си по същата формула и т.н. Тогава общата оценка за обучаваната тема може да се формира по формулата:

$$(16) \quad \text{Mark} = \frac{1}{n} \sum_{j=1}^n \text{Mark_Term}[j],$$

където $\text{Mark_Term}[j]$ е оценката на j -тото основно понятие в изучаваната тема от предметната област, а n е броя на тези основни понятия.

- *Моделиране с библиотека от грешки*

При този подход (bugs theory) предварително се дефинират типично грешно усвоявани или липсващи елементи от знания. Така моделът на потребителя се състои от модел на експерта от една страна и списък от типично допускани грешки в областта.

Интересен подход за събиране и анализ на информация за настроеността на обучаемите в процеса на обучение е представен в [194]. Събраните данни се представят в структури - „карти на креативност“. Създава се концепция за анализиране на картите на креативността.

Персонализацията на една системата позволява да се осъществява адаптация на създаваните към потребителя въздействия във функции на обяснения по степен на тяхната детайлизираност, насоченост, честота на включване и др., което се определя от конкретните характеристики на когнитивния стил на потребителя.

Източниците за получаване на данни, необходима за изграждане на модела на потребителя могат да бъдат: от диалога на системата с потребителя; от поведението на потребителя в конкретната ситуация; от вътрешните връзки между обектите и познанията в предметната област (структурни); базиращи се на оценки за средното ниво на подготвеност на потребителя (фонови).

Голяма част от тези данни се събират от средата за електронно обучение и се съхраняват в релационна база от данни.

Базата от данни на средата за обучение Мудъл е релационна база от данни, която съдържа над 200 таблици [135]. Таблиците са групирани на база функционалности, като името на всяка таблица съдържа името на модула, който обслужва.

Използвайки натрупаните данни, се генерират справки за участниците и работата им като: справка за продължителност на конкретно обучение, дата на започване и на завършване на участието; справка за участниците в дадена дейност, данни за обучаемите и курсовете, в които участват; общ брой участници; брой курсове по специалности; брой участници, преминали обучение в отделните курсове и др. В Мудъл не се предоставят стандартни средства за изпълнение на такива справки. За тяхното изготвяне е необходимо познаване на структурата и на съдържанието на таблиците с данни за потребителите. Примерните SQL кодове са създадени за MS SQLServer, но могат да бъдат прилагани и на други системи за управление на бази от данни. Основни таблици от базата от данни на Мудъл са следните пет:

- Таблица **role** съдържа информация за потребителските роли в системата. Ролите се определят от администратора на системата и имена им могат да се различават при различните инсталации на Мудъл.

- Таблица **user** описва потребителите на системата, с техните основни характеристики: име (Fname), фамилия (Lname), адрес (Address), e-mail и др.

За допълнителни характеристики относно потребителите, може да се създаде

таблица, свързвана с таблицата *user* по номера на потребителя, например *user_info*. Всеки отделен запис от таблицата съдържа информация за това, кое допълнително поле (*ID_filed*), за кой потребител (*ID_user*), каква стойност има (*Data*).

- Таблица *course* съдържа данните за курсовете като кратко и пълно име на курса (*Shortname* и *Fullname*) и неговите характеристики. В допълнение всеки курс има зададено категория (*Category*), което изпълнява ролята на външен ключ и служи за връзка с първичния ключ на допълнителна таблицата *course_categories*. В тази таблица са описани като категории основните направления и специалности.

- Таблица *context* - задава различни контекстно зависими характеристики. В Модъл има дефинирани като константи няколко нива на контекст (*context_level*, описани във файла *lib/accesslib.php*). В зависимост от стойността на *context_level*, полето *instanceid* се свързва с различни таблици. Например, ако *context_level* е 50, то *instanceid* съдържа ID на курс от таблицата *course*.

- Таблица *role_assignments* описва кой потребител (*ID_user*) каква роля (*ID_role*) в какъв контекст (*ID_context*) притежава. Например, един потребител може да е с роля студент за даден курс и с роля автор за друг.

Използвайки тези таблици, примерна заявка за извеждане на списък с имената на курсовете и специалността, в която се чете курса има вида:

```
SELECT course_categories.Name, course.Fullname, course.Shortname
FROM course_categories LEFT JOIN course
ON course_categories.ID=course.Category
```

Пример на заявка за групиране на курсовете по специалности и извеждане на броя на създадените курсове за всяка специалност има вида:

```
SELECT course_categories.Name, COUNT(*) AS num_course
FROM course, course_categories
WHERE course_categories.ID=course.Category
GROUP BY course_categories.Name
ORDER BY num_course
```

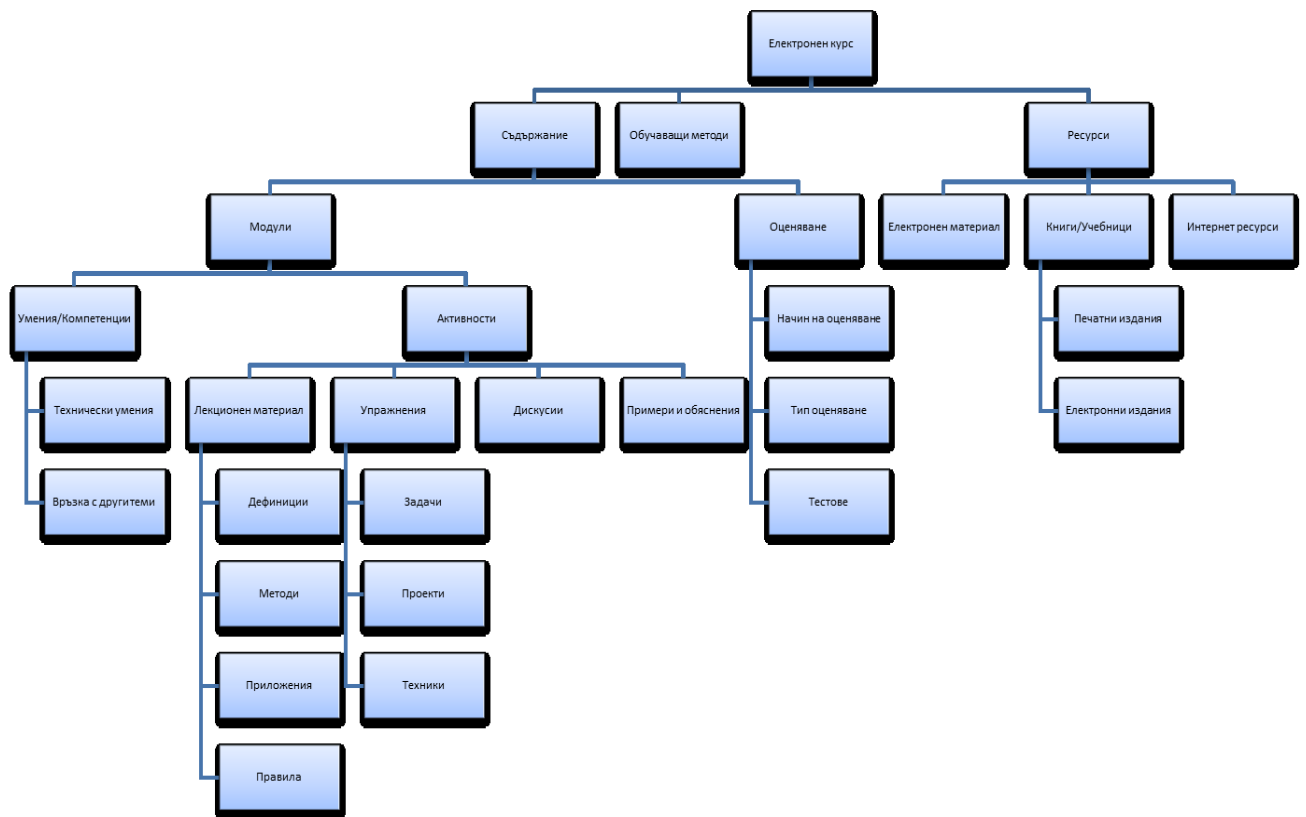
Алтернативна SQL-заявка изпълняваща същото условие има вида:

```
SELECT course_categories.Name,
(SELECT COUNT(*))
FROM course
where course_categories.ID=course.Category) AS num_course
FROM course_categories
ORDER BY num_course
```

Целта е автоматизирано изследване и оценяване на качеството на провежданите електронни и дистанционни форми на обучение от справки за индикатори, извлечени от базата от данни на използваната система за електронно обучение.

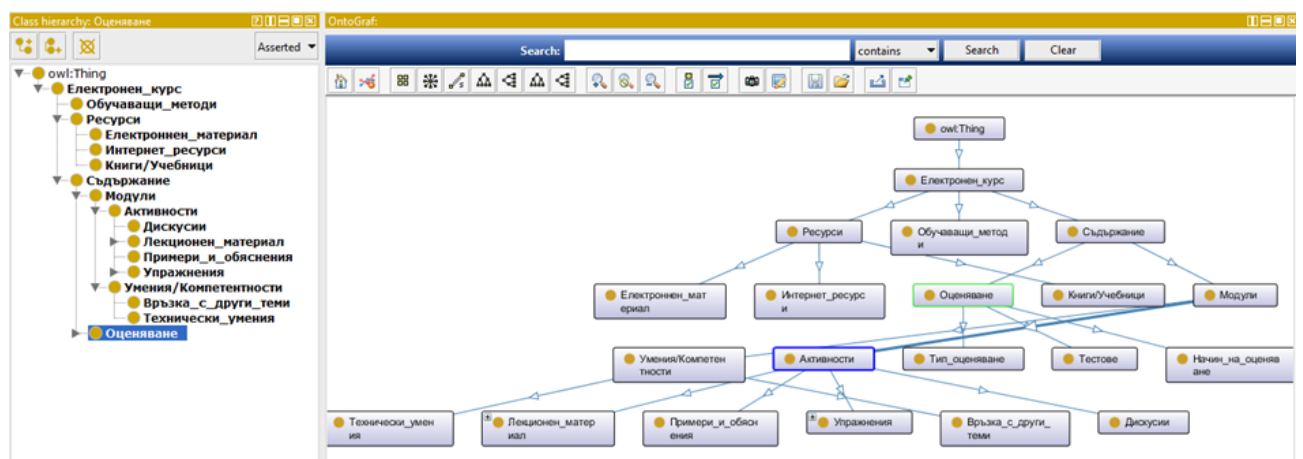
За да може обучението да бъде формално представено на мета-ниво, то неговите базови аспекти (знание, процеси и участници) трябва да бъдат моделирани и формализирани по начин, който осигурява дефиниране, намиране и използване на релации

между тях. В дисертационния труд се разработва и предлага модел на учебния процес в образователното пространство, базиран на онтологии за представяне на знанията. На фигура 29 е представена схема на разработвания онтологичен модел на електронен курс.

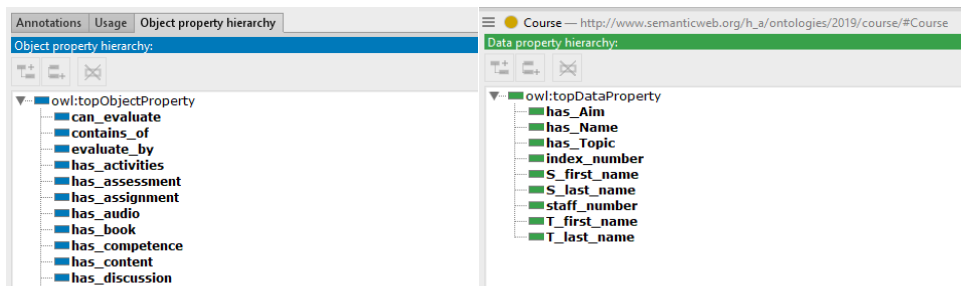


Фиг. 29. Онтологичен модел на електронен курс.

Прототип на онтологията е разработена и тествана чрез системата Protege. На фигура 30 е представена йерархията от класове, а на фигура - 31 част от *object properties* и *data properties* на разработения прототип на онтологията за домейна.



Фиг. 30. Онтологичен модел на електронен курс чрез системата Protege



Фиг. 31. Онтологичен модел чрез системата Protege - object и data properties.

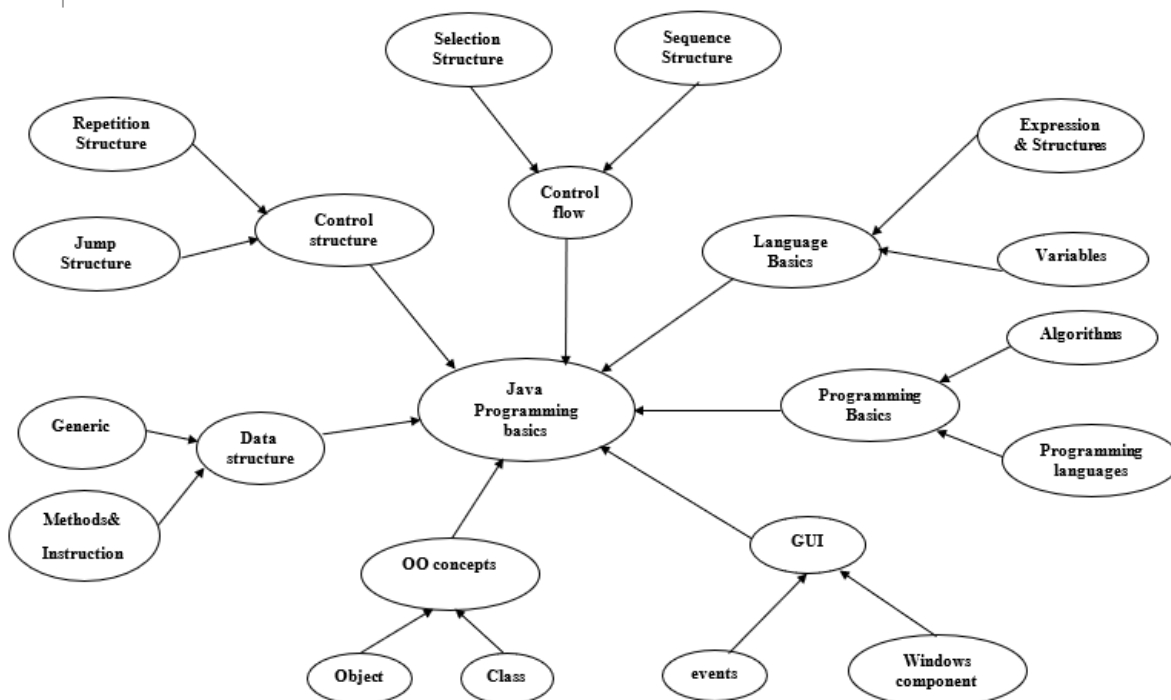
Могат да се използват различни формализми за представяне на знанието за преподаваната областта като: логика (предикатна, размита), онтология и др.

Тук се използва онтология, защото предлага общ речник, свързани понятия, техните свойства и връзки. Предимства на използване на онтологии за представяне на знанията от преподаваната област са:

- знанията са достъпни и разбираеми за софтуерните агенти;
- възможност за повторно използване на знанията;
- възможност за комбиниране на различни източници на знания.

Онтологията дефинира логическата структура на учебния материал и дава знания за йерархични и навигационни връзки между елементите.

Моделът на знанията от областта на обучение представя основните концепции и техните връзки, които трябва да бъдат преподавани и проследявани като ниво на усвоеност. Основните концепции за дисциплината „Основи на Java програмиране“ са представени на фигура 32.



Фиг. 32. Основни концепции от област на обучение „Основи на Java програмиране“

На базата на проведени експерименти с обучаеми в електронна среда по различни дисциплини тук се предлага обща рамка на архитектура на обучаваща система с модел на обучаемия, който включва три типа фактори: фактори на компетентност, емоционални фактори, фактори на въздействие на социалната среда. Чрез предлагания модел на обучаемия може да се проследи неговото поведение и промяна на знанията, уменията и компетентостите му.

Предложеният модел на потребителя е обособен като отделен модул, съдържащ данни за потребителя, и система от функции, грижещи се за събирането, съхранението, обработката и интерпретацията на тези данни. Този модул е свързан с модела на предметната област на обучение и обучаващият модул на системата. За целите на изследването се създава модел на обучаемия на базата на измерване на стойността на познаване на отделните понятия в дадената предметна област.

Данните в предлагания модел на обучаемия, се съхраняват в следните структури:

(1) *история на обучението* – състои се от факти от следния вид:

`student_history(name: <име на обучаемия>, <събитие>).`

В полето `name` е записано името на обучаемия, чрез което системата го идентифицира и различава от останалите обучаеми. В разглежданият проект съществуват четири вида събития: лекция, въпрос, задача и заявка (при събитието заявка, обучаемият решава задача, която сам си е поставил, а не е зададена от системата). За всяко от тях в полето `<събитие>` се записва един от следните факти:

```
lecture([ number: <номер на лекцията>,
          date:<дата, на която е протекло събитието>,
          time:<време за протичане на събитието> ]).
question([number: <номер на въпроса>,
          answer:<отговор, даден от обучаемия>,
          eval:<оценка>,
          date:<дата, на която е протекло събитието>,
          time:<време за протичане на събитието> ]).
problem([ number: <номер на задачата>,
          eval_sintaxis:<оценка на синтаксиса>,
          eval_semantic:<оценка на семантиката>,
          date:<дата, на която е протекло събитието>,
          time:<време за протичане на събитието> ]).
query([eval_sintaxis:<оценка на решаваната задача>,
        date:<дата, на която е протекло събитието>,
        time:<време за протичане на събитието> ]).
```

Като задължителна компонента във всеки от фактите присъстват датата и времето (в минути) за протичане на събитието.

(2) *индивидуален модел* – съдържа информация за текущото ниво на знания и умения и текущото емоционално състояние на обучаемия.



Фиг. 33. Архитектура на обучаваща система с модел на обучаемия

В индивидуалния модел се съдържат три типа данни.

- Първият тип представя оценки относно елементарни умения и типични грешки при работа в областта на обучение. Тези оценки се обобщават в общата оценка, показваща уменията на обучаемия при работа в областта.

- Вторият тип показва нивото на усвояване от обучаемия на всяка от единиците знание. За всеки обучаем се поддържа списък от елементи от вида: <номер на ЕЗ> : <оценка на ЕЗ>. оценките за единиците знания могат да се формират чрез изводи, базирани на домейн-независими правила като: ако потребителят ползва дадено знание или функционалност коректно, следователно той го знае; ако потребителя иска помощ за дадена единица знание, значи не го знае; ако иска по-детайлна или пълна информация, следователно го знае и др. Тук се прилагат средства за оценяване на единиците знания, базирани на система от правила.

- Третият тип касае работата на обучаемия по време на обучаващата сесия: съдържа номера на текущата учебна програма, по която се провежда обучението; текуща трудност на задаваните въпроси; трудност на поставяните задачи; нивото на подробност, използвано по време на оказване на помощ или извеждане на съобщение. Тук се включват и данни за обучаемия като: пол, възраст, семеен и социален статус (включва публикации във форум и брой отговори, групово взаимодействие), както и някои индивидуални особености на обучаемия, които са от значение при бучението. Тази компонента е в процес на експериментални изследвания и затрудненията произтичат от факта, че не е известна количествена теория за връзките между психологическите характеристики на обучаемия. Характеристика, реализирана на настоящия етап от работата, е “ниво на забравяне”. Нивото на забравяне тук се дефинира като период от време, за който обучаемият започва да греша (забравя) определено знание или умение.

Всички оценки, указващи нивото на усвоеност или неусвоеност на дадено знание или оценките на въпроси, задачи или заявки на обучаемия, представляват реални числа в интервала $[-1,1]$. В началото, когато за обучаемия още не е извлечена информация, оценките получават стойност (0). По нататъшно пресмятане и настройване на оценките от първия тип се извършва след всяка задача или отговор на въпрос от потребителя.

Обучението е индивидуален процес на търсене на информация, навигиране в средата, извеждане на хипотези и правене на заключения. Затова разглеждаме знанията на обучаемия като подмножество на знанията от съответната предметна област, на базата на модела на припокриването.

Комбинирайки този подход с подхода за поддържане на библиотека от грешки, за всеки от предварително дефинираните типове грешки се поддържат броячи за случаите, когато е било възможно да бъде допусната грешка и броя на грешните опити. След приключване на събитието за всеки от типовете грешка се формира оценка по формулата:

$$(17) \quad Mark_current = 1 - Count_error / Count$$

Тук $Mark_current$ е междинна оценка, която е число в интервала $[0,1]$ и всъщност представлява процента на опитите от страна на потребителя, при които не е допусната грешка. $Count$ е общият брой опити, а $Count_error$ е броят на грешните опити. За да се получи окончателна оценка $Mark$ за умението в интервала $[-1,1]$, се пресмята по формулата:

$$(18) \quad Mark = 2 \cdot Mark_current - 1.$$

Вземайки предвид предните две формули, за оценката получаваме:

$$(19) \quad Mark = 1 - 2 \cdot Count_error / Count.$$

В случая, когато общият брой опити е 0, на оценката присвояваме стойност 0. След като е формирана оценка на типа грешка след събитието, тя се отразява на оценката за съответния тип грешка в модела на потребителя по формулата:

$$(20) \quad Mark_new = (1 - K) \cdot Mark + K \cdot \text{Оценка_на_задачата}.$$

Тук K е число в интервала $[0,1]$ и представлява влиянието, което оценката на типа грешка указва върху оценката в модела на обучаемия. Пресмята се по формулата:

$$(21) \quad K = 0.5 \cdot (Count - 1) / Count$$

По този начин колкото е по-голям броят на опитите, толкова по-голямо влияние ще окаже оценката на задачата върху оценката в модела на обучаемия. Освен това, се умножава оценката по 0.5, което ще ограничи нежелателно влияние, което старата оценка оказва върху новата. Така в оценката за всяко от уменията се съдържа информация от предишни задачи и от последната задача.

Пресмятането на оценките от втория тип от модела на потребителя, а именно оценките за нивото на усвоеност (или неусвоеност) на отделните единици знание се извършва като, за начална стойност се присвоява 0, а след всяка решавана задача или поставен въпрос, оценката се променя по формулата:

$$(22) \quad \text{Оценка_нова} = (1 - K) \cdot \text{Оценка_стара} + K \cdot \text{Оценка_на_проблема}$$

Но тук $\text{оценка_на_проблема}$ е оценката на въпроса или задачата, зададена от преподавателя (експерта) на системата. Както и по-горе, K е влиянието, което оценката на проблема оказва върху оценката на единицата знание и се пресмята по формулата:

$$(23) \quad K = \text{Трудност_на_проблема} / \text{Трудност_на_ЕЗ}$$

По този начин, колкото трудността на проблема (въпроса или задачата) е по-голяма, отнесена към трудността на единицата знание, толкова по-голямо влияние ще окаже оценката на проблема върху оценката на единицата знание. Както и по-горе, удобно е да се редуцира K с 0.5, за да не се изгуби влиянието на решаваните предходни задачи в оценката на единицата знание.

Оценките от интервала $[-1,1]$ се интерпретират по следния начин. Колкото оценката е по-близо до 1, толкова системата е по-сигурна, че обучаемият е усвоил вярно дадената единица знание. Колкото оценката е по-близо до -1, толкова системата е по-сигурна, че обучаемият е усвоил грешно дадената единица знание. А колкото оценката е по-близо до нула, толкова повече системата не е сигурна в знанията на обучаемия. При такъв подход на оценяване се отчитат характерни черти, както на подхода “моделиране с припокриване”, така и на подхода “моделиране с библиотека от грешки”. Тук се има предвид следното: ако оценката е положителна, тя представя нивото на знанията на обучаемия в сравнение със знанията на експерта (характерно за първия подход), ако оценката е отрицателна, тя представя грешно усвоени знания (както при втория подход). Въз основа на това може да се твърди, че представеният подход е от хибриден тип.

Освен това се използват и броячи, с които се следи броя на използваните обучителни ресурси и типа на посетените интернет ресурси, които в различни комбинации могат да осигурят различно представяне на учебните материали. Информационните ресурси се оценяват по брой, видове и др.

Относно характеристиките на някои индивидуални особености на обучаемия, тук се разглежда „*ниво на забравяне*”. Нека например, в някоя от предишните сесии обучаемият е отговорил вярно на даден въпрос. При по-късно задаване на въпрос от същата тема, системата съставя предполагаема оценка, която би получил обучаемият. След получаването на отговор и формиране на действителна оценка, системата я сравнява с предполагаемата оценка и ако първата е по-голяма от втората, оценката за забравяне се увеличава с единица, а в противен случай се намалява с единица.

Този метод на пресмятане на стойностите относно индивидуални психологически характеристики на обучаемия, може да се обобщи по следния начин:

$$\begin{aligned} \text{Предполагаема_оценка} &= F(\text{Оценка_ЕЗ}, \\ &\quad \text{Обективна_характеристика}, \\ &\quad \text{Стойност_на_характеристиката}), \\ \text{Стойност_характеристиката_нова} &= G(\text{Стойност_стара}, \\ &\quad \text{Предполагаема_оценка}, \\ &\quad \text{Действителна_оценка}). \end{aligned}$$

Функциите F и G зависят от естеството на поставения проблем и на психологическата характеристика. Разглеждат се възможностите за моделиране на когнитивните способности на обучаемия чрез механизъм, по който системата де е в състояние на адаптира предлаганите обучаващи материали и допълнителни ресурси (речници, справочници, карти и т.н.) към отделния обучаем. Целта е обучаващият процес да съответства най-пълно на познавателната активност на обучаемия и да спомага за подобряване на неговото абстрактно и индуктивно мислене.

Разпознаване на емоционално състояние в модела на обучаемия при E-Learning

За целите на провежданото изследване в дисертационния труд е разработено софтуерно приложение за разпознаване на човешки емоции по образа на лицето. Разработеният софтуер се базира на системата за кодиране на лицевото действие (FACS) на Ekman и Friesen [76]. Лицевият израз на емоциите се определя от включените мускули и жестовете, които го характеризират. Съществуват определени отличителни модели на емоционални реакции, обобщени и споделени от повечето хора. Те се считат за основни емоции: щастие (Joy), тъга (Sadness), гняв (Anger), изненада (Surprise), страх (Fear) и отвращение (Disgust). Други подобни кодиращи системи са EMFACS, MAX, AFFEX и CANDIDE-3. Разработеният софтуер и проведеното изследване са представени в [140].

Първата стъпка е категоризиране на зададените снимки със изображения на лица, в отделни поддиректории. Те са седем на брой (включвайки и *Neutral*), като всяка за от тях отговаря за конкретна емоция и съдържа около петдесет примерни изображения, свързани със съответната емоция. Следваща стъпка е използването на *Dlib* библиотеката с отворен код, чрез която се осъществява обработката на съответните снимки. *Dlib* е библиотека от алгоритми за машинно обучение и инструменти за създаване на софтуер с общо предназначение, която се използва в много области: индустрията, обучението, роботика, мобилни телефони, големи компютърни среди с висока производителност и редица други. Чрез използването на *srape_predictor_68_face_landmarks.dat*, лицата от изображенията се маркират със съответни основни точки. На базата на поставените точки, в следващата стъпка се извършват математически изчисления, с помощта на които се определят б нормализирани вектора на всяко лице. Всеки вектор с лицеви характеристики е изчислен като данните се нормализират и се съберат разстоянията между маркираните точки. Чрез средствата на софтуера „ML.Net“ се имплементира многокласов класификационен алгоритъм, базиран на ML агент от тип Supervised Learning.

След обучението на модела, той може да разпознава емоционалното състояние на подадено изображение (снимка) на лице. В резултат на работата, програмата връща предвиждане относно емоционалното състояние. На базата на тестови данни се извеждат оценки за точността на предвиждането. Анализът на резултатите от метриците за оценка на точността показва, че най-голяма точност на предвиждането се постига при емоционалните състояния *Joy*, *Surprise* и *Sadness*. Състоянието *Anger* често се бърка от алгоритъма със *Sadness*, *Joy* или *Neutral*. Алгоритъмът може да бъде подобрен чрез обучение с много по-голям обем тренировъчни данни от различните типове емоционални състояния. Но трябва да се използва балансиран набор от данни, защото алгоритмите за класификация могат да се повлияят от недостиг на данни от определен тип, което да води до грешни предвиждания. Алтернативен подход е прилагане на алгоритъм от тип *Unsupervised Learning*, при който се изследва структурата на данните без указания, данните за обучение не са етикетирани. Могат да се използват *Clustering* алгоритми или *Deep Learning*. Прилагайки различни класификатори може да се експериментира с получаваните резултати за повишаване на точността, спрямо различни групи и категории данни. Целта е разработваното приложение да се интегрира към среда за e-Learning и да се търсят промените в емоционалните състояния на учащите се, докато се справят с когнитивните задачи в процеса на обучение.

Устройство за подпомагане на потребители с увреден слух

Проектирано и експериментално изследвано е устройство [219], което произвежда широка гама от цветове и нива на интензитет, за да визуализира различни аспекти на звука. Звуковата визуализация за хора с увреден слух осигурява подобро усещане за заобикалящата среда.

За преобразуване от звукова честота към цветови стойности е приложена методологията и математическите постановки на Ди Паскуале [75], като звуковата честота F , Hz се умножава по 2^{40} и се представя в GHz. Дължината на вълната λ , nm се изчислява по формулата:

$$(24) \quad \lambda = \frac{V_L}{F}, nm$$

където V_L , nm/s е скоростта на светлината във вакуум ($V_L=299792458000000000$ nm/s), F , Hz е честотата на звука. Например, при $F=699$ Hz, умножено по 2^{40} е 768558627815424 Hz или $768\,558,627815424$ GHz. Разделяйки скоростта на светлината на честотата на звука ($299792458000000000/768558627815424$), се получава $\lambda = 390,071$ nm, което е синия цвят.

За изграждане на техническия инструмент за визуализиране на звук чрез цвят са избрани хардуерните компоненти: наносъвместим микрокомпютър с платка Atmega 328p и CH340 USB към сериен конвертор (Kuongshun Electronic Ltd.), цифров RGB диод с драйвер WS2811 (Pololu Corp.), електричен микрофон с усилвател MAX9814 (Adafruit Industries), USB AC/DC зарядно устройство 5V DC, 2A (Hama GmbH & Co KG).

Основната логика на програмата, изпълнявана от микрокомпютъра е в циклично изпълнение на дейностите: чете от микрофона и изчислява честотата на входния сигнал; търси най-близката съответстваща честота в предварително дефинирана честотна карта и се извличат съответните RGB стойности за тази честота; извлечените RGB стойности се визуализират чрез RGB LED; следва нулиране на променливите и начало на следващ входен поток, изчисляване честотата и съответно актуализиране цвета на светодиода.

Например: стойност на честотата $F=710$ Hz, попада между честотите на дванадесетия и тринадесетия стандартен цвят. Следва интерполиране на RGB стойностите $RGB=[36, 0, 179]$. Честотата е по-близо до тринадесетия стандартен цвят, затова полученият цвят е син.

Направен е сравнителен анализ между цветовете, получени от предложеното устройство и онлайн инструмент. При ниски честоти се отчитат относително малки разлики между предложеното устройство и онлайн инструмента, с цветови разлики около или близки до 0, но при високи честоти (над 100 Hz) се наблюдава значителна разлика.

Предлаганото в настоящата работа устройство има фиксиран брой цветове. По този начин потребителите могат лесно да научат и запомнят асоциациите между цветовете и звуковите честоти. Това опростява процеса на обучение и позволява на хората с увреден слух точно да интерпретират и да реагират на слухова информация в реално време.

Работа трябва да продължи с изследвания, насочени към обобщаване на моделите за преобразуване на звуковите честоти в цвят.

3.5. Когнитивен агент в среда за електронно обучение

Когнитивният агент [188] е автономен софтуер, който има способността да развива своите знания. Като се има предвид различната степен на интелигентност, когнитивните агенти могат да бъдат реализирани на базата на правила, които развиват знанията си с помощта на вградени в системата механизми за извод или извличат нови знания с помощта на техники на машинно обучение и Data Mining техники.

Общата рамка на архитектурата, на изгражданата за това изследване система, базирана на правила - **Rule Based Systems** включва компонентите [181]:

о **База от знания** (множество от правила) - съдържа всички знания необходими на агента. В изгражданата обучаваща среда това са правила от вида "IF условия THEN действия", които представят общите знания за областта.

Базата съхранява знанията за предметната област, представени под формата на импликации от вида:

(<име_на_правило> <лява_страна><дясна_страна>)

Лявата страна на правилото е условие (предпоставка), а дясната страна е следствие (действие). Най-често лявата страна представлява конюнкция от елементи на условието. Дясната страна е поредица от елементи, описващи действия, които ще се изпълнят.

Структурата на елементите на условията и елементите на десните страни на правилата е същата като тази на елементите в работната памет. Имаме зададени начална и крайни ситуации. В дадена ситуация (състояние) трябва да намерим подходящо правило, при активирането, на което действието му поражда нова ситуация (състояние).

о **База от данни**: съдържа данните, които са установени (известни) към текущия момент. Произхода на данните в работната памет е:

- зададени по условие при дефинирането на проблема;
- от интерпретатора на правилата, получени в процеса на извод;

Базата от данни може да бъде променяна посредством операторите:

- REMOVE(елемент) – отстранява елемент от базата от данни;
- ADD(елемент) – добавя елемент към базата от данни;

о **Машината за извод** (Inference Engine) - интерпретаторът избира и прилага правилата, които могат да доведат до промяна на състоянието на базата от данни. Процесът на работа на интерпретатора на правилата е цикличен, като на всяка стъпка от работния цикъл на интерпретатора се прави избор на правило, което да се изпълни на текущата стъпка, като се извършват множество операции на съпоставяне по образец и изпълнение на избраното правило. За реализиране на тази схема съществуват различни подходи: свързване напред, свързване назад и смесени подходи [181]. В случая на свързване назад, системата търси път като идентифицира множества от приложими правила с дадена дясна страна и избор на правила от тези множества като подцели, които след това се опитва да установи.

о **Базата от мета-знания** - множество от мета-правила, задаващи връзката между правилата [62]. Начинът, по който се избират правилата от множеството на приложимите правила е стратегията за решаване на конфликти (Conflict Resolution). Обикновено

стратегииите използват някои от следните принципи: правилата се избират произволно; избира се идентифицираното първо приложимо правило; задейства се правилото с най-голям брой условия, по-дългото правило е по-специфично за конкретното решение; правилата са снабдени с приоритети (тежести), които определят кое правило да се избере; избират се различни правила в последователни етапи и други подходи.

В случая се прилагат приоритети на правилата, които са динамични. Динамичните приоритети се определят въз основа на важността на действията в настоящата ситуация. Например: ако *оценката е висока*, то се задава нисък приоритет на правилото за търсене на следваща задача за решаване. Тогава обучаемият ще търси следваща задача за решаване само, ако няма други дейности за изпълнение. Но ако е със *слаба оценка*, правилото за решаване на следваща задача трябва да има висок приоритет. Обучаемият трябва да решава следваща задача, за да повиши оценката си. Това може да се приложи например с помощта на правилата:

- едно правило за ситуацията „*няма повече задачи за решаване И оценката е слаба*“
- друго правило „*няма повече задачи за решаване И оценката е висока*“.

Използването на динамични приоритети позволява правилото постепенно да става важно, вместо внезапно да се превръща в правило с висок приоритет. Този подход включва търсене измежду всички възможни правила, преди да се реши кое да се задейства. Подходът е гъвкав, но отнемащ време.

За да се реализира обучаващият агент като когнитивен, базирайки се на данните, натрупвани през годините в средата за електронно обучение Мудъл, изследваме възможността за извличане на знания и връзки между данните за студентите в процеса на обучението. Правилата, които се извличат и анализират стават голям брой, който непрекъснато нараства. Идеята е, да групираме правилата в отделни групи (контексти). Когато се налага агент да вземе решение за обучаващия обект на база на правилата – той да не обработва последователно всички правила, а само правилата от групата, която е с най-голяма степен на сходство с търсената цел. Сходството между две правила измерва степента на прилика между тях, различието определя степента на несъответствие. Интуитивно, правилата са подобни, когато те споделят много идентични компоненти.

След проведени експерименти с прототип на обучаващ агент, реализиран чрез система базирана на правила се установи, че програмната задача може да поддържа много сложно поведение, при условие че са създадени достатъчно добри правила.

От друга страна **Reinforcement Learning** (Обучение с утвърждение/ Подсилващо обучение) става все по-популярен тип машинно обучение. Това е новаторска техника, при която автономните агенти използват алгоритмите за проба-грешка и функция за кумулативно награждаване на обучението. Предимството се състои в калкулирането на оптимални действия, които агентите могат да предприемат в рамките на сценарии, обусловени от средата. Това обучение се прилага обикновено при липса на предварително зададен „правилен“ набор от обучителни данни. Такъв подход е коренно различен от начина на действие при контролирано обучение, където целта е намаляване на отклонението съобразно предварително зададени правилни данни (вход/изход).

Обучението протича като последователност от пробни действия, които постепенно водят до утвърждаване на добрите действия и избягване на неподходящите. Резултатът от обучението е оптимална стратегия за действие във всяка ситуация. Стратегията е оптимална, ако успява да максимизира сумата от всички награди, получени по време на изпълнението.

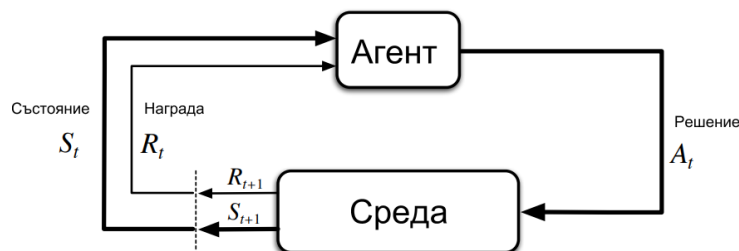
Формално обучението чрез Reinforcement Learning може да се разглежда като [133]:

- Множество S , на всички състояния на обкръжението на агента.
- Множество A , всички възможни действия на агента.
- Множество от правила за преходи между състоянията.
- Правила, които определят R - наградите, които агентът може да получи при преходите.
- Правила, които описват целите на агента.

Reinforcement learning третира света на играта като *state machine*. Агентът взаимодейства със заобикалящата го среда на дискретни времеви стъпки (фигура 34).

Във всяко време t агентът:

- Получава наблюдението o_t
- Което обикновено включва наградата r_t
- Избира действие Action a_t , което се изпраща към околната среда
- Средата преминава към ново състояние s_{t+1}
- Определя се наградата r_{t+1} свързана с прехода (s_t, a_t, s_{t+1}) .



Фиг. 34. Схема на взаимодействие на агента със заобикалящата го среда

Целта на агента е да максимизира сумарната награда, която може да получи от всяка една стъпка: $R = r_0 + r_1 + r_2 + \dots + r_H$

Награди, които могат да бъдат получени по-скоро във времето са по-високо ценени от награди, които могат да бъдат получени в по-далечното бъдеще. За тази цел се въвежда регулация на преоценителния коефициент γ :

$$(25) \quad R = r_0 + \gamma r_1 + \gamma^2 r_2 + \dots + \gamma^H r_H$$

За целта разглеждаме γ в интервала $(0,1)$. При $\gamma=1$ уравненията (24) и (25) са еквивалентни.

Поведението на агента в средата се определя от стратегия π , която той следва. π се разглежда като функция, която приема стойности от множеството на състоянията на агента $s \in S$ и връща решение $a \in A$, което агента взема. Задачата е по зададено описание на действията на агента и динамиката на средата, да се определи оптимална стратегия π^* , при която сумарната награда R да е максимална.

A^s е броя възможни стратегии, които съществуват – във всяко едно състояние, агента може да вземе A на брой възможни решения. Само една от тези стратегии е оптимална – носи на агента максимална възможна награда. Агентът трябва да достигне целта си с колкото може по-малко стъпки. Търсим полезността на различните възможни състояния на агента.

Полезността се обозначава с $V(s)$, за всеки $s \in \mathcal{S}$. $V^*(s)$ определя максималната награда, която може да добие агента, ако започне в състояние s и следва възможно най-добрата от всички стратегии π , докато не стигне в крайно състояние [40]:

$$(26) \quad V^*(s) = \max_{\pi} \mathbb{E} \left[\sum_{t=0}^H \gamma^t R(s_t, a_t, s_{t+1}) \mid \pi, s_0 = s \right]$$

Връзката между $R(s, a, s')$ и $V^*(s)$. $R(s, a, s')$ е функция, която определя моменталната награда, която агента получава, ако премине в състояние s' от s , при взето решение a – полезността на това състояние s е същата като съответната моментална награда. Полезността на всички останали състояния, които нямат моментални награди свързани с тях, е пропорционална на разстоянието от тях до най-близкото състояние с моментална награда. Така $V^*(s)$ определя евентуалната очаквана награда, която агента би получил, ако във времева стъпка K той вземе възможно най-доброто решение и след това продължи оптимално през останалите $K-1$ времеви стъпки. Придържането към оптимално решение в оставащите $K-1$ времеви стъпки води до рекурсивна връзка между стойностите на полезност на състоянията на агента:

(27) *Рекурсивно уравнение на Белман:*

$$V_k^*(s) \leftarrow \max_a \sum_{s'} P(s'|s, a) (R(s, a, s') + \gamma V_{k-1}^*(s'))$$

Търсим стойностите на полезност при H оставащи на брой възможни стъпки на агента. Използвайки рекурсивната зависимост можем да започнем с $K=0$ и да извършваме обновяване на стойностите, докато $K=H$. Уравнението на Белман гарантира, че колкото по-голямо е K , толкова по малко ще се променя V_{K+1} спрямо V_K , така, след определена стойност на K ще открием оптималните стойности на полезност и ще изведем оптимална стратегия – колкото повече K расте, толкова по малка относителна промяна наблюдаваме в V_{K-1} , V_K , V_{K+1} , които клонят към V^* [180].

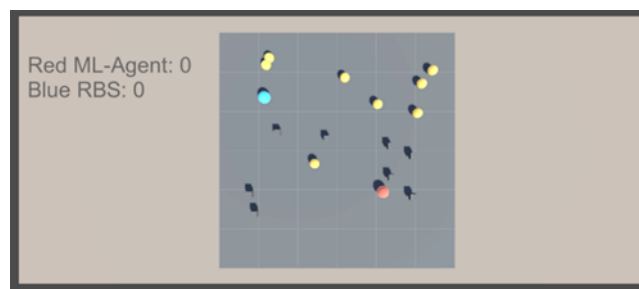
Конкретна задача, поставена в дисертационния труд е да се анализира и сравни поведението на два агента, реализирани на базата на два различни подхода, базирани на машинно обучение. Резултатите от проведеното изследване и разработеният за целта софтуер са представени в [141].

За целите на анализа е създадена игра, която предоставя платформа и генерира x брой бомби и монети върху нея. Играта създава два агента, представени като две различни топки на платформата – червена и синя. Платформата няма оградящи стени, което означава, че топките (агентите) могат да паднат. Ако агент падне, той губи играта. Всеки агент може да събира монети, като всяка монета носи на агента точка. Ако агент докосне бомба, той губи точка. Играта приключва, когато монетите свършат или когато агент падне от платформата. Агентът с повече точки печели.

Един агент, представен със синя топка в играта, поддържа поведение въз основа на система от правила, създадени от програмиста и вградени в базата знания на системата. В този случай при изграждането на играта се прилага подходът *Rule-Based System*.

От друга страна, агент, представен с червена топка, се реализира въз основа на подхода за машинно обучение с *Reinforcement Learning*. При него изкуственият интелект научава сам кое е най-доброто действие във всяка ситуация и с времето оптимизира решенията, които взема.

Разгледани са разликите в поведението на създадените решения чрез избраните два подхода. Хипотезата е, че агентът, реализиран с машинно обучение чрез *Reinforcement Learning*, ще има по-разнообразно поведение и след достатъчно време за обучение ще работи по-ефективно от агента, реализиран като система, базирана на правила.



Фигура 35. Изглед на разработения софтуер

Причината за избора на *Reinforcement Learning* е широкото използване на този подход в областта на игралната индустрия [133]. Повече експерименти и задълбочаване на разбирането на този тип машинно обучение, допринасят за разширяване на обхвата на използването му в съвременните среди за обучение. От друга страна алтернативата на системите, базирани на правила е подход доказал във времето своите предимства и ефективност за бързо постигане на поставените цели.

Проектът [141] е създаден с помощта на платформата *Unity*, заедно с пакет на *Unity Technology* за разработване на агенти за машинно обучение *ML-Agents*. Математическият алгоритъм, основан на рекурсивните уравнения на Белман (представен по-горе) е основа в реализацията на инструмента. *ML-Agents* пакета предоставя разнообразие от сценарии за обучение и валидиране, които включват различни конфигурации и взаимодействия между агенти, наказания и награди. Всеки агент може да има уникален набор от състояния и наблюдения, да предприема действия в средата и да получава награди за събития в нея.

- *Методология на изследването*

Първият агент с *Reinforcement Learning*, е създаден чрез наследяване от клас *Agent*. Този клас позволява използването на вече създадени функции за обучение на агенти. Предефинира се методът *OnEpisodeBegin*, в който се задават първоначалните параметри за играта (монети и бомби и техните позиции).

Друг предварително дефиниран метод, наследен от агент, е *CollectObservations*, който предава текуща информация за посоката към най-близката монета. *OnActionReceived* метод задава разрешените движения на агента.

ML агентът има два компонента на *Ray Perception Sensor*, които му позволяват да наблюдава обектите в света на играта. Чрез първия компонент, агентът получава

информация за монети и бомби в района. А другият е насочен под ъгъл, така че агентът да наблюдава къде платформата свършва.

ML_agent е основния клас в този проект, той определя броя и позициите на монети и бомби. Той също следи резултата и определя победителя в края на играта, след което извиква функцията *EndEpisode*. Когато агент докосне обект, действието се засича и се извиква функцията *OnTrigger*, която позволява да се види с кой обект агентът е влязъл в контакт. Резултатът се увеличава при контакт с монета и намалява при контакт с бомба. За да се установи дали даден агент е паднал от платформата, се следи стойността на позиция "Y". Ако стойността е отрицателна., то агентът е паднал от платформата и той губи играта, а текущото ниво приключва. Текущият резултат на екрана показва колко точки са събрали двата агенти в текущата сесия. Когато ML агентът спечели, цветът на платформата става зелен и червен, когато загуби.

Вторият агент, базиран на система от правила, е имплементиран чрез *RBS* класа и дейността на агента се задава от реализираните методи в класа. В зависимост от сценариите се прилагат правилата:

- Първо правило: „Ако има монети на платформата, отиди да вземеш най-близката“.
- Второ правило „Ако другият агент е близо до края на платформата, опитай да го свалиш от платформата“.
- Трето правило: „Ако *RBS* агента има повече монети от другия агент, той не трябва да се страхува от бомбите, за да завърши играта по-бързо и да не позволи на противника да събере повече монети.

Изпълнението на правилата се основава на подхода свързване напред (*forward chaining*) и системата изпълнява последователно действията:

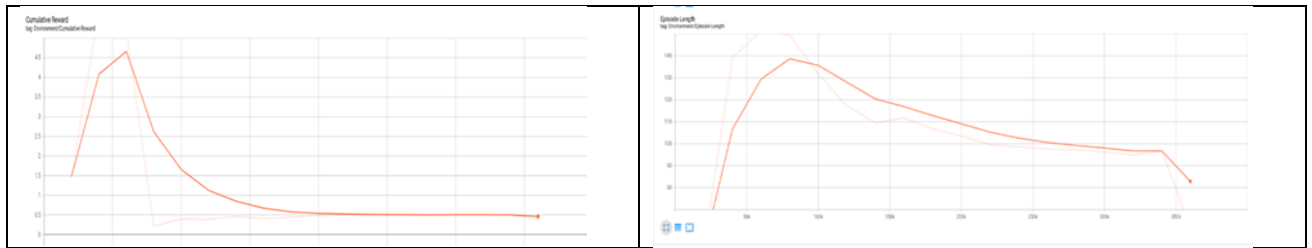
- (1) идентифициране на набор от приложими правила, чиято условна част е изпълнена.
- (2) избор на първото правило от това множество.
- (3) изпълнение на правилото.

• *Тестване и резултати*

Резултатите от работата на двата агента могат да бъдат обобщени със следното твърдение - агентът, реализиран с *Reinforcement Learning* (RL), превъзхожда базирания на *Rule-Based System* (RBS) агент и може да бъде допълнително обучаван. Но в играта поради вградените фактори на случайност при разпределението на монети, в много случаи RBS агентът, печели, защото попада в по-добра позиция. Понастоящем, проведените опити показват, че ако играта се наблюдава достатъчно голям брой пъти, процентът на печалба и на двата агенти ще бъде около 50%.

Интересни модели се появяват в поведението на RL агента по време на обучението му чрез задаване на различни стойности на параметрите. Например, за RL агент със зададен параметър да използва непрекъснати действия, той открива потенциално оптимален елиптичен маршрут, наподобяващ "0", като по този начин поддържа висока скорост и преминава през повечето от монетите, но не може да избегне всички бомби. Това се дължи на стремежа на агента да завърши играта бързо и да не позволи на противника да събере повече монети. Когато се използва алтернативата *Discrete Action*,

поведението на RL агента е близко до поведението на RBS агента, но запазва своята ефективност при движението за събиране на монети.



Фигура 36 а) Кумулативна награда на RL агента. б) Продължителност на епизода на RL агента

На фигура 36а е дадена диаграма на кумулативната награда на RL агента по време на играта. Тя показва, че в началото наградата, която агентът събира, се увеличава бързо, тъй като този агент първоначално е обучен без опонент и всички монети са събрани от агента без състезание. По време на играта, когато другия агент също започва да събира монети, точките, получени от RL агента, намаляват поради невъзможността да събере всички монети.

На фигура 36б е дадена диаграма на средната продължителност в стъпки на работата на RL агента по време на неговото обучение. В началото агентът не се наказва, когато иска решение за действие и поради това агентът няма стимул да завърши нивото по-бързо. След като агентът натрупа опит, започва да се прилага наказание за по-дълго време за завършване. В резултат на това той се стреми да завършва играта по-бързо.

ИЗВОД: *Reinforcement Learning* агентът постепенно научава кое е най-доброто действие във всяка дадена ситуация и с времето оптимизира решенията, които взема. Той първоначално работи неефективно, но с течение на времето ще оптимизира действията си. От друга страна, при условие, че са създадени достатъчно добри правила, *Rule-Based System* е програмна задача, която може да поддържа много сложно поведение.

Интересна тенденция в по-сложните приложения е съвместно прилагане на *Reinforcement Learning* с контролирано и неконтролирано обучение, ако използваните самостоятелно методи не дават достатъчно добър резултат.

3.6. Модели за йерархично многокомпонентно оценяване на обучаеми

Тук са представени *нови модели* за йерархично многокомпонентно оценяване на обучаемите, които имат за цел комплексно оценяване на различни мисловни умения от висок и нисък порядък, теоретични знания и практически умения, и др. Използва се подход за разрито оценяване за формализиране на субективната логика на оценителя.

Една от значимите таксономии, издържали изпитанията на времето, е таксономията на Блум [44]. Тя дефинира йерархия на мисловните умения, в която по-високите нива на мислене включват всички познавателни умения от по-долните нива. Нивата са структурирани като: *Knowledge* (Знания), *Comprehension* (Разбиране), *Application* (Приложение), *Analysis* (Анализ), *Synthesis* (Синтез) и *Evaluation* (Оценяване). Всяко ниво се определя от множество когнитивни умения, дейности и методи за оценка. Счита се, че в процеса на обучение, обучаемият преминава последователно през всички нива.

Таксономията на Блум предоставя рамка за определяне и степенуване на целите и задачите на обучението, планиране на дейностите в процеса на обучението, формулиране на проблеми и поставяне на задачи пред обучаемите, избор на оценъчни инструменти, адекватни на поставяните цели, и др. Една общоприета класификация определя уменията от горните три нива на Блум (*Analysis*, *Synthesis* и *Evaluation*) като *higher-order thinking skills* (HOTS), а тези от долните три нива (*Knowledge*, *Comprehension* и *Application*) – като *lower-order thinking skills* (LOTS). HOTS са изключително ценни и много важни за прогреса на всеки член на обществото, а също така и на обществото като цяло. Това включва способности за критично мислене, умения за трансфер на знания, умения за решаване на проблеми, и др. [47].

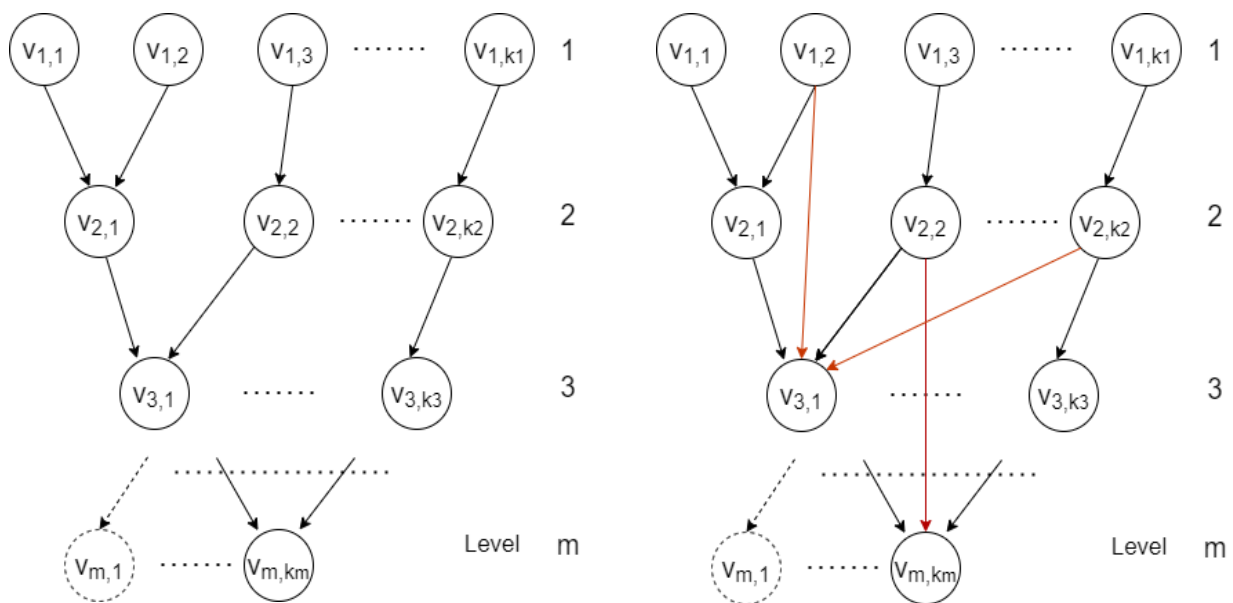
Оценката на знанията и уменията на студентите трябва да е обективна и при различни проверки да се поставят еднакви оценки. Това в много случаи е трудно постижимо, особено когато се отнася за оценяване на HOTS, които изискват креативно мислене. Именно това мотивира работа в насока създаване на комплексни многокомпонентни модели за оценяване на студентските постижения. Описаните тук и публикувани в [98] модели черпят идеи от теорията на размитите множества.

При многокомпонентно оценяване, крайната оценка е функция на множество оценъчни компоненти, които са измерители на различни знания и умения, включително теоретични и практически. *Оценъчните компоненти* могат да са разнообразни – тестове, задачи, курсови работи, казуси, проекти, есета и др. Те могат да изискват както самостоятелна работа от обучаемите, така и работа в екип. Възможно е да бъдат оценявани по различни начини – оценяване от преподавател, самооценяване, колегиално оценяване и др., и чрез различни оценъчни скали/системи. Въпреки добрите резултати за автоматизирано оценяване, получени с невронни мрежи и други методи на изкуствения интелект, в общия случай липсва яснота/обосновка за обективността на поставяните оценки. Субективният фактор се проявява най-често в гранични случаи, например за оценъчна скала с обхват от 2 до 6, гранични проблемни случаи са такива, при които реалните оценки са около 2.5, 3.5, 4.5, 5.5.

Възможен подход за създаването на комплексно решение, е изследването на множество методи за оценяване, с цел тяхното сравнение и евентуално едновременно използване в автоматизирана система за оценяване. Работейки в тази насока, тук се

предлага **моделиране на йерархичната организация на оценъчните компоненти и зависимостите помежду им, които оценителят явно или неявно използва при оценяване** (фиг. 27). Например, при множество дисциплини, основни оценъчни компоненти са „практика“ и „теория“, а в използваните термините тук: **thinking skills**, съответни техни подкомпоненти могат да бъдат теоретични и практически **LOTS** и **HOTS**. От друга страна ако изберем за основни компоненти **HOTS** и **LOTS**, те биха могли да имат под-компоненти за оценяване на теоретични знания и практически умения.

Създаването на конкретна йерархия от компоненти и подкомпоненти може да бъде извършено с помощта на предварително стандартизирани модели, избрани от оценителя. При достатъчно входно-изходни образци и ефективно обучение на невронна мрежа, получените резултати могат да отразяват тази йерархична организация, дори без тя да е указана явно.



Фигура 37. а) **Модел 1:** Йерархична дървовидна организация на оценъчните компоненти

Фигура 37. б) **Модел 2:** Йерархична графова организация на оценъчните компоненти

На фиг. 37 а) е представен общ **Модел 1 на йерархична дървовидна организация на оценъчните компоненти**, при който всеки основен компонент, в частност и крайната оценка, зависи от множество подкомпоненти на предходното ниво. При това, всеки подкомпонент влияе само върху един компонент от по-високо ниво.

Модел 2 на йерархична графова организация на оценъчните компоненти (фиг. 37 б), разширява и обобщава възможностите на Модел 1. Основни разширения се получават от възможността един компонент да влияе на множество компоненти от произволни по-високи нива, както и да зависи от компоненти на различни по-ниски нива.

Представените модели могат да се използват при множество различни подходи за оценяване, при които крайната оценката е функция на множество оценъчни компоненти.

Основни характеристики на предложените моделите са:

- Ниво 1 на компонентната йерархия описва конкретни стойности от проведени оценявания, които могат да бъдат с различни оценъчни скали, включително писмени или устни, практически или теоретични. Примери за такива стойности са: точки от тестове за теория и практика; точки от тестове за умения, според таксономията на Блум; оценки от задачи, курсови проекти, есета, казуси, и др.

- На всяко следващо ниво, компонентите формират оценка, която е функция на оценките на подчинените му под-компоненти от предходното ниво. В частност, всички или част от функциите могат да бъдат с размитата логика.

- На последно ниво има един компонент, т.е. $k_m=1$, но това не е задължително. Възможно е да има няколко крайни оценки, например за различни оценявани когнитивни умения. При това $k_m \geq 1$.

Възможен вариант за функции на ниво 2 е нормирането на всички стойности на компонентите от ниво 1 до една обща система за оценяване (например от 2 до 6). Аналогично, функциите на всяко следващо ниво биха могли да запазват нормираността на родителските компоненти в съответната скала за оценяване. Причина за подобен подход е придържането към логиката на оценителя, който често разсъждава в категориите/рамките на конкретна скала за оценяване. Общия модел допуска всякакви междинни резултати. Възможно е например крайната оценката да е кумулативна и да се образува като сума от подкомпонентите.

Обикновено, оценъчните под-компоненти имат стойности в пространството на реалните положителни числа R^+ . При различни подходи за оценяване, стойностите могат да бъдат и в други пространства на реални, комплексни или други числа. Крайната оценка E задължително принадлежи на пространство от предварително определени възможни стойности – например тя трябва да е цяло число в интервала от 2 до 6, от 1 до 5, от А до F, и др.

При формалното математическо описание на *Модел 1* и *Модел 2*, всяко ниво с номер i притежава k_i на брой компоненти, определящи множеството:

$$(28) \quad V_i = (v_{i,1}, v_{i,2}, \dots, v_{i,k_i}) \in R^{k_i}, k_i \in N, i = 1, \dots, m$$

Компонентите на ниво 1 са конкретни стойности, получени при оценяване. На всяко следващо ниво $i = 2..m$ стойностите се получават в резултат на прилагането на функции върху компонентите от предходното ниво. В *Модел 1* броят на компонентите в дадено ниво намалява или се запазва равен на броя на компонентите в предходното ниво $k_s \leq k_{s-1}, s = 2, \dots, m$, което е следствие на обединяването на под-компонентите в компоненти. Броят на обобщаващите функции и съответно на компонентите, може да нараства в определени нива при *Модел 2*.

Стойностите на компонентите, след ниво 1, при *Модел 1* се изчисляват като **функция** върху стойности на компонентите от преките родителски нива:

$$(29) \quad v_{i,j} = f_{i,j}(P_{i-1,j}), P_{i-1,j} = (v_{i-1,s_1}, v_{i-1,s_2}, \dots, v_{i-1,s_j}) \subseteq V_{i-1},$$

$$\forall i = 2, \dots, m, j = 1, \dots, k_i, s_j \leq k_{i-1} \text{ и}$$

компонентите на V_{i-1} се срещат само веднъж в някое $P_{i-1,j}$.

При *Модел 2*, параметри на функциите на дадено ниво, могат да бъдат произволни оценъчни компоненти от всички предходни нива, като при това отпадат множество ограничения:

$$(30) \quad v_{i,j} = f_{i,j}(Q), Q \subseteq \bigcup_{r=1}^{i-1} V_r, \forall i = 2..m, j = 1..k_i$$

Експеримент с използване на размита логика при йерархично многокомпонентно оценяване

В проведения експеримент по дисциплината „Програмиране в Интернет“, чрез четири изпитни компонента, оценяваме теоретични знания и практически умения за програмиране:

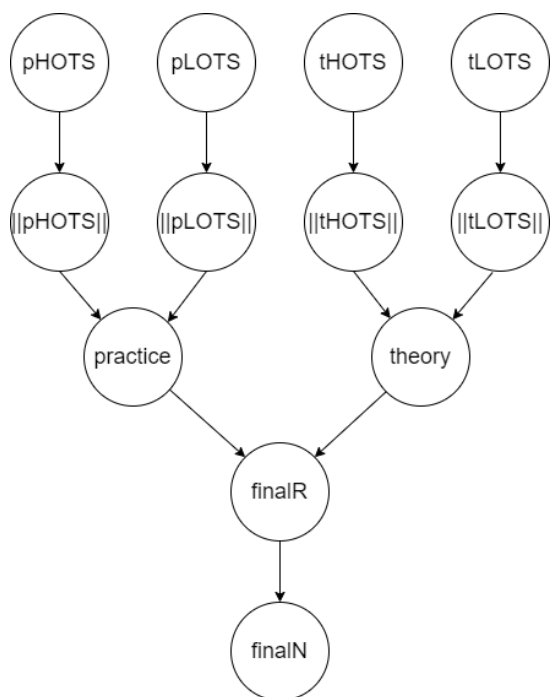
- tLOTS – теоретични LOTS, оценяват се чрез точки в интервал [0, 30];
- pLOTS – практически LOTS, оценката е цяло число в интервала [0, 15];
- tHOTS – теоретични HOTS, оценката е цяло число в интервала [0, 15];
- pHOTS – практически HOTS, оценката е реално число в интервала [2, 6].

Оценката pHOTS се формира от практическа задача и е в интервала [2, 6]. Точките на останалите оценъчни компоненти се формират при решаване на тест в хартиен или електронен вариант.

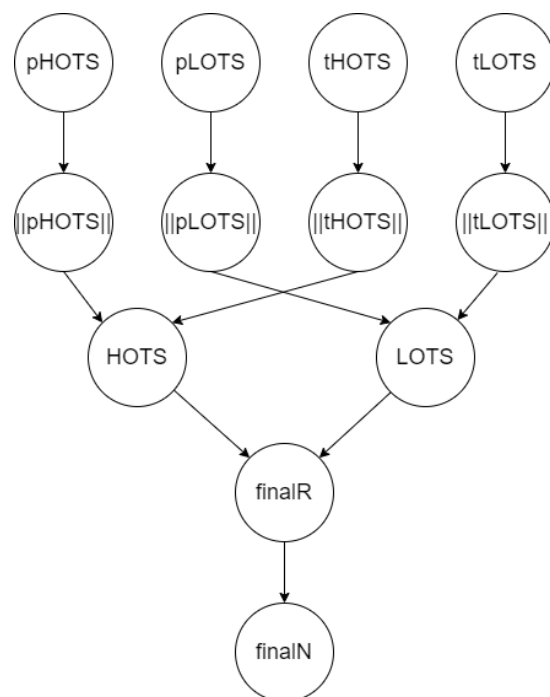
Крайната оценка finalN е цяло число в скалата за оценяване [2, 6], т.е. finalN \in [2, 6] \subset N. Базирайки се на *Модел 1 на йерархична дървовидна организация на компонентите за оценяване*, дефинираме два основни възможни варианта за *формализиране логиката на оценителя* за оценяване по дисциплината (фиг. 37). При необходимост, чрез дефиниране на допълнителни връзки/зависимости между компоненти на различни нива, може да преминем към *Модел 2*. В практиката при различни ситуации и дисциплини могат да бъдат създадени и експериментирани и други варианти. Важното при всички случаи е, да се определи йерархията от компоненти и връзки между тях, както и множеството от съответните им агрегиращи функции, подходящи за дадено оценяване.

Основни компоненти, формиращи крайната оценка при *Вариант 1 за теоретико-практическото йерархично многокомпонентно оценяване* са *теория* и *практика* (фиг. 38 а). Под-компоненти на теорията са *нормализираните форми* на първичните компоненти *tLOTS* и *tHOTS*, а на практиката – нормализираните форми на *pLOTS* и *pHOTS*. Целта на нормализираните форми е свеждането на първичните компоненти до еднаква скала за оценяване (в случая от 2 до 6), така че да бъдат представени до по-близки за оценителя мисловни рамки за оценяването.

От друга страна, при *Вариант 2 за HOTS-LOTS йерархично многокомпонентно оценяване* (фиг. 38 б), за основни компоненти са приети HOTS и LOTS, а техни под-компоненти са съответно нормализираните форми на практическите и теоретични HOTS, и практическите и теоретични LOTS.



Фигура 38. а) **Вариант 1:** Теоретико-практическо йерархично многокомпонентно оценяване за **Модел 1**



Фигура 38 б) **Вариант 2:** HOTS-LOTS йерархично многокомпонентно оценяване за **Модел 1**

И в двата моделирани варианта на първо функционално ниво са нормализирани форми на основните оценъчни компоненти. Конкретните нормализиращи функции могат да следват *либерален* или *строг подход* за оценяване, или тяхна модификация, в зависимост от преценката на оценителя. При *либералния подход*, точките от първичните компоненти се свеждат директно до интервала [2, 6]:

$$(31.1) \quad ||tLOTS|| = 2 + 4 \frac{tLOTS}{30}$$

$$(31.2) \quad ||tHOTS|| = 2 + 4 \frac{tHOTS}{15}$$

$$(31.3) \quad ||pLOTS|| = 2 + 4 \frac{pLOTS}{15}$$

$$(31.4) \quad ||pHOTS|| = pHOTS.$$

При *строгият подход за оценяване*, точките първоначално се изобразяват в интервала [0, 6]. Оценки под 2,5 се закръглят на 2. Съответно, функциите на принадлежност (*membership functions*) за компонентите от второ ниво са:

$$(32.1.1) \quad ||tLOTS|| = \begin{cases} 2, & 6 \frac{tLOTS}{30} < 2.5 \\ 6 \frac{tLOTS}{30}, & 6 \frac{tLOTS}{30} \geq 2.5 \end{cases}$$

$$(32.2.1) \quad ||tHOTS|| = \begin{cases} 2, & 6 \frac{tHOTS}{15} < 2.5 \\ 6 \frac{tHOTS}{15}, & 6 \frac{tHOTS}{15} \geq 2.5 \end{cases}$$

$$(32.3.1) \quad ||pLOTS|| = \begin{cases} 2, & 6 \frac{pLOTS}{15} < 2.5 \\ 6 \frac{pLOTS}{15}, & 6 \frac{pLOTS}{15} \geq 2.5 \end{cases}$$

$$(32.4) \quad ||pHOTS|| = pHOTS.$$

Представени с IF-THEN логика, формулите изглеждат по следния начин:

$$(32.1.2) \quad \begin{aligned} \|\mathbf{tLOTS}\|: & \text{ IF } 6 \frac{tLOTS}{30} < 2.5 \text{ THEN } \|\mathbf{tLOTS}\| \text{ IS } 2 \\ & \text{ ELSE } \|\mathbf{tLOTS}\| \text{ IS } 6 \frac{tLOTS}{30} \end{aligned}$$

$$(32.2.2) \quad \begin{aligned} \|\mathbf{tHOTS}\|: & \text{ IF } 6 \frac{tHOTS}{15} < 2.5 \text{ THEN } \|\mathbf{tHOTS}\| \text{ IS } 2 \\ & \text{ ELSE } \|\mathbf{tHOTS}\| \text{ IS } 6 \frac{tHOTS}{15} \end{aligned}$$

$$(32.3.2) \quad \begin{aligned} \|\mathbf{pLOTS}\|: & \text{ IF } 6 \frac{pLOTS}{15} < 2.5 \text{ THEN } \|\mathbf{pLOTS}\| \text{ IS } 2 \\ & \text{ ELSE } \|\mathbf{pLOTS}\| \text{ IS } 6 \frac{pLOTS}{15} \end{aligned}$$

Средно-аритметичните стойности от под-компонентите не винаги са удачно решение и затова се стига до субективна преценка от оценителя. Има множество специфични случаи, които е необходимо да се формализират, за да се опише логиката на оценителя. Някои от използваните при нашето оценяване разсъждения за поставяне оценка на компонента **практика** са следните:

- Ако rHOTS или pLOTS е 2, оценката на компонент практика е най-много 3, дори и другия под-компонент да има оценка 6.
- Ако оценката rHOTS е висока, а pLOTS – ниска, то възникват съмнения за нечестни прояви на студента при решаването на задачата, формираща оценката rHOTS. Тогава, поставяме оценката да бъде по-ниска от средно-аритметичната стойност. Това се случва най-често, когато rHOTS е близо до 6, а pLOTS – до 3.
- Ако pLOTS е висока, а rHOTS – ниска, то показаните на теста знания гарантират, че студента има необходимата основа за бъдещо развитие по преподаваната дисциплина. Въпреки, че не се е представил добре при решаването на задачата, то заслужава стимул в разумни граници. Тези граници трябва да са предварително формализирани с точни числови стойности и правила.

Подобни, разсъждения са валидни и за теоретичните знания. Тъй като тестовете, формиращи tHOTS и tLOTS обикновено се изпълняват в контролирана среда, нечестни прояви се наблюдават по-рядко. При това, високите оценки tHOTS е добре да се толерират дори и при частичната липса на теоретични знания tLOTS.

Използваните в експеримента конкретни правила за поставяне на оценка на практика и теория при **Вариант 1 за теоретико-практическото йерархично многокомпонентно оценяване** са следните:

(33.1) ***practice***($\|pLOTS\|$, $\|pHOTS\|$):

$$\begin{aligned}
 & \text{IF } \frac{\|pHOTS\| + \|pLOTS\|}{2} < 2.5 \text{ THEN } \text{practice IS } 2 \\
 & \text{ELSE IF } \|pLOTS\| = 2 \text{ OR } \|pHOTS\| = 2 \text{ THEN } \text{practice IS } 3 \\
 & \text{ELSE IF } (\|pHOTS\| - \|pLOTS\|) \\
 & \quad \geq 2 \text{ THEN } \text{practice IS } \left(\frac{\|pHOTS\| + \|pLOTS\|}{2} - 0.5 \right) \\
 & \text{ELSE } \text{practice IS } \left(\frac{\|pHOTS\| + \|pLOTS\|}{2} \right)
 \end{aligned}$$

(33.2) ***theory***($\|tLOTS\|$, $\|tHOTS\|$):

$$\begin{aligned}
 & \text{IF } \frac{\|tHOTS\| + \|tLOTS\|}{2} < 2.5 \text{ THEN } \text{theory IS } 2 \\
 & \text{ELSE IF } (\|tHOTS\| - \|tLOTS\|) \\
 & \quad \geq 2 \text{ THEN } \text{theory IS } \left(\frac{\|tHOTS\| + \|tLOTS\|}{2} + 0.3 \right) \\
 & \text{ELSE } \text{theory IS } \left(\frac{\|tHOTS\| + \|tLOTS\|}{2} \right)
 \end{aligned}$$

За определяне на **крайната оценка** при представения **Вариант 1** е приложено теорията и практиката да са равнопоставени и правилата са формулирани като:

(34) ***finalR***(*theory*, *practice*):

$$\begin{aligned}
 & \text{IF } \frac{\text{theory} + \text{practice}}{2} < 2.5 \text{ THEN } \text{finalR IS } 2 \\
 & \text{ELSE IF } \text{theory} = 2 \text{ OR } \text{practice} = 2 \text{ THEN } \text{finalR IS } 3 \\
 & \text{ELSE } \text{finalR IS } \left(\frac{\text{theory} + \text{practice}}{2} \right)
 \end{aligned}$$

Крайната оценка *finalN* се определя като се закръгли *finalR* към най-близкото цяло число:

(35) ***finalN***(*finalR*) = *ROUND*(*finalR*)

Определянето на оценките HOTS и LOTS при **Вариант 2 за HOTS-LOTS йерархично многокомпонентно оценяване** също са обект на подобни разсъждения и могат да бъдат персонализирани в зависимост от вижданията на конкретния оценител.

Формулите за оценка на компонентите **практика** и **теория** (33.1) и (33.2) са съобразени с **Модел 1**, затова участват компоненти единствено от предходното ниво. В определени гранични случаи е удачно използването на подкомпоненти от по-високо ниво и съответно използването на **Модел 2**.

Предимство на размитата логика при оценяване е възможността за постигане на точно автоматизирано оценяване според субективната логика на оценителя. Основен недостатък е трудността за неспециалисти да формализират правилата за формиране на оценката.

ИЗВОД от проведеното изследване е, че използването на размита логика гарантира получаването на коректни оценки, при коректно разписани правила, докато използването на методи на изкуствения интелект изисква достатъчно голям брой подходящи входно-изходни образци за обучение. Тук следва да отбележим, че всички автоматизирани подходи изискват последващо наблюдение и контрол от страна на преподавателя на автоматично формираните оценки.

Комплексното и обективно оценяване на знанията и уменията на обучаемите е важен елемент от образователния процес както за обучаемите, така и за преподавателите. От една страна, оценяването дава възможност на преподавателите да проследяват напредъка на обучаемите, да оценяват дали целите на обучението са изпълнени, да преценяват нуждите на обучаемите, да оценяват ефективността на преподаването и ученето. От друга страна, оценяването дава възможност на обучаемите да определят силните си страни и пропуските си, да се сравнят с колегите си, и да поемат отговорност за собствения си напредък. Предложените модели за йерархично многокомпонентно оценяване на обучаеми имат за цел комплексно оценяване на различни мисловни умения от висок и нисък порядък, теоретични знания и практически умения и др. Те позволяват използването на размита логика и могат да бъдат адаптирани за употреба при оценяване на различни дисциплини, в различни възрастови групи.

ГЛАВА 4. ОБОБЩЕНОМРЕЖОВИ МОДЕЛИ ВЪВ ВИРТУАЛНО ОБРАЗОВАТЕЛНО ПРОСТРАНСТВО

4.1. Модел на процесите на персонализиране и използване на среда за обучение

Образователните пространства трябва да предоставят възможности за: активно и интерактивно участие; работа в екип; търсене и споделяне на информация; дискутиране и представяне; произвеждане на нови знания; подпомагане дейностите на обучаемите и обучаващите; връзка с експерти и не на последно място персонализиране на обучението. Инфраструктурата на образователното пространство включва:

- *Изграждащи елементи* на пространството – идентичности, населяващи пространството. Те могат да бъдат: обучаеми, обучаващи, администратори, персонални асистенти, дигитални библиотеки, електронни услуги и т.н.

- *Взаимовръзки* – съществуващите между изграждащите елементи взаимоотношения, осигуряващи съвместната им работа при опериране в пространството.

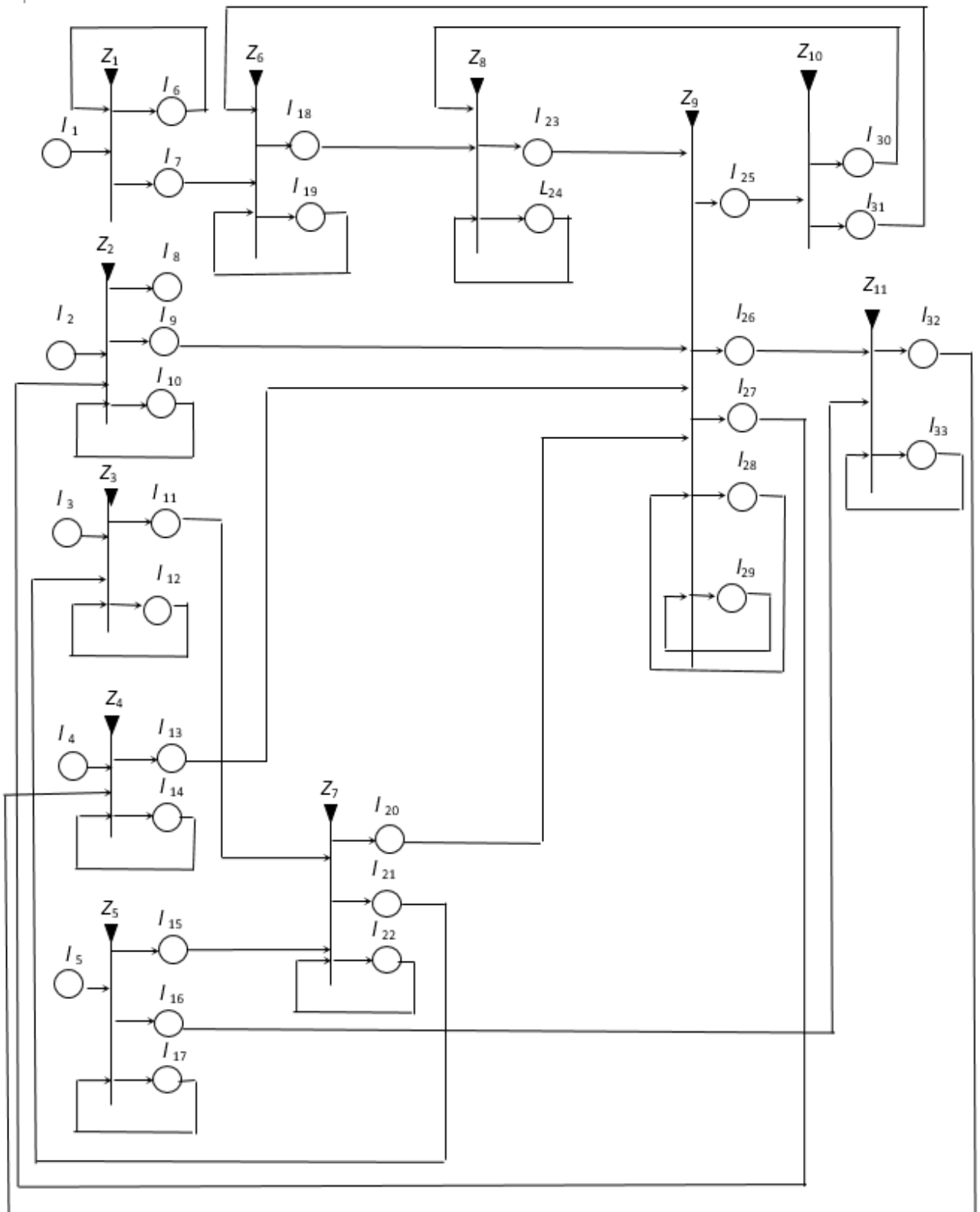
Контекстно-зависимостта характеризира поведението на отделните идентичности в пространството. Концепцията за контекстна зависимост се характеризира посредством два основни атрибута [193]:

- *Персонализация* – в смисъл на планирано адаптиране на учебното съдържание, в зависимост от обучаемите. Персонализацията може да бъде разглеждана от различни гледни точки, като основа за всяка от тях е кореспондираща класификационна схема. За целта за всеки обучаем в пространството се поддържа индивидуален потребителски профил.

- *Адаптивност* – промяна на обучаващия процес, в зависимост от възникнали промени в пространството. За постигане на това свойство се извършва моделирането на потребителя като периодично се измерват и оценяват различни индикатори, показващи напредъка на процеса на обучение и степента на усвояване на разглеждания материал. Моделът получава от околната среда информация за това, дали обучаемият е завършил определена тема, както и знанието, което се съдържа в нея. В зависимост от натрупваните данни в модела на обучаемия могат да се представят различни сценарии за достъп до образователните услуги и учебния материал.

Създадена е серия от модели, свързани с наблюдение и анализ на дейностите на обучаемите в образователна среда, представени в публикациите [144, 154, 156, 202]. Обобщеномрежови модели за работа с големи данни във виртуално колаборативно пространство са създадени в [146, 147, 155].

Тук се представя обобщен модел на процесите на персонализиране и използване на среда за обучение (фигура 39). В този модел студентите (обучаваните обекти) се интерпретират чрез σ -ядра, обучаващите се интерпретират чрез τ -ядра, обслужващият административен персонал (инспектори от учебен отдел) чрез α -ядра, обучителните курсове (SCORM-пакети) в електронната библиотека чрез λ -ядра, електронните услуги предоставяни от средата за електронно обучение чрез Υ -ядра и софтуерните агенти, в ролята на персонални асистенти на обучаемите чрез β -ядра. Този модел е минимален редуциран обобщеномрежов модел [31, 32].



Фигура 39. OM модел на процесите на персонализиране и използване на среда за обучение

Обобщеномрежовият модел се състои от 11 прехода:

Z_1 – постъпване на нов студент (обучаем) в електронната среда за обучение;

Z_2 – постъпване на обучаващ преподавател в електронната среда;

- Z₃ – обслужване на обучението от административен учебен отдел;
- Z₄ – въвеждане на нови курсове за обучение в електронната библиотека;
- Z₅ – работа с електронни услуги, предлагани от обучаващата среда;
- Z₆ – избор на курс за обучение от студент в e-Learning средата;
- Z₇ – процес на администриране на нов курс за обучение;
- Z₈ – работен процес на агента (персонален асистент на студента);
- Z₉ – процес на обучение на студента по избрания електронен курс;
- Z₁₀ – процес на оценяване на обучението на студента по избрания курс;
- Z₁₁ – процес на изграждане на учебно съдържание на електронен курс.

σ-ядрата постъпват през позиция l₁ с начална характеристика:

“име, ID (факултетен номер)”.

$$Z_1 = \langle \{l_1, l_6\}, \{l_6, l_7\}, r_1 \rangle,$$

| | | |
|----------------|------------------|------------------|
| | l ₆ | l ₇ |
| l ₁ | W _{1,6} | W _{1,7} |
| l ₆ | W _{6,6} | W _{6,7} |

където:

W_{1,6} = “неуспешна регистрация на вход в системата от студент”;

W_{1,7} = „успешна регистрация за вход в системата от студент“;

W_{6,6} = W_{1,6};

W_{6,7} = W_{1,7};

σ-ядрото постъпва в позиция l₇ с характеристика:

“име, ID, начално време на обучението”,

докато ядрото постъпва в позиция l₆ с характеристика:

“име, ID”.

$$Z_2 = \langle \{l_2, l_{27}, l_{10}\}, \{l_8, l_9, l_{10}\}, r_2 \rangle,$$

| | | | |
|-----------------|-------------------|-------------------|-----------------|
| | l ₈ | l ₉ | l ₁₀ |
| l ₂ | false | W _{2,9} | true |
| l ₂₇ | W _{27,8} | W _{27,9} | true |
| l ₁₀ | W _{10,8} | W _{10,9} | true |

където

W_{27,8} = “обучаващ напуска средата”;

W_{10,8} = W_{27,8}.

W_{27,9} = “обучаващ провежда обучение или изпитване по електронния курс”;

W_{2,9} = W_{10,9} = W_{27,9}.

Обучаващите (преподавателите) са представени в модела чрез τ-ядра, които постъпват чрез позиция l₂ с характеристика:

“име на преподавател, ID”;

едно τ -ядро цикли в позиция l_{10} с характеристика:

“налични преподаватели в обучаващата среда”,

Ядрото в позиция l_9 не получава нова характеристика.

$$Z_3 = \langle \{l_3, l_{21}, l_{12}\}, \{l_{11}, l_{12}\}, r_3 \rangle;$$

| | | |
|-------------|--------------|-------------|
| | l_{11} | l_{12} |
| $r_3 = l_3$ | $W_{3,11}$ | <i>true</i> |
| l_{22} | <i>false</i> | $W_{21,12}$ |
| l_{12} | $W_{12,11}$ | <i>true</i> |

където:

$W_{3,11}$ = “взето е решение за поставяне на нов електронен курс”;

$W_{12,11} = W_{3,11}$;

$W_{21,12}$ = “постъпила е документация за електронен курс”;

Обслужващият административен персонал (инспектори от учебен отдел) са представени в модела чрез α -ядра, които постъпват през позиция l_3 с характеристика:

“име на инспектор, ID”;

α -ядро от позиция l_{21} се слива с α -ядро от позиция l_3 и ново формираното ядро постъпва в позиция l_{11} с характеристика:

„име на инспектор, административно решение за нов електронен курс”.

Едно α -ядро цикли в позиция l_{12} с характеристика:

“База от данни с наличен административен персонал (инспектори от учебен отдел)”.

$$Z_4 = \langle \{l_4, l_{32}, l_{14}\}, \{l_{13}, l_{14}\}, r_4 \rangle$$

| | | |
|-------------|--------------|-------------|
| | l_{13} | l_{14} |
| $r_4 = l_4$ | $W_{4,13}$ | $W_{4,14}$ |
| l_{32} | <i>false</i> | <i>true</i> |
| l_{14} | <i>false</i> | <i>true</i> |

където:

$W_{4,13}$ = „генерирана е информация за нов курс”;

$W_{4,14}$ = „постъпил е нов електронен курс в електронната библиотека”;

Обучаващите курсове (SCORM пакети) в електронната библиотека са представени в модела с λ -ядра, които постъпват през позиция l_4 с характеристика:

“e-курс, план, програма, ID”;

λ -ядрото от позиция l_{13} получава характеристика:

„електронен курс, ID”.

Едно λ -ядро цикли в позиция l_{14} с характеристика:

“База от данни с налични електронни курсове (SCORM пакети)”.

$$Z_5 = \langle \{l_5, l_{17}\}, \{l_{15}, l_{16}, l_{17}\}, r_5 \rangle;$$

| | | | |
|---------|----------|-------------|-------------------------|
| | l_{11} | l_{12} | |
| $r_5 =$ | l_5 | $W_{5,15}$ | $W_{5,16}$ <i>true</i> |
| | l_{17} | $W_{17,15}$ | $W_{17,16}$ <i>true</i> |

където

$W_{5,15}$ = “използвана е електронната услуга за администриране на нов курс”;

$W_{5,16}$ = “използвана е електронната услуга за изграждане на нов курс”;

$W_{17,15} = W_{5,15}$;

$W_{17,16} = W_{5,16}$;

Електронните услуги, предоставяни от средата за електронно обучение са представени в модела чрез γ -ядра, които постъпват чрез позиция l_5 с начална характеристика:

“електронна услуга, ID”;

По време на функциониране на прехода Z_5 , γ -ядрата в позиция l_{15} получават характеристиката:

“електронна услуга, администриране на курс ID”;

докато в позиция l_{16} ядрото не получава нова характеристика.

Едно γ -ядро цикли в позиция l_{17} с характеристика:

“налични електронни услуги”.

$$Z_6 = \langle \{l_7, l_{19}, l_{31}\}, \{l_{18}, l_{19}\}, r_6 \rangle$$

| | | | |
|---------|----------|--------------|-------------|
| | l_{18} | l_{19} | |
| $r_6 =$ | l_7 | $W_{7,18}$ | <i>true</i> |
| | l_{19} | $W_{19,18}$ | <i>true</i> |
| | l_{31} | <i>false</i> | <i>true</i> |

където:

$W_{7,18}$ = “студентът започва обучение по курс в обучаващата среда”;

$W_{19,18}$ = “студентът продължава обучението по курс в обучаващата среда”;

По време на функциониране на прехода Z_6 , σ -ядрото в позиция l_{18} получава характеристиката:

“студент, ID (факултетен номер), курс за обучение”;

едно σ -ядро цикли в позиция l_{19} с характеристика:

“База от данни с обучавани студенти и техните оценки”.

$$Z_7 = \langle \{l_{11}, l_{15}, l_{22}\}, \{l_{20}, l_{21}, l_{22}\}, r_7 \rangle;$$

| | l_{20} | l_{21} | l_{22} |
|----------------|-------------|-------------|-------------|
| $r_7 = l_{11}$ | $W_{11,20}$ | $W_{11,21}$ | $true$ |
| l_{15} | $true$ | $true$ | $true$ |
| l_{22} | $true$ | $false$ | $W_{22,22}$ |

където

$W_{11,20}$ = “генерирана е заявка за създаване на учебно съдържание на електронен курс”;

$W_{11,21}$ = “създадена е документацията на нов електронен курс”.

$W_{22,22}$ = “курсът е в процес на създаване и администриране”.

По време на функциониране на прехода Z_7 α -ядрото в позиция l_{11} се слива с γ -ядро от позиция l_{15} и ново формираното ядро постъпва в позиция l_{21} с характеристика:

„документация на електронен курс (учебна програма, план, преподавателски състав) ”.

α -ядрото в позиция l_{20} получава характеристиката:

„заявка за създаване на учебно съдържание на електронен курс ”,

докато в позиция l_{22} то не получава нова характеристика.

$$Z_8 = \langle \{l_{18}, l_{24}, l_{30}\}, \{l_{23}, l_{24}\}, r_6 \rangle;$$

| | l_{23} | l_{24} |
|----------------|-------------|----------|
| $r_8 = l_{18}$ | $W_{18,23}$ | $true$ |
| l_{24} | $true$ | $true$ |
| l_{30} | $W_{30,23}$ | $false$ |

където

$W_{18,23}$ = “генериран е софтуерен агент (персонален асистент на студента).”;

$W_{30,23}$ = “ софтуерният агент продължава работата със студента в електронния курс”.

По време на функциониране на прехода Z_8 за всяко α -ядро постъпващо в позиция l_{23} се създава ново β -ядро в позиция l_{24} , представлящо софтуерния агент (персонален асистент на студента).

β -ядрото в позиция l_{24} има характеристиката:

„база от данни от софтуерни агенти за текущо обучаваните студенти ”.

$$Z_9 = \langle \{l_9, l_{13}, l_{20}, l_{23}, l_{28}, l_{29}\}, \{l_{25}, l_{26}, l_{27}, l_{28}, l_{29}\}, r_8 \rangle;$$

| $r_9 =$ | l_{25} | l_{26} | l_{27} | l_{28} | l_{29} |
|----------|-------------|-------------|------------|----------|-------------|
| l_9 | $W_{9,25}$ | $false$ | $W_{9,27}$ | $true$ | $true$ |
| l_{13} | $true$ | $W_{13,26}$ | $false$ | $true$ | $true$ |
| l_{20} | $true$ | $false$ | $false$ | $false$ | $true$ |
| l_{23} | $W_{23,25}$ | $false$ | $false$ | $false$ | $W_{23,29}$ |
| l_{28} | $false$ | $W_{28,26}$ | $false$ | $true$ | $false$ |
| l_{29} | $W_{29,25}$ | $false$ | $false$ | $false$ | $true$ |

където:

$W_{23,25} = W_{29,25}$;

$W_{23,29}$ = „обучаемият е в процес на обучение“;
 $W_{28,25}$ = “обучаващият провежда изпит”;
 $W_{28,26}$ = “генерирана е нова единица (Asset) за електронния курс”;
 $W_{28,27}$ = “обучаващият (преподавателят) завършва курса за обучение със студентите”;
 $W_{28,28}$ = $\neg W_{28,27}$;
 $W_{29,25}$ = “студент се явява на изпит”;
 $W_{29,29}$ = $\neg W_{29,25}$.

По време на функциониране на прехода Z_9 , α -ядро в позиция l_{25} получава характеристика:

“студент, ID (факултетен номер), оценка от изпита”;

λ -ядрото в позиция l_{26} получава характеристика:

“нова учебна единица (Asset) за електронния курс”;

τ -ядрото в позиция l_{27} получава характеристика:

“преподавател ID, проведен обучаващ курс”;

τ -ядрото в позиция l_{28} получава характеристика:

“преподавател ID, курс на обучение, работа по нови единици от електронния курс”;

докато σ -ядро цикли в позиция l_{29} с характеристика:

“студент, ID (факултетен номер), процес на обучение”.

$$Z_{10} = \langle \{l_{25}\}, \{l_{30}, l_{31}\}, r_{10} \rangle$$

$$r_{10} = \begin{array}{c|cc} & l_{30} & l_{31} \\ \hline l_{25} & W_{25,30} & W_{25,31} \end{array}$$

където:

$W_{25,30}$ = “студентът продължава обучението по курса (неуспешно положен изпит) и резултатът от изпита е изпратен на агента (персоналния асистент)”;

$W_{25,31}$ = “резултатът от изпита е изпратен на студента”.

По време на функциониране на прехода Z_{10} :

σ -ядрото в позиция l_{31} получава характеристиката:

“студент ID, резултат от изпита - оценка”;

σ -ядрото в позиция l_{30} получава характеристиката:

“студент ID, слаб резултат от изпита”.

$$Z_{11} = \langle \{l_{26}, l_{16}, l_{33}\}, \{l_{32}, l_{33}\}, r_{11} \rangle$$

$$r_{11} = \begin{array}{c|cc} & l_{32} & l_{33} \\ \hline l_{26} & false & true \\ l_{16} & true & true \\ l_{33} & W_{33,32} & W_{33,33} \end{array}$$

където:

$W_{33,32}$ = “генериран е нов SCO обект за SCORM пакета на електронния курс”;

$W_{33,33}$ = “генериран е нов Asset за SCO обект от SCORM пакета на електронния курс”.

По време на функциониране на прехода Z_{11} λ -ядрото в позиция I_{32} получава характеристиката:

“new SCO обект за SCORM пакета”,

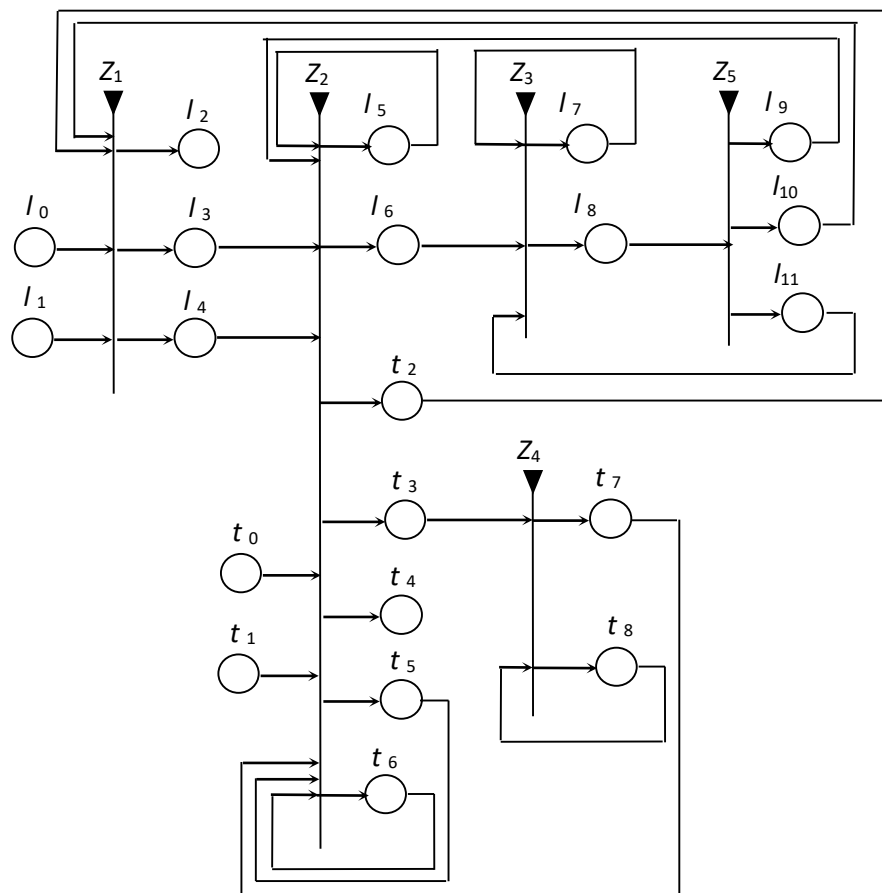
докато λ -ядрото в позиция I_{33} получава характеристиката:

“нов Asset от създавания SCO обект за SCORM пакета на електронния курс”.

ИЗВОД: Създаденият обобщеномрежов модел [152], дава възможност за проследяване на процесите на персонализация и използване на различни интелигентни инструменти за електронно обучение. Може да се извлече информация за предпочитанията на обучаемите и резултатите от тяхното обучение. На базата на модела и натрупана статистика от реални данни могат да се правят оценки и да се откриват тенденции за развитието на процесите, свързани с електронното обучение и неговото обслужване. Могат да бъдат въведени допълнителни параметри на модела и допълнителни характеристики на ядрата, като се вземат предвид факторите, влияещи върху процеса на обучение, с цел повишаване на интереса на обучаемите. Активното участие на обучаемите в процеса на усвояване на знания и придобиване на умения в голяма степен може да бъде повлияно от качеството на използваното образователно пространство.

4.2. Модел на процеса на прилагане средства за извличане на знания от данните в среди за обучение чрез апарата на обобщените мрежи

Създаденият обобщеномрежов модел описва възможности за избор и прилагане на подходящи техники за извличане на знания от данните в средите за обучение [143]. Този модел, представен на фиг. 40, е редуциран обобщеномрежов модел, но от по-висок клас спрямо модела, представен на фиг. 39. Тук за всеки преход е дефинирано условие за активиране, като се задава чрез булев израз. Ако стойността му е *“true”*, съответният преход може да се активира, ако е *“false”* – не. Моделът съдържа 5 прехода и 21 позиции. Позициите са два типа и са свързани с два типа ядра, които постъпват в съответните типове позиции: α -ядра и l -позиции представят процеса на извличане на знания от данните, β -ядра и t -позиции представят критериите за ограничаване на средствата и избора на подходящи техники за извличане на знания от данните. За краткост се използва означението α - и β -ядра вместо α_i - и β_j -ядра, където i, j са номерата на съответните ядра и само влизащите в обобщената мрежа ядра са номерирани с α_0, α_1 и β_1, β_2 .



Фигура 40. Модел на процесите на избор и прилагане на средства за извличане на знания от данните в обучаваща среда

В позиция t_6 стои β_0 -ядро с начална характеристика:

„налични средства за извличане на знания от данните“.

На следващия преход от функционирането на мрежата, β -ядрото се разделя на две ядра. Оригиналото β_0 -ядро остава в позиция t_6 , докато другото β -ядро преминавайки през преход Z_3 , се придвижва към преход Z_5 .

През позиции l_0 и l_1 в мрежата постъпват ядрата α_0 и α_1 , с начални характеристики съответно:

„начални хипотези“;

„начални данни“.

Ядрата β_1 и β_2 постъпват в мрежата през позиции t_0 и t_1 . и получават начални характеристики съответно:

„нова техника за извличане на знания от данните“;

„критерии за избор на техника за извличане на знания от данните“.

Следва описание на отделните преходи.

| | | | | |
|----------|--------------|--------------|--------------|--|
| | l_2 | l_3 | l_4 | |
| l_0 | <i>false</i> | <i>false</i> | <i>true</i> | |
| l_1 | <i>false</i> | <i>false</i> | <i>true</i> | $\vee(\wedge(l_0, l_1), l_{10}, t_2)>$, |
| l_{10} | $W_{10,2}$ | $W_{10,3}$ | <i>false</i> | |
| t_2 | <i>false</i> | <i>true</i> | <i>false</i> | |

където:

$W_{10,2}$ = “приложена е техниката за извличане на знания от данните”,

$W_{10,3}$ = $\neg W_{10,2}$,

където $\neg P$ означава отрицание на предиката P .

По време на прехода Z_1 α_0 и α_1 -ядрата, постъпващи от позиции l_0 и l_1 в позиция l_4 се сливат в едно ново α -ядро с характеристика:

“Начални хипотези, начални данни”.

При следващото активиране на прехода Z_1 β -ядрото, постъпващо от позиция t_2 в позиция l_3 получава характеристиката:

“Цел, техника за извличане на знания от данните”.

$Z_2 = \langle \{l_3, l_4, l_5, l_9, t_0, t_1, t_5, t_6, t_7\}, \{l_5, l_6, t_2, t_3, t_4, t_5, t_6\},$

| | | | | | | | |
|-------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | l_5 | l_6 | t_2 | t_3 | t_4 | t_5 | t_6 |
| l_3 | <i>true</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> |
| l_4 | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>true</i> |
| l_5 | $W_{5,5}$ | $W_{5,6}$ | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> |
| l_9 | <i>true</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> |
| t_0 | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>true</i> |
| t_5 | <i>false</i> | <i>false</i> | <i>false</i> | <i>true</i> | <i>false</i> | <i>false</i> | <i>false</i> |
| t_6 | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> | $W_{6,4}$ | $W_{6,5}$ | <i>true</i> |
| t_7 | <i>false</i> | <i>false</i> | <i>true</i> | <i>false</i> | <i>false</i> | <i>false</i> | <i>false</i> |
| t_1 | <i>false</i> | <i>false</i> | <i>false</i> | <i>true</i> | <i>false</i> | <i>false</i> | <i>false</i> |

$\vee(l_3, l_4, l_5, l_9, t_0, \wedge(t_1, t_5), t_6, t_7)>$,

където

$W_{5,5}$ = “има техника за извличане на знания от данните, която не е прилагана”,

$W_{5,6}$ = ”техниката за извличане на знания от данните е вече е приложена”,

$W_{6,4}$ = ”техниката за извличане на знания от данните е отхвърлена/неподходяща”,

$W_{6,5}$ = ”техника за извличане на знания от данните е избрана”.

α -ядрата получават характеристики в позиция l_3 и в позиция l_4 съответно:

*“Избрани техники за извличане на знания от данните,
критерии за избор на data mining техники”,
“Отхвърлена техника за извличане на знания от данните”.*

β_1 -ядрото, постъпващо в прехода Z_2 от позиция t_0 се слива с оригиналното β_0 -ядро, което стои в позиция t_6 .

β_3 -ядрото, постъпващо в позиция t_5 получава характеристиката:

“Избрани техники за извличане на знания от данните”.

β -ядрата, които постъпват в позиции t_2 и t_3 не получават нова характеристика.

$$Z_3 = \langle \{l_6, l_7, l_{11}\}, \{l_7, l_8\}, \begin{array}{c|cc} & l_7 & l_8 \\ \hline l_6 & true & false \\ l_7 & W_{7,7} & W_{7,8} \\ l_{11} & true & false \end{array}, \vee(l_6, l_7, l_{11}) \rangle,$$

където:

$W_{7,7}$ = “има следващи стъпки от работата на текущата техника за извличане на знания”,

$W_{7,8}$ = “избрана е следващата стъпка от работата на текущата техника за извличане на знания”

α -ядрото, което постъпва в позиция l_7 не получава нова характеристика, докато α -ядрото, което постъпва в позиция l_8 получава характеристиката:

“Текуща стъпка от избраната техника за извличане на знания”

$$Z_4 = \langle \{t_3, t_8\}, \{t_7, t_8\}, \begin{array}{c|cc} & t_7 & t_8 \\ \hline t_3 & false & true \\ t_8 & W_{8,7} & W_{8,8} \end{array} \rangle, \vee(t_3, t_8)$$

$W_{8,7}$ = избрана е техника за поставената цел”,

$W_{8,8}$ = $\neg W_{8,7}$.

β -ядрата, които постъпват в позиция t_8 не получават нова характеристика, докато β -ядрата, които постъпват в позиция t_7 получават характеристиката:

“Цел, избрани техники за извличане на знания от данните”.

$$Z_5 = \langle \{l_8\}, \{l_9, l_{10}, l_{11}\}, \begin{array}{c|ccc} & l_9 & l_{10} & l_{11} \\ \hline l_8 & W_{8,9} & W_{8,10} & W_{8,11} \end{array} \rangle,$$

$W_{8,9}$ = “има друга техника за извличане на знания, която ще бъде приложена”,

$W_{8,10}$ = “последната възможна техника за извличане на знания е приложена”,

$W_{8,11}$ = “има следваща стъпка от текущата техника за извличане на знания от данните”.

α -ядрата, които постъпват в позиции l_9 и l_{11} не получават нови характеристики, докато α -ядрата, постъпващи в позиция l_{10} получават характеристиката:

“Цел, средство за извличане на знания от данните, оценка от работата”.

ИЗВОД: Интегрирането на системите за обучение със средства за извличане на знания от данните е необходимо за процесите, свързани с персонализиране на курсове за електронно и дистанционно обучение. На базата на получените резултати могат бъдат въведени допълнителни мерки за анализ и промяна на обучаващите курсове както и нови критерии за анализ. Това от своя страна е път към повишаване на качеството на обучението във висшето училище.

4.3. Обобщеномрежов модел на процес за многокомпонентно оценяване

Обективното оценяване на знанията и уменията на обучаемите има множество характеристики: истински резултат (*true score*), надеждност (*reliability*) и валидност (*validity*), честност, диференцираност, всестранност на оценката и др. [187]. То предоставя на преподавателите по-добра възможност както за оценка на усвояването на знанията от обучаемите, така и за анализ на ефективността на преподаването.

Създаден е обобщеномрежов *модел на процес за многокомпонентно оценяване*, който включва шест етапа:

1. **Създаване на мета-модел на тест.** Мета-моделът на тест се дефинира чрез типове оценъчни компоненти и модели за оценяване. Типовете оценъчни компоненти определят какви знания и умения се оценяват, например според конкретна таксономия или създадена от обучаващия класификация, подходяща за специфичен домейн. Моделите за оценяване могат да са основани на различни формули с различни тегла за оценъчните компоненти, на размита логика, и др.
2. **Създаване на модел на тест.** Моделът на тест включва характеристики за целевата група, предмета на обучение, началото и продължителността на теста, и др. На този етап, за всеки един от типовете оценъчни компоненти на мета-модела се задават съответен брой тестови единици, които ще съдържат инстанциите на тестовете.
3. **Конфигуриране на тест.** Тук се задават конкретни тестови единици. Те се въвеждат или избират от база данни с въпроси. За всяка конкретна тестова единица трябва да се укаже към кой тип оценъчни елементи принадлежи. На този етап се задават и конкретните обучаеми, които правят теста.
4. **Провеждане на теста** е процес, при който се генерират конкретни тестове. Те трябва да включват по определен брой тестови единици от всеки тип оценъчни компоненти – така, както е зададено в мета-модела и модела. Обучаемите въвеждат своите отговори и решения.
5. **Процес на оценяване.** Включва автоматично оценяване на теста според избрания модел и ръчно оценяване на отворените въпроси и задачи, ако има такива.
6. **Анализ на резултатите и оценяване на теста.** Създателят на теста използва получените резултати, за да оцени теста и ефективността на приложените подходи за оценяване.

Създаденият обобщеномрежов модел описва процеса на моделиране и конструиране на тестове за многокомпонентно оценяване на знанията и уменията на обучаемите [102]. Този модел е минимален редуциран обобщеномрежов модел и съдържа множеството от преходи:

$$A = \{Z_1, Z_2, Z_3, Z_4, Z_5, Z_6\},$$

където:

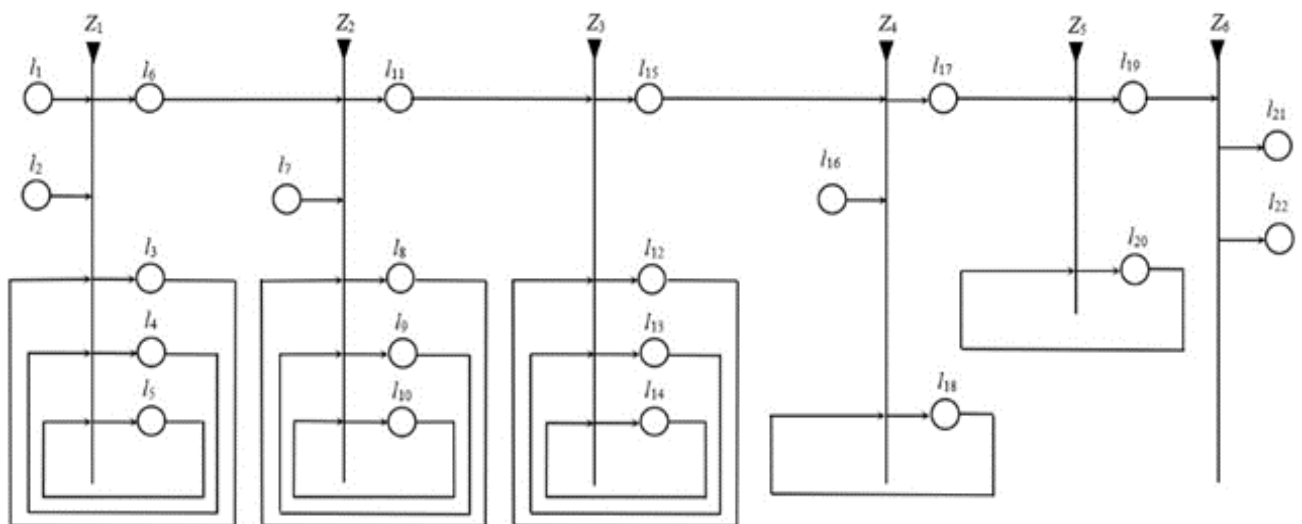
- Z_1 – създаване на мета-модел на тест;
- Z_2 – създаване на модел на тест;
- Z_3 – конфигуриране на тест;
- Z_4 – провеждане на тест;

Z_5 – процес на оценяване;

Z_6 – анализ на резултатите и оценяване на теста.

Обобщеномрежовият модел е представен на фиг. 41. За описание на процесите в модела са използвани следните ядра:

- α -ядра интерпретират потребителите – администратори, създатели на тестове, автори на въпроси, обучаеми;
- β - ядра – типове оценъчни компоненти, за оценяване на практически или теоретични умения, умения за мислене от нисък или висок порядък, умения, категоризирани според конкретна таксономия за обучение или др.;
- γ - ядра интерпретират модели за оценяване – среднопретеглена оценка на оценъчните компоненти, оценка, базирана на теглови коефициенти, размита логика или др.;
- δ - ядро – мета-модел на тест;
- η - ядро – модел на тест;
- λ - ядро – банка от данни с тестови въпроси;
- μ - ядро – оценъчен компонент (например: тестов въпрос, задача, казус или др.).



Фигура 41. Обобщеномрежов модел на процес за многокомпонентно оценяване

Процесът се стартира от администратор на системата, представен чрез α -ядро, който постъпва в мрежата през позиция l_1 с начална характеристика:

„уникален идентификатор и име на потребител на софтуерната система“.

В позиция l_2 се стартира заявка за създаване на нов мета-модел на тест, чрез δ -ядро с начална характеристика:

„уникален идентификатор и наименование на мета-модел на тест“.

В позиция l_3 цикли α -ядро с характеристика:

„списък с налични потребители на софтуерната система“.

В позиция l_4 и в позиция l_5 циклят съответно β -ядро и γ -ядро с начални характеристики:

„типове оценъчни компоненти“ и

„налични модели за оценяване“.

$$Z_1 = \langle \{ l_1, l_2, l_3, l_4, l_5 \}, \{ l_3, l_4, l_5, l_6 \}, r_1 \rangle,$$

| $r_1 =$ | l_3 | l_4 | l_5 | l_6 |
|---------|--------------|--------------|--------------|-----------|
| l_1 | <i>false</i> | <i>false</i> | <i>false</i> | $W_{1,6}$ |
| l_2 | <i>false</i> | <i>false</i> | <i>false</i> | $W_{2,6}$ |
| l_3 | <i>true</i> | <i>false</i> | <i>false</i> | $W_{3,6}$ |
| l_4 | <i>false</i> | <i>true</i> | <i>false</i> | $W_{4,6}$ |
| l_5 | <i>false</i> | <i>false</i> | <i>true</i> | $W_{5,6}$ |

където:

$W_{1,6}$ = „потребител е оторизиран в системата като администратор“;

$W_{2,6}$ = „инициализиран е нов мета-модел на тест“;

$W_{3,6}$ = „избран е създател на тест“;

$W_{4,6}$ = „зададен е списък с типове оценъчни компоненти, характерни за мета-модела на теста“;

$W_{5,6}$ = „зададени са модели за оценяване“.

По време на активиране на прехода Z_1 , α -ядрото от позиция l_3 се разделя на две ядра. Едното остава в позиция l_3 , а другото ядро се слива с δ -ядрото от позиция l_2 и постъпва в позиция l_6 като получава характеристика:

„администратор, уникален идентификатор на мета-модел на тест, списък с типове оценъчни компоненти, списък с модели за оценяване, създател на тест“.

Преход Z_2 представя процеса на създаването на модел на тест. От входната позиция l_6 постъпва δ -ядро и се слива с налично δ -ядро в позиция l_9 .

Във входна позиция l_7 постъпва η -ядро, което интерпретира стартиране на заявка за създаване на нов модел на тест, с начална характеристика:

„заявка за създаване на нов модел на тест“.

В позиция l_8 , постъпва η -ядрото с характеристика:

„уникален идентификатор и наименование на модел на тест, дисциплина, целева група, продължителност на теста, начало на теста, списък със създатели на тестови въпроси“.

В позиция l_9 постъпва β -ядро с характеристика:

„уникален идентификатор на тест, с брой оценъчни компоненти“,

В позиция l_{10} цикли δ -ядро с характеристика:

„списък мета-модели на тестове“.

$$Z_2 = \langle \{ l_6, l_7, l_8, l_9, l_{10} \}, \{ l_8, l_9, l_{10}, l_{11} \}, r_2 \rangle,$$

| $r_2 =$ | l_8 | l_9 | l_{10} | l_{11} |
|----------|--------------|--------------|--------------|-------------|
| l_6 | <i>false</i> | <i>false</i> | <i>false</i> | $W_{6,11}$ |
| l_7 | <i>false</i> | <i>false</i> | <i>false</i> | $W_{7,11}$ |
| l_8 | <i>true</i> | <i>false</i> | <i>false</i> | $W_{8,11}$ |
| l_9 | <i>false</i> | <i>true</i> | <i>false</i> | $W_{9,11}$ |
| l_{10} | <i>false</i> | <i>false</i> | <i>true</i> | $W_{10,11}$ |

където:

$W_{6,11}$ = „избрани са данните по подразбиране за мета-модел (администратор, уникален идентификатор на мета-модел на тест, списък с типове оценъчни компоненти, списък с модели за оценяване, създател на тест)“;

$W_{7,11}$ = „инициирана е заявка за създаване на модел на тест и е зададен идентификационен номер на тест“;

$W_{8,11}$ = „зададени са характеристики на модела на теста (дисциплина, целева група, продължителност на теста, начало на теста, списък със създатели на тестови въпроси)“;

$W_{9,11}$ = „за всеки тип компонент е определен съответен брой оценъчни компоненти“;

$W_{10,11}$ = „избран е мета-модел на тест“.

При активиране на прехода Z_2 , η -ядрото преминава от позиция l_7 в позиция l_8 като при следващо активиране на прехода η -ядро от позиция l_8 се слива с δ -ядро от позиция l_9 и постъпва в позиция l_{11} , с характеристика:

„уникален идентификатор на модел на тест, дисциплина, целева група, продължителност на теста, начало на теста, списък със създатели на тестови въпроси, уникален идентификатор на мета-модел на тест, списък с типове оценъчни компоненти, списък с модели за оценяване, списък с брой оценъчни компоненти“.

В преход Z_3 се представя процеса по конфигуриране на теста. В позиция l_{12} цикли λ -ядро с характеристика:

„банка с налични тестови въпроси (оценъчни компоненти)“.

В позиция l_{13} постъпва μ -ядро с характеристика:

„избран оценъчен компонент за теста“.

В позиция l_{14} цикли α -ядро с характеристика:

„списък с обучаеми“.

$Z_3 = \langle \{ l_{11}, l_{12}, l_{13}, l_{14} \}, \{ l_{12}, l_{13}, l_{14}, l_{15} \}, r_3 \rangle,$

| | | | | | |
|---------|----------|--------------|--------------|--------------|--------------|
| $r_3 =$ | | l_{12} | l_{13} | l_{14} | l_{15} |
| | l_{11} | <i>true</i> | <i>true</i> | <i>true</i> | <i>false</i> |
| | l_{12} | <i>true</i> | <i>false</i> | <i>false</i> | $W_{12,15}$ |
| | l_{13} | <i>false</i> | <i>true</i> | <i>false</i> | $W_{13,15}$ |
| | l_{14} | <i>false</i> | <i>false</i> | <i>true</i> | $W_{14,15}$ |

където:

$W_{12,15}$ = „тестови въпрос от банката с данни е добавен към теста“;

$W_{13,15}$ = „създаден е нов тестови въпрос и е добавен към теста“;

$W_{14,15}$ = „избрани са обучаемите, които ще изпълняват теста“.

При активиране на преход Z_3 , η -ядро от позиция l_{11} се слива с μ - ядро от позиция l_{13} и α - ядро от позиция l_{14} , и постъпва в позиция l_{15} със следната характеристика:

„уникален идентификатор на модел на тест, дисциплина, целева група, продължителност на теста, начало на теста, списък със създатели на тестови въпроси, уникален идентификатор на мета-модел на тест, списък с типове оценъчни

компоненти, списък с модели за оценяване, списък с брой оценъчни компоненти, списък с тестови въпроси, списък с обучаеми“.

$$Z_4 = \langle \{ l_{15}, l_{16}, l_{18} \}, \{ l_{17}, l_{18} \}, r_4 \rangle,$$

$$r_4 = \begin{array}{c|cc} & l_{17} & l_{18} \\ \hline l_{15} & false & true \\ l_{16} & false & W_{16,18} \\ l_{18} & W_{18,17} & W_{18,18} \end{array}$$

където:

$W_{16,18}$ = „времето за стартиране на теста е настъпило“;

$W_{18,17}$ = „даден е отговор на тестови въпрос и теста е завършен“.

$W_{18,18}$ = „даден е отговор на тестови въпрос, теста не е завършен“.

При активиране на преход Z_4 от начална позиция l_{16} постъпва μ -ядро с начална характеристика:

„време, в което се активира теста за изпълнение“.

μ -ядрото от позиция l_{16} се слива с η -ядро от позиция l_{18} и постъпва в позиция l_{17} с характеристика:

„уникален идентификатор на модел на тест, дисциплина, целева група, продължителност на теста, начало на теста, списък със създатели на тестови въпроси, уникален идентификатор на мета-модел на тест, списък с типове оценъчни компоненти, списък с модели за оценяване, списък с брой оценъчни компоненти, списък с тестови въпроси, попълнени тестови въпроси, списък с обучаеми“.

$$Z_5 = \langle \{ l_{17}, l_{20} \}, \{ l_{19}, l_{20} \}, r_5 \rangle,$$

$$r_5 = \begin{array}{c|cc} & l_{19} & l_{20} \\ \hline l_{17} & W_{17,19} & true \\ l_{20} & W_{20,19} & W_{20,20} \end{array}$$

където:

$W_{17,19}$ = „в теста няма отворени въпроси за оценяване“;

$W_{20,19}$ = „поставена е оценка на всички тестови въпроси“;

$W_{20,20}$ = „даден е отговор на тестови въпрос, но теста не е завършен“.

В преход Z_5 се представя ръчно оценяване на отворените тестови въпроси. η -ядро от позиция l_{17} се слива с μ -ядро от позиция l_{20} и постъпва в позиция l_{19} характеристика:

„уникален идентификатор на модел на тест, дисциплина, целева група, продължителност на теста, начало на теста, списък със създатели на тестови въпроси, уникален идентификатор на мета-модел на тест, списък с типове оценъчни компоненти, списък с модели за оценяване, списък с брой оценъчни компоненти, списък с тестови въпроси, попълнени тестови въпроси, оценки на тестовите въпроси, списък с обучаеми“.

Предложеният метод за йерархично многокомпонентно оценяване на обучаемите, (в точка 3.1.6.) може да се интерпретира в този преход.

$$Z_6 = \langle \{ l_{19} \}, \{ l_{21}, l_{22} \}, r_6 \rangle,$$

$$r_6 = \begin{array}{c|cc} & l_{21} & l_{22} \\ \hline l_{19} & W_{19,21} & W_{19,22} \end{array}$$

където:

$W_{19,21}$ = „направен е анализ на резултатите“;

$W_{19,22}$ = „поставена е оценка на теста“.

След постъпване на η -ядро в позиция l_{19} се активира преход Z_6 . В позиция l_{21} ядрото получава характеристика:

„анализ на резултатите от теста“,

а в позиция l_{22} получава характеристика:

„оценка на теста“.

ИЗВОД: Предложен е обобщеномрежов модел, който има за цел от една страна да обобщи процеса на оценяване, а от друга – да даде възможност за персонализация на начина за формиране на тестове и оценяване на обучаемите по време на обучение [102]. Това е постигнато чрез дефиниране на мета-модел и модели за оценяване, които задават рамки за създаване на конкретни тестове и подходи за оценяването им. Разработването на софтуерна система, имплементираща описания процес, ще предостави на преподавателите гъвкава платформа за експериментиране с различни стандартни и авторски подходи за конструиране на тестове и оценяване на обучаемите.

4.4. Обобщеномрежов модел на процесите в проектно-базирано обучение

Проектно-базираното обучение предоставя огромни възможности за обогатяване на традиционните педагогически подходи за усвояване на знанията. Чрез поставяне и разработване на проекти към курсовете на обучение могат да се следват модерните технологии и динамиката на развитието им. Това предполага внимателен подбор на теми и съдържание на курсови проекти, както и тяхното развитие при изучаване на нови дисциплини.

По своята същност проектно-базираното обучение е педагогически модел на междупредметни дейности, насочени към реално съществуваща проблематика [42]. Това е предизвикателство към обучаемите към конструиране, усвояване и затвърдяване на знания и умения от по-високо ниво. Основни умения, които обучаемите формират са: да се научат да идентифицират етапите в разработването на проект, да планират дейността си, да спазват планираните срокове, да работят съвместно с други членове на екипа, да оценяват дейността на другите членове на екипа, да самооценяват действията си, да участват в дискусиите по темата на проекта, като формират и защитават аргументирано собствени идеи и умения.

Процес на управление на работата по проекти в обучението

Успешното управление зависи от ръководителя на дисциплината: той може да разпределя задачите за изпълнение, да създава график за работата, да контролира изпълнението на подзадачите и да следи цялостната работа. За целта могат периодично да се правят срещи за представяне на текущи резултати и обсъждане на дейностите или да се работи в подходяща интерактивна среда като например Moodle, Acolad или среда на база Wiki технологията.

(1) Основните дейности при дефиниране на проекти

Подготовката и планирането на проектите е свързано с определяне на етапите на изпълнение, подзадачите, сроковете, информационните ресурси, и др. Тези дейности могат да се формализират като:

- *Спецификация на задачата за конкретен проект.* Определят се целите и задачите, както и фазите, през които ще премине разработката на текущия проект.

- *Дефиниране на подзадачите.* Необходимо е да се дефинират всички задачи (основни и подзадачи), да се определи времето, за което трябва да се приключи дадена задача и кой ще се заеме с нея.

- *Дефиниране и разпределяне на ресурсите,* нужни за проекта. Под ресурси трябва да се разбират човешки, софтуерни, хардуерни, както и фактора време.

- *Управление на изпълнението на задачите.* Вече дефинираните задачи трябва да бъдат управлявани при изпълнението им, т.е. да се следи развитието на проекта, да се представят приключените подзадачи, задачите, които са в процес на работа и задачите, които предстоят.

- *Събиране на различни данни за статистика и за измерване на развитието на проекта.* Статистиката, която се събира, трябва да служи като основа за измерване на други величини, определящи самия проект – качество, сложност, големина и др.

- *Описание на възможните рискове за проекта.* Ръководителят на проекта описва всеки възможен риск за проекта, разработва план за избягване на риска и управлението му, ако това стане необходимо.

- *Генериране на различни справки,* за работата по проекта.

(2) Планиране на екипи (групи)

Основни дейности, свързани по планирането на групите, които ще работят по проектите са:

- *Определяне на броя на участниците в екип по даден проект.* Това зависи от обема на проекта и евентуалните задачи за изпълнение.

- *Определяне на структурата на екипа.* Тя може да се формира според интересите на участниците, според техните възможности или по друг признак.

- *Определяне на ръководител на екипа,* който може да бъде определен от преподавателя или да бъде избран от участниците в екипа на проекта.

(3) Поставяне на задачата пред студентите

При дефинирането на проекта последователно трябва да бъдат представени на студентите:

- предварителна информация;
- задачата, поставена интересно и мотивиращо;
- постъпково описание на процеса при изпълнение на задачата;
- набор от препоръчителни информационни източници;
- насоки за организиране и съхраняване на информацията;
- начини и критерии за оценяване на изпълнението на задачата.

(4) Дейности на обучаемите, изпълняващи проекти

При изпълнението на проект, обучаемите: обмислят поставения въпрос или задача и могат да го конкретизират или обобщават; събират и анализират данни от различни източници; споделят и дискутират идеи; правят предположения, хипотези и предвиждания; провеждат експерименти; създават реферати, бази от данни, модели, прототипи; правят доказателства, обобщения и изводи; защитават идеите си пред колеги; поставят въпроси и обсъждат проблеми.

(5) Проверка и оценяване

Планирането на средствата за проверяване и оценяване на резултатите от работата по проекти включва:

- определяне какво точно ще се проверява: постигане на поставените цели, спазване на сроковете, качество на разработения продукт и други показатели;

- определяне чрез какви средствата ще се извършва проверката: текуща проверка (наблюдение, попълване на чек-листове), финална проверка и оценка на готовите продукти, работа на екипа и отделните му членове, представяне и защита на проекта;

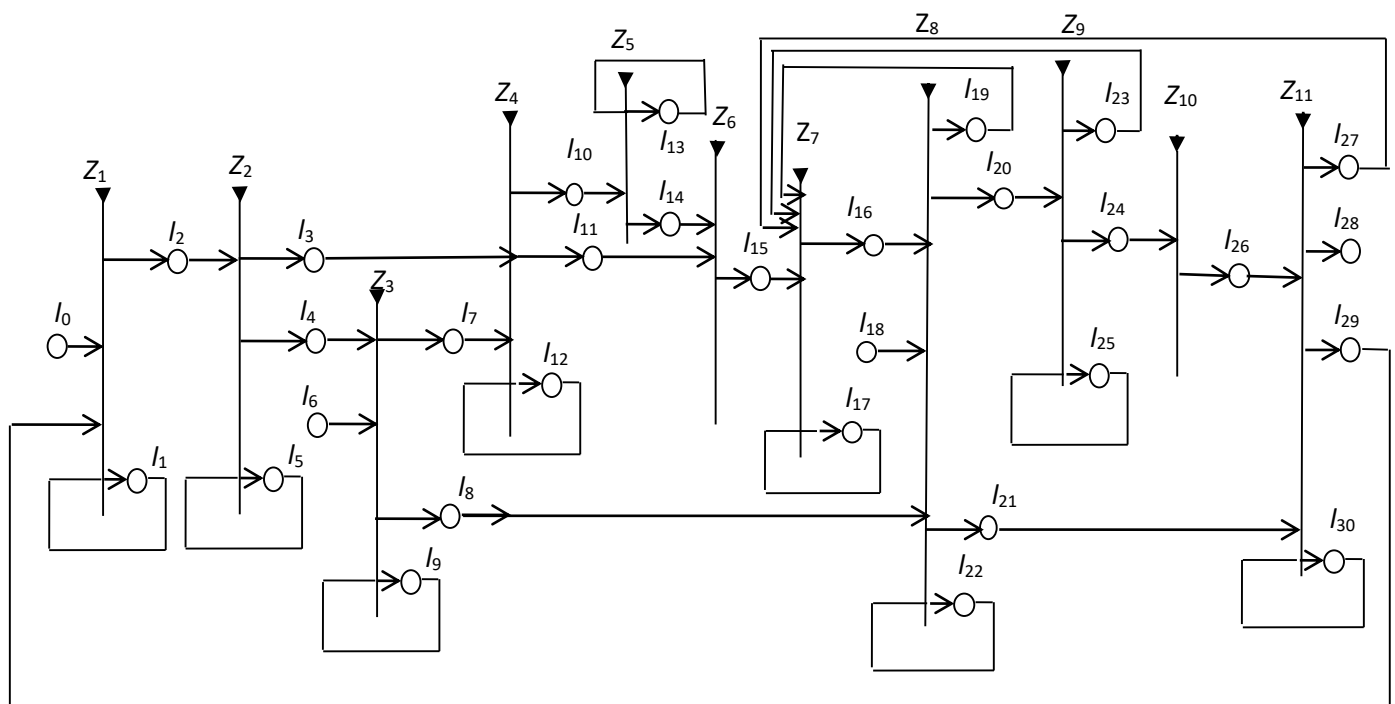
- цитирана библиография (литература, интернет ресурси, помощни материали, ръководства и др.), които са използвани от студентите в работата им през различните етапи;

- публичната изява като естествен завършек за разработвания проект, която трябва да бъде стимулирана и подкрепяна за публикуване и представяне на различни студентски форуми.

Ако преподавателят използва хранилище с данни, в които е събрана информация за минали проекти, то след приключване на работата по текущите проекти, базата от данни трябва да бъде обновена. Предимство на хранилището с разработени вече проекти е, че проверяващият може да сравнява различни проекти и да бъдат открити взаимствани (чужди) разработки. Освен това натрупваната информация може да се използва като база за набирание на статистически данни за уменията на студентите и за обучаваната група като цяло. Могат да се оценяват степента на развитие и грешките, които студентите допускат, и да се акцентира на тях в следващи занятия.

Моделиране на процеса на обучение чрез работа по проекти

В обобщеномрежвия модел на фиг. 42, задачите (проектите) се интерпретират от α -ядра, преподавателите са интерпретирани от β -ядра, обучаемите (обектите на обучение) от γ -ядра, критериите за оценка съответно чрез δ -ядра [149]. Този модел е минимален редуциран обобщеномрежов модел.



Фигура 42. Обобщеномрежов модел на процесите в проектно-базирано обучение

Моделът се състои от 11 прехода със следното значение:

- Z_1 = Подготовка за проектно-базираното обучение
- Z_2 = Стартиране конкретно проектно-базираното обучение
- Z_3 = Формиране на екипи от студенти
- Z_4 = Определяне на задачите за всеки екип

- Z_5 = Дефиниране на подзадачите на всеки проект
- Z_6 = Разпределение на подзадачите по участниците в екипа
- Z_7 = Работа по конкретна подзадача на проекта
- Z_8 = Процес на работа по цялостния проект
- Z_9 = Завършване на работата по проекта
- Z_{10} = Представяне и защита на проекта
- Z_{11} = Оценяване на проектите

В началото β -ядро постъпва в мрежата през позиция l_0 с начална характеристика:
„преподавател, ID”.

$$Z_1 = \langle \{l_0, l_1, l_{29}\}, \{l_1, l_2\}, r_1 \rangle,$$

| | | | |
|-------------|--|-------------|--------------|
| | | l_1 | l_2 |
| $r_1 = l_0$ | | <i>true</i> | <i>false</i> |
| l_1 | | <i>true</i> | $W_{1,2}$ |
| l_{29} | | <i>true</i> | <i>false</i> |

където предикатът $W_{1,2}$ има следното значение:

$W_{1,2}$ = “взето е решение от преподавател за провеждане на проектно-базирано обучение”;

Постъпващото β -ядро влиза в позиция l_1 и се слива с β -ядрото, което постоянно стои там с характеристика:

“списък с преподаватели”

При решение на преподавател да прилага проектно-базирано обучение, едно β -ядро от позиция l_1 постъпва в позиция l_2 и получава характеристиката:

“преподавател, ID, предмет на обучение”.

$$Z_2 = \langle \{l_2, l_5\}, \{l_3, l_4, l_5\}, r_2 \rangle,$$

| | | | | |
|-------------|--|--------------|--------------|-------------|
| | | l_3 | l_4 | l_5 |
| $r_2 = l_2$ | | <i>false</i> | <i>false</i> | <i>true</i> |
| l_5 | | $W_{5,3}$ | $W_{5,4}$ | <i>true</i> |

където:

$W_{5,3}$ = “формулирана е проектна задача”;

$W_{5,4}$ = “стартирано е проектно-базирано обучение от преподавател”.

В позиция l_3 постъпва едно α -ядро с характеристика:

„спецификация на проектната задача”.

β ядрото от позиция l_5 има характеристика:

„списък с преподаватели, работещи по обучение чрез проекти”.

При активиране на прехода β ядрото от позиция l_5 се разцепва на две ядра, едното остава в позиция l_5 , а другото преминава в позиция l_4 с характеристика

„преподавател-ръководител на проектно обучение”.

$$Z_3 = \langle \{l_4, l_6, l_9\}, \{l_7, l_8, l_9\}, r_3 \rangle,$$

| | l_7 | l_8 | l_9 |
|-------|-----------|-------|-------|
| l_4 | false | true | false |
| l_6 | false | false | true |
| l_9 | $W_{9,7}$ | false | true |

където:

$W_{9,7}$ = “избран е студент към екип”.

Чрез входната позиция l_6 постъпва γ ядро с начална характеристика:

„студент - участник в обучението, ID”.

В позиция l_9 стои γ ядро с характеристика:

„списък на студентите, участници в проектното обучение”.

При активиране на прехода в позиция l_7 постъпва γ ядро с характеристика:

„студент ID, номер на екип”.

β ядрото от позиция l_4 преминава в позиция l_8 като запазва характеристиката си:

„преподавател-ръководител на проектно обучение”.

$$Z_4 = \langle \{l_3, l_7, l_{12}\}, \{l_{10}, l_{11}, l_{12}\}, r_4 \rangle,$$

| | l_{10} | l_{11} | l_{12} |
|----------|------------|-------------|----------|
| l_3 | $W_{3,10}$ | false | false |
| l_7 | false | false | true |
| l_{12} | false | $W_{12,11}$ | true |

където:

$W_{3,10}$ = “определен е конкретен проект за всеки екип”;

$W_{12,11}$ = “включен е студент към екип”.

При прехода α -ядрото от позиция l_3 преминава в позиция l_{10} с характеристика:

„спецификация на конкретен проект, номер на екип”.

В позиция l_{12} стои γ ядро с характеристика:

„списък със студенти, участници в екип”.

При активиране на прехода γ -ядрото от позиция l_{12} се разцепва на γ_i ядра за всеки индивидуален участник в екипа.

По време на работа на мрежата всяко γ_i ядро постъпва в позиция l_{11} и получава характеристика:

„студент ID, номер на екип”.

$$Z_5 = \langle \{l_{10}, l_{13}\}, \{l_{13}, l_{14}\}, r_5 \rangle,$$

| | | |
|----------------|-------------|-------------|
| | l_{13} | l_{14} |
| $r_5 = l_{10}$ | $W_{10,13}$ | $false$ |
| l_{13} | $true$ | $W_{13,14}$ |

където:

$W_{10,13}$ = “дефинирана е подзадача за конкретен проект”;

$W_{13,14}$ = “избрана е подзадача за изпълнение”.

При прехода α -ядрото от позиция l_{13} получава характеристика:

„номер на проект, подзадачи за проекта”.

α -ядрото от позиция l_{13} се разцепва на α_i -ядра за всяка подзадача и по време на работа на мрежата всяко α_i ядро постъпва в позиция l_{14} с характеристика:

„номер на проект, име на подзадача”.

$$Z_6 = \langle \{l_{11}, l_{14}\}, \{l_{15}\}, r_6 \rangle,$$

| | |
|----------------|----------|
| | l_{15} |
| $r_6 = l_{11}$ | $true$ |
| l_{14} | $true$ |

По време на прехода γ_i -ядрото, постъпващо от позиция l_{11} с характеристика:

„студент ID, номер на екип”

се слива с α_i -ядро, постъпващо от позиция l_{14} с характеристика:

„номер на проект, име на подзадача”

и полученото ядро постъпва в позиция l_{15} с характеристика:

„студент γ_i , подзадача α_i ”.

$$Z_7 = \langle \{l_{15}, l_{17}, l_{19}, l_{23}, l_{27}\}, \{l_{16}, l_{17}\}, r_7 \rangle,$$

| | | |
|----------------|-------------|-------------|
| | l_{16} | l_{17} |
| l_{15} | $false$ | $true$ |
| $r_7 = l_{17}$ | $W_{17,16}$ | $W_{17,17}$ |
| l_{19} | $false$ | $true$ |
| l_{23} | $false$ | $true$ |
| l_{27} | $false$ | $true$ |

където:

$W_{17,16}$ = “работата на студента по подзадачата е приключила”;

$W_{17,17} = \neg W_{17,16}$

където $\neg P$ е отрицанието на предиката P .

По време на прехода γ -ядрото, постъпващо от позиция l_{16} получава характеристиката:

„студент γ_i , подзадача α_i е приключила”,

докато γ -ядрото в позиция l_{17} получава характеристиката:

„студент γ_i , подзадача α_i ”.

$$Z_8 = \langle \{ l_8, l_{16}, l_{18}, l_{22} \}, \{ l_{19}, l_{20}, l_{21}, l_{22} \}, r_8 \rangle,$$

| | l_{19} | l_{20} | l_{21} | l_{22} |
|----------------|-------------|-------------|----------|----------|
| l_8 | false | false | true | false |
| $r_8 = l_{16}$ | $W_{16,19}$ | $W_{16,20}$ | false | false |
| l_{18} | false | false | false | true |
| l_{22} | false | false | false | true |

където:

$W_{16,19}$ = “работата по подзадачата е продължава”;

$W_{16,20}$ = “работата по подзадачата е приключила”.

По време на прехода α -ядрото в позиция l_{19} получава характеристиката:

„подзадачата се нуждае от доработване”,

докато в позиция l_{20} получава характеристиката:

„подзадачата е изпълнена”.

През входната позиция l_{18} постъпва δ -ядро с характеристика:

„нов критерий за оценяване на проект”.

В позиция l_{22} стои δ -ядро с характеристика:

„списък от критерий за оценяване на проекти”.

По време на прехода δ -ядрото от позиция l_{22} се разцепва на две ядра, едното остава да цикли в позиция l_{22} , а второто се слива с β ядрото, постъпващо от позиция l_8 и се формира β ядро в позиция l_{21} с характеристика:

„преподавател, критерии за оценяване”.

$$Z_9 = \langle \{ l_{20}, l_{25} \}, \{ l_{23}, l_{24}, l_{25} \}, r_9 \rangle,$$

| | l_{23} | l_{24} | l_{25} |
|----------------|-------------|-------------|-------------|
| $r_9 = l_{20}$ | $W_{20,23}$ | $W_{20,24}$ | $W_{20,25}$ |
| l_{25} | false | $W_{25,24}$ | true |

където:

$W_{20,23}$ = “студент започва работа по нова подзадача”;

$W_{20,24}$ = “проектът е готов (всички подзадачи са приключени)”;

$W_{20,25}$ = “студент изчаква другите членове на екипа да приключат работа по подзадачите”;

$W_{25,24} = W_{20,24}$.

По време на прехода α -ядрото постъпващо от позиция l_{23} получава характеристиката:

„студент γ_i , подзадача α_i ”,

в позиция l_{25} получава характеристиката:

„студент γ_i изчаква членовете на екипа да приключат работа по подзадачите”,

докато α -ядрото в позиция l_{24} получава характеристика:

„работата по проекта е приключена”.

$$Z_{10} = \langle \{l_{24}\}, \{l_{26}\}, r_{10} \rangle,$$

$$r_{10} = \begin{array}{c|c} & l_{26} \\ \hline l_{24} & W_{24,26} \end{array}$$

където:

$W_{24,26}$ = “проведено е представяне и защита на проекта”.

По време на прехода α -ядрото постъпващо от позиция l_{26} получава характеристиката:

„номер на екип, оценка от защита на проекта”.

$$Z_{11} = \langle \{l_{21}, l_{26}, l_{30}\}, \{l_{27}, l_{28}, l_{29}, l_{30}\}, r_{11} \rangle,$$

$$r_{11} = \begin{array}{c|cccc} & l_{27} & l_{28} & l_{29} & l_{30} \\ \hline l_{21} & false & false & true & false \\ l_{26} & W_{26,27} & W_{26,28} & false & true \\ l_{30} & false & false & false & true \end{array}$$

където:

$W_{26,27}$ = “получена е слаба оценка за проекта”;

$W_{26,28}$ = “получена е положителна оценка за проекта”.

По време на прехода β -ядро от позиция l_{21} се слива с α -ядрото от позиция l_{26} и новополученото ядро постъпва в позиция l_{27} с характеристика:

„номер на екип, слаба оценка за проекта”,

или в позиция l_{28} с характеристика:

„номер на екип, положителна оценка за проекта”.

β ядрото в позиция l_{29} не получава нова характеристика. α -ядрото в позиция l_{30} има характеристика:

„оценки на студентите, относно работата им по проектите”.

ИЗВОД: Разгледани са стъпките и техните характеристики при провеждане на проектно-базираното обучение в допълнение на традиционните педагогически подходи. Изгражда се обощеномрежов модел на процеса на обучение, посредством разработване на проект [149]. Моделът служи за анализиране на характеристики и възможности при работа по адаптирани към знанията и уменията на студентите проекти, чието разработване води до интегриране на знания, придобивани в различни курсове от обучението. Интеграцията на проектно-базираното обучение с възможностите на електронното и веб-базираното обучение правят процеса на учене по-привлекателен и по-разнообразен за студентите.

4.5. Модел за игровизация на електронен курс за обучение

Игровизацията на обучението има за цел да интегрира игрови елементи и техники в процеса на е-обучение, за подпомагане усвояването на учебното съдържание. Тези игри имат образователни, тренировъчни или информационни цели. Обикновено те са [24]:

- *Gamed-based Learning* - обединява учебни ресурси с видео и електронни игри за постигане на учебните цели. Игровата среда води до мотивиране и желание за самоусъвършенстване от страна на обучаемите;

- *Симулационни игри* – свързани с пресъздаване на реални събития или процеси, предназначени за търсене и достигане до решения на проблеми;

- *Edutainment* – представят съдържание, предназначено едновременно да обучава и забавлява (терминът произхожда от сливане на думите *education* и *entertainment*).

Игровизацията предлага учебният процес да бъде организиран в игрови нива (заклучени или отключени с входни изисквания), които представят различни секции с учебни ресурси и дейности за преминаване. Курсът на обучение се въвежда с интересен контекст (сюжет/история) на дейностите, които трябва да бъдат извършени. Правилата за обучение се въвеждат като игрови правила. Всяко ниво може да включва предизвикателства - учебни дейности (четене на текстов файл, гледане на видео, изпълнение на тест, писане/решаване на задания и т.н.), които трябва да бъдат осъществени за постигане на учебните цели на нивото. Някои от заданията са индивидуални, а други са съвместни, където обучаемите работят в екипи. Заданията имат времеви ограничения за приключване на дейността.

За завършване на някои дейности обучаемите получават очаквани бонуси (например точки). За постигане на набор от изисквания обучаемите могат да получават значки: (Champion), Суперзвезда (Superstar), Авантюрист (Adventurer), Изследовател (Explorer) и т.н. За отлични резултати или за извършване на конкретна дейност, могат да получат неочаквани награди (допълнителна интересна информация, точки/оценка, материални награди (вкл. виртуални предмети) и т.н.). За специфична завършена дейност могат да се получават предимства като комбо (помощ, препоръки, по-подробни примери, удвояване на точки от дадена дейност и т.н.). Някои от учебните елементи могат да са заключени (скрити съкровища) и да бъдат отключени, когато обучаемите отговорят на определени изисквания.

Обучаемите могат да участват в игровизирания курс с конкретна игрова роля, която има и визуално представяне чрез някакъв образ (аватар). Въз основа на събраните точки и текущото игрово ниво обучаемите се подреждат в класацията, където могат да се видят водещите участници в учебния процес. По време на целия процес обучаемите имат достъп до информация за своя напредък в обучението като прогрес в играта и във всеки текущ момент – за своя статус (текущ аватар, достигнато игрово ниво, спечелени точки, значки и други награди).

Разработени са серия от модели за проследяване на процеса на обучение и резултатите на обучаемите [158, 164, 204]. Използването на методи на изкуствения интелект дава възможност за предвиждане на резултатите и частична автоматизация на процеса на оценяване. Обобщеномрежов модел за процесите по избор и изграждане на

подходяща E-Test система е даден в [143]. Нов подход за формиране на текущи оценки за знанията и уменията на обучаемите, базирани на размити множества е представен в [98]. Настоящия модел обобщава тези модели и ги разширява с възможности за прилагане на модули за игровизация на обучението и проследяване на обучаемите при работата им.

Създаденият обобщен модел на процеса на игровизация на обучаващ курс е минимален редуциран обобщеномрежов модел [153] и съдържа множеството от преходи:

$$A = \{Z_1, Z_2, Z_3, Z_4, Z_5, Z_6\}$$

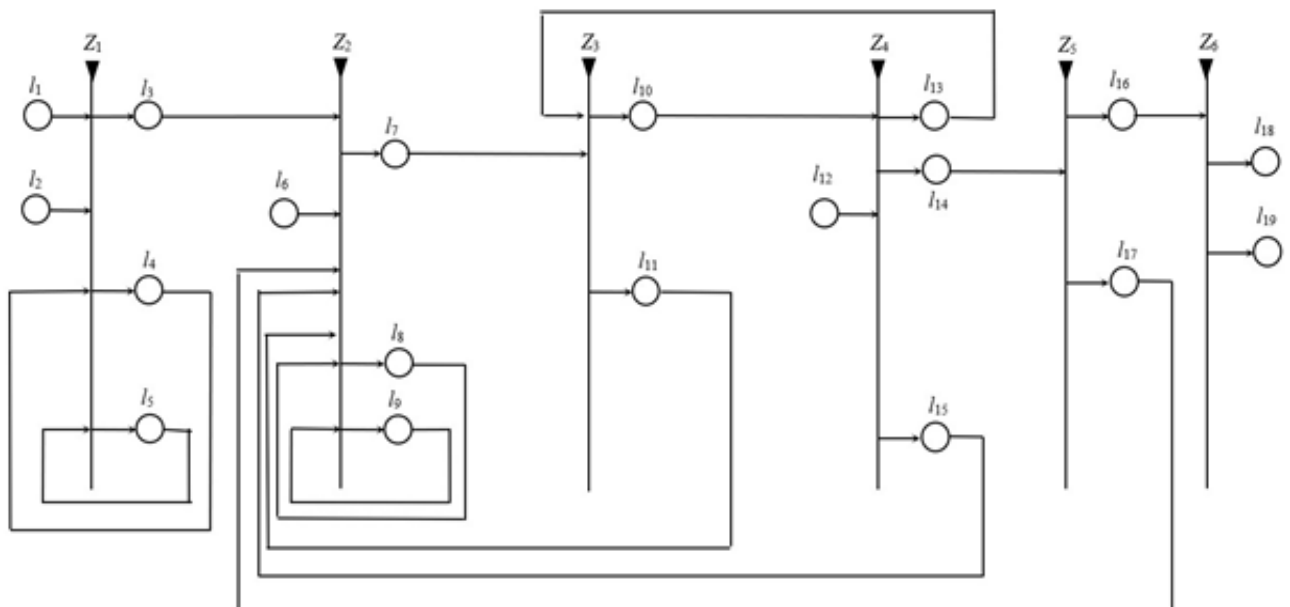
където:

- Z_1 – Създаване на модел и игровизация на обучението;
- Z_2 – Дефиниране на правила за обучението в игровизиран курс;
- Z_3 – Провеждане на обучение с електронния курс;
- Z_4 – Оценяване на обучаемите и награждаване;
- Z_5 – Класиране и анализ на резултатите на обучаемите;
- Z_6 – Анализ на обучението и оценяване на курса.

Следните ядра се използват за описание на процесите:

- α -ядро – потребители на обучаващата среда (администратори, автори на курса, студенти);
- β -ядро – електронни курсове за обучение и инструменти за игровизация (плагини, софтуерни компоненти и др.);
- γ -ядро – оценъчни модели;
- η -ядро – данни за обучаваните обекти по време на обучението;
- κ -ядро – специфични данни за игровизираните модули за обучаваните обекти;
- μ -ядро – правила и критерии за прилагане на игрови елементи и награждаване.

Обобщеномрежовият модел на процеса на игровизация на обучаващ курс е представен на фиг. 43.



Фигура 43. Обобщеномрежов модел на процеса на игровизация на електронен обучаващ курс

Игровизацията на обучението, се разделя на два основни типа: структурна промяна на обучаващ курс и съдържателна промяна на обучаващ курс.

При структурната промяна обучаемите преминават през стандартните учебни ресурси, но към курса се включват различни игрови елементи като: точки, значки, класации на първенците, аватари, медали и награди и др. Най-използваните игровите елементи при игровизация на обучението са:

- Аватар - представяне на различни роли на обучаемите, които могат да бъдат използвани в мисии (учебни дейности).
- Ниво - различните нива на учебния процес, могат да бъдат разглеждани като нива на игра.
- Бонус – получаване на различни бонуси за извършени учебни действия.
- Значка - отличителен знак за натрупани постижения в различни учебни дейности.
- Награда - наградите се получават неочаквано за постигане на определени успехи.
- Класация - участниците подредени по-успех.
- Отбор/Екип - за реализиране на групови учебни дейности и отчитане на способността на участниците за работа в екип.
- Ресурс - игрови ресурси могат да бъдат различните видове учебни ресурси.
- Време – поставяне на времеви ограничения за учебни дейности.
- Прогрес - представяне на процентното съотношение на изпълнените цели на обучението.
- Статус - разглежда текущото състояние на обучаемия – текущ аватар, оценяване на обучаемия, учебна активност и др.

При съдържателна промяна на обучаващ курс се използват игрови техники за представяне на учебното съдържание под формата на правила, нива и умения, използвани в електронните игри, които трябва да бъдат постигнати от обучаемите. Например:

- Правилата на учебния процес могат да бъдат разгледани като правила на игра;
- Комуникация - обучаемите могат да споделят идеи и проблеми, да осъществяват сътрудничество и работа по групови учебни дейности и за социална комуникация;
- Обратната връзка – преподавателят може да даде мнение за работа на обучаемия;
- Мисия/ Предизвикателство/ Приключение – могат да бъдат всички учебни активности, които обучаемият трябва да осъществи в рамките на учебния курс и към тях може да бъде добавен игрови сюжет, който да описва целта на мисията;
- Скрито съкровище - скрити учебни ресурси, които могат да бъдат открити/отворени само при постигане на определени условия (например при завършване на мисия);
- Сюжет / История – това е контекста на учебния процес, може да описва история с различни мисии за изпълнение.

Преход 1: Създаване на модел за игровизация на обучаващ курс

$$Z_1 = \langle \{ l_1, l_2, l_4, l_5 \}, \{ l_3, l_4, l_5 \}, r_1 \rangle,$$

| | | | | |
|---------|-------|--------------|--------------|--------------|
| $r_1 =$ | | l_3 | l_4 | l_5 |
| | l_1 | $W_{1,3}$ | <i>true</i> | <i>false</i> |
| | l_2 | <i>false</i> | <i>false</i> | $W_{2,5}$ |
| | l_4 | $W_{4,3}$ | <i>true</i> | <i>false</i> |
| | l_5 | $W_{5,3}$ | <i>false</i> | <i>true</i> |

където:

$W_{1,3}$ = „постъпил е нов потребител на обучаващата среда (администратор, автори на курс, студент“ или др.);

$W_{2,5}$ = „постъпил е нов обучаващ курс или инструмент за игровизация“;

$W_{4,3}$ = „потребител на обучаващата среда започва работа с обучаващ курс“;

$W_{5,3}$ = „приложен е инструмент (модул) за игровизация върху обучаващ курс“.

Процесът се стартира при постъпване на α -ядро в обобщената мрежа от позиция l_1 , с начална характеристика:

„*ID потребител, роля, дисциплина*“.

През позиция l_2 могат да постъпват нови инструменти за игровизация чрез β -ядра, с начална характеристика:

„*ID, разработен софтуер за игровизация (плагин, игровизиран модул)*“

В позиция l_4 цикли α -ядро с характеристика:

„*списък с налични потребители на обучаващата среда*“.

В позиция l_5 цикли β -ядро с характеристика:

„*електронни курсове за обучение и игровизирани модули*“.

β -ядрото от позиция l_5 се разделя на две ядра, едното остава в позиция l_5 , а другото ядро се слива с α -ядрото от позиция l_1 и постъпва в позиция l_3 като получава характеристика:

„*ID потребител, ID обучаващ курс, приложен инструмент за игровизация*“.

Преход 2: Дефиниране на правила за обучението в игровизиран курс

$$Z_2 = \langle \{ l_3, l_6, l_8, l_9, l_{11}, l_{15}, l_{17} \}, \{ l_7, l_8, l_9 \}, r_2 \rangle,$$

| | | | | |
|---------|----------|--------------|--------------|--------------|
| $r_2 =$ | | l_7 | l_8 | l_9 |
| | l_3 | $W_{3,7}$ | $W_{3,8}$ | $W_{3,9}$ |
| | l_6 | <i>true</i> | <i>false</i> | <i>true</i> |
| | l_8 | $W_{8,7}$ | <i>true</i> | <i>false</i> |
| | l_9 | $W_{9,7}$ | <i>false</i> | <i>true</i> |
| | l_{11} | <i>false</i> | $W_{11,8}$ | $W_{11,9}$ |
| | l_{15} | <i>false</i> | $W_{15,8}$ | $W_{15,9}$ |
| | l_{17} | <i>false</i> | $W_{17,8}$ | <i>false</i> |

където:

$W_{3,7}$ = „потребител започва обучение по избран курс в обучаващата среда“;

$W_{3,8}$ = „въведени са данни в базата от данни за потребителя и избрания курс“;

$W_{3,9}$ = “въведени са специфични данни за игровизация на курс“;
 $W_{8,7}$ = “получени са данни за обучаемия, свързани с обучаващия курс“;
 $W_{9,7}$ = “получени са данни за обучаемия, свързани с игровизацията на курс“;
 $W_{11,8}$ = „постъпили са данни за текущо оценяване на обучаем в базата от данни“;
 $W_{15,8}$ = „постъпили са нови данни за обучаем в базата от данни“;
 $W_{17,8}$ = „постъпили са финални данни за обучаем в базата от данни“.
 $W_{11,9}$ = „постъпили са специфични данни за обучаем с игровизиран курс“;
 $W_{15,9}$ = „постъпили са данни за класиране на обучаем в игровизиран курс“.

По време на активиране на прехода Z_2 , през позиция l_6 постъпва μ -ядро с характеристика:

„правила и критерии за прилагане на игрови елементи и награждаване“,
 докато α -ядрото от позиция l_3 постъпва в позиция l_7 и получава характеристиката:
 „ID потребител, ID обучаващ курс, приложен инструмент за игровизация,
 критерии за прилагане на игрови елементи“.

В позиция l_8 цикли η -ядро с характеристика:

„база от данни за обучаваните обекти по време на обучението“.

Базата от данни при различните среди за обучение може да има различна структура, но тя съхранява данни, относно обучаемите и постигнатите резултати по време на обучението.

При игровизация на обучаващ курс към стандартните таблици в базата от данни се добавят и таблици с данни, необходими за игровизираните модули, поради необходимост от съхранение на специфична информация, която включва например:

- данни за аватара на обучаемия в различните курсове;
- данни за наградените действия на обучаемите и получен брой точки;
- данни за достигнато ниво, общ брой точки и прогреса за нивото в различните курсове.

Новите таблици са свързани към базовите таблици и съдържат данни: *user* и *course*, защото информацията е специфична за всеки курс и потребител. Освен това се пази и информация за качените на сървъра допълнителни файлове. За целта в позиция l_9 цикли κ -ядро с характеристика:

„специфични данни за игровизираните модули за обучаваните обекти“.

Преход 3: Провеждане на обучение. Обучаемите провеждат обучение, като използват учебните ресурси и дейности на игровизирания курс.

$$Z_3 = \langle \{ l_7, l_{13} \}, \{ l_{10}, l_{11} \}, r_{13} \rangle,$$

$$r_3 = \begin{array}{c|cc} & l_{10} & l_{11} \\ \hline l_7 & W_{7,10} & W_{7,11} \\ l_{13} & false & W_{13,11} \end{array}$$

където:

$W_{7,10}$ = „потребителят е завършил обучението по избрания курс“ ;
 $W_{7,11}$ = „потребителят продължава обучението по избрания курс“ ;
 $W_{13,11} = W_{7,11}$.

По време на активиране на прехода Z_3 α -ядро постъпва в позиция l_{10} или в позиция l_{11} , като в позиция l_{10} получава характеристика:

„ID потребител, ID обучаващ курс, приложен инструмент за игровизация, допълнителни данни за игровизация“,

а в позиция l_{11} не получава нова характеристика.

Преход 4: Оценяване на обучаемите и награждаване. Някои от учебните активности се оценяват, за да се установи дали са постигнати текущите учебни цели. След оценяване, обучаемите се награждават чрез различни средства (стимули): бонуси, значки, награди и скрити съкровища.

$$Z_4 = \langle \{ l_{10}, l_{12} \}, \{ l_{13}, l_{14}, l_{15} \}, r_4 \rangle,$$

$$r_4 = \begin{array}{c|ccc} & l_{13} & l_{14} & l_{15} \\ \hline l_{10} & W_{10,13} & W_{10,14} & W_{10,15} \\ l_{12} & true & true & false \end{array}$$

където:

$W_{10,13}$ = „текущата оценка е по-ниска от дефинирана прагова стойност в критериите за оценяване и е необходимо допълнително обучение“;

$W_{10,14}$ = „текущата оценка е по-висока от дефинирана прагова стойност в критериите за оценяване“;

$W_{10,15}$ = „постъпили са специфични данни за обучаем в игровизиран курс“.

По време на активиране на прехода Z_4 , през позиция l_{12} постъпва γ -ядро с характеристика:

„оценъчни модели и критерии за оценяване“,

α -ядрото от позиция l_{10} се слива с γ -ядрото от позиция l_{12} и постъпва в позиция l_{13} или l_{14} като и в двете позиции получава характеристиката:

„ID потребител, ID обучаващ курс, приложен инструмент за игровизация, текуща оценка“.

Преход 5: Класиране и анализ на резултатите на обучаемите. В резултат на дейността си, обучаемите получават точки и/или ново ниво, които се отразяват на позицията на участника в класацията.

$$Z_5 = \langle \{ l_{14} \}, \{ l_{16}, l_{17} \}, r_5 \rangle,$$

$$r_5 = \begin{array}{c|cc} & l_{16} & l_{17} \\ \hline l_{14} & W_{14,16} & W_{14,17} \end{array}$$

където:

$W_{14,16}$ = „извършено е класиране на обучаемите“.

$W_{14,17}$ = „постъпили са данни за класиране на обучаем в игровизиран курс“.

По време на активиране на прехода Z_5 , α -ядрото от позиция l_{14} постъпва в позиция l_{16} като получава характеристика:

„ID потребител, ID обучаващ курс, приложен инструмент за игровизация, текуща оценка, класиране“.

Преход 6: Анализ на обучението и оценяване на курса.

$$Z_6 = \langle \{ l_{16} \}, \{ l_{18}, l_{19} \}, r_5 \rangle,$$

$$r_6 = \frac{\quad}{l_{16}} \left| \begin{array}{cc} l_{18} & l_{19} \\ W_{16,18} & W_{16,19} \end{array} \right.$$

където:

$W_{16,18}$ = „направен е анализ на резултатите от обучението с игровизирания курс“;

$W_{16,19}$ = „извършено е оценяване на игровизирания курс за обучение“.

При активиране на прехода Z_6 , α -ядра постъпват в позиция l_{18} или в позиция l_{19} .

В позиция l_{18} α -ядрото получава характеристиката:

„анализ на резултатите от обучението с игровизирания курс“,

а в позиция l_{19} получава характеристика:

„оценка на игровизирания курс за обучение“.

Игрите притежават силен мотивиращ механизъм, затова те все по-често се използват, за да мотивират обучаемите за постигане на по-ефективно обучение чрез игрови ситуации. В процеса на игровизация на обучението трябва да се постигнат игровите (учебните) цели при спазване на правилата на играта (учебните правила), като трябва да се изпълнят редица игрови (учебни) дейности чрез използване на игрови (учебни) ресурси. Примери за такива среди са: GENIE, The Knowledge Arcade, TalentLMS, Frog, Expertus One, Accord LMS, Axonify, etc. [26, 55, 121]. За съжаление тези системи, не са свободно достъпни, а се предлагат като платени софтуери. Средата Мудъл предоставя възможности за създаване на персонализирана учебна среда и доставя набор от инструменти за поддръжка с отворен код, като модули за игровизация могат да се използват: LevelUp, Ranking block и Stash.

Създаденият обобщеномрежов модел на процеса на игровизация на обучаващ курс предлага решение на един от проблемите, формулиран в книгата на акад. Красимир Атанасов „Generalized Nets and Intuitionistic Fuzziness in Data Mining“ [34].

4.6. Модели за осигуряване на качеството и акредитация на висшето образование

Осигуряването на качеството във висшето образование е непрекъснат процес, който изисква много усилия от страна на образователните институции и акредитиращите организации. Основните форми са две – *вътрешно и външно осигуряване на качеството*. Висшите училища изграждат *вътрешни системи за осигуряване на качеството*, като са свободни да установяват собствени правила и критерии за оценяване на различни обекти и субекти на образователния процес. Те подлежат и на *външно оценяване – акредитация, което се извършва от лицензирани национални и международни агенции*.

Необходимостта от съизмеримост на качеството на висшето образование в Европейския съюз мотивира дефинирането на единни стандарти за качество. През 2005 година са разработени Standards and Guidelines for Quality Assurance in the European Higher Education Area (ENQA). Документа (ESG) [50] съдържа множество стандарти и насоки за прилагането им. България е една от страните, които прилагат стандарта ESG. Осигуряването на качеството на висшето образование в България се извършва от Националната агенция за оценяване и акредитация (НАОА)[5].

Процедурите за осигуряване на качеството протичат по обща схема и включват няколко основни дейности: самооценяване, посещение на експертна група, оценяване и следакредитационен контрол. Етапите на една оценъчна процедура могат да бъдат определени по следния начин:

- **Самооценяване.** Оценяваната институция осигурява база за провеждане на външното осигуряване на качеството, като провежда предварителна самооценка и подготвя доклад-самооценка. Той следва да бъде разработен съобразно изискванията на критериалната система на НАОА за съответната процедура и да съдържа самооценка и анализ на постигнатото според основните стандарти, които са обект на проверката и оценяването.
- **Посещение на експертна група (ЕГ) в оценяваната институция.** ЕГ проверява предоставената информация и ако е необходимо, изисква допълнителна информация. ЕГ изготвя доклад за извършената проверка, който съдържа оценка и конкретни насоки за бъдещи действия, насочени към повишаване на качеството на обучението.
- **Оценяване.** След обсъждания и гласувания, първо в постоянната комисия, определила ЕГ, а след това и в акредитационния съвет, в НАОА се взема окончателно решение за оценката и се мотивира със съответни становища и доклади.
- **Следакредитационен контрол.** Включва процедури за последващ контрол в образователната институция, за да се проследи изпълнението на направените препоръки.

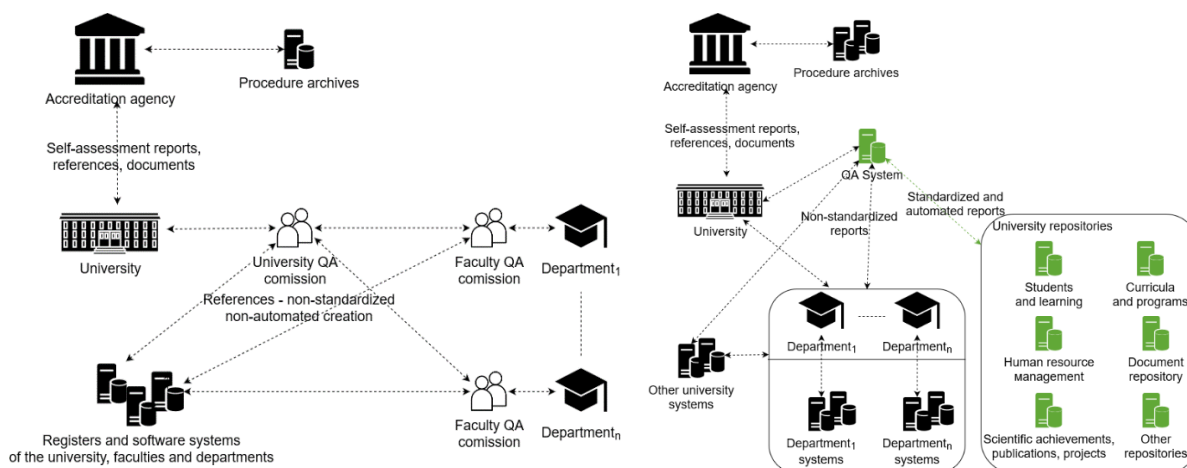
Акредитационните процедури се провеждат периодично – през периоди, които варират от 3 до 6 години, в зависимост от типа на акредитационната процедура и/или предходната оценка. Предстои унифициране на периода на 5 години за бакалавърските и магистърските програми. Този процес е свързан с *няколко месечно пренасочване на усилията на голяма част от академичния състав от образователния процес, към*

акредитационната процедура. Университетските служители се откъсват от основните си дейности и полагат извънредни усилия, за да докажат спазването на акредитационните критерии и да получат висока институционална оценка.

Процедурите за оценяване и акредитация на висшето образование трябва да бъдат естествена и неразделна част от образователния процес. Те трябва да се провеждат без много усилия, незабележимо за студенти и преподаватели. За целта обаче, е необходимо образователните институции да поддържат хранилища с актуална информация за дейността си, а акредитиращите – да осигурят софтуерни инструменти за автоматизирана оценка на качеството, съгласувани с информацията в националния център за информация и документация (НАЦИД).

За изготвянето на доклад-самооценка, е необходимо събирането на голям обем разнотипна информация и документи, и изготвянето на множество разнообразни справки. Това включва справки за структура на академичния състав, проведени анкети, участия в научни и образователни форуми, брой на международните научни договори, информация докторанти, публикации и цитирания на академичния състав и др. Много висши училища са изградили и поддържат софтуерни системи за публикационната и научната дейност на академичния състав, но не винаги е възможно необходимата за акредитацията информация да бъде извлечена автоматично. Значимият фактор за това е начина на структуриране на тази информация в университетската система, и дали има съответствие с изискванията на НАОА. В случай, че във висшето училище няма собствена централизирана софтуерна система, тази информация се събира от академичния състав по катедри, обобщава се на ниво факултет, а след това и на ниво университет.

В практиката обикновено се използват различни приложения, които частично автоматизират различни дейности по осигуряване на качеството (QA). Модели на текущото състояние са създадени в [99, 102]. Обобщен модел на текущото състояние е представен на фиг. 44.



Фигура 44. Модел на процесите при изготвяне на доклад-самооценка от висшите училища

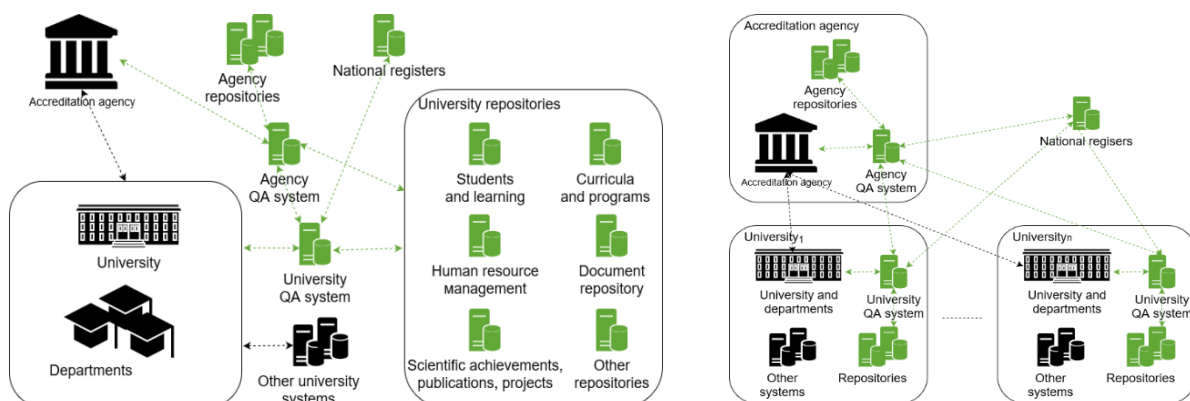
а) Модел без университетска софтуерна QA система за качество

б) Модел с университетска софтуерна QA система за качество

При множество висши училища липсва система, подпомагаща създаването на доклад-самооценка (фиг. 44 а). В този случай разпределянето на задачите по изготвянето му става йерархично: НАОА открива процедура за акредитация и осведомява университета; комисията по акредитация разпределя задачи по изготвяне на справки и подготовка на текстове за отделните стандарти към звената; звената събират информация от различни регистри, университетски софтуерни системи със структурирани или неструктурирани данни, а също и не дигитализирана информация. Въпреки събраните данни от отделни служители, създаването на крайния вариант на доклада-самооценка е трудно, поради необходимостта от обобщаване на предоставените в различни формати данни и привеждането на текстовете в документ с общ литературен стил. Наличието на софтуерни инструменти, подпомагащи създаването на доклад-самооценка (фиг. 44 б) предоставя възможности за: текущ контрол и проследяване на дейностите по съставяне на доклад-самооценка; автоматизация на достъпа до стандартизирани справки от вътрешни и външни регистри; сътрудничество между участниците в оценъчните процедури и др.

4.6.1. Слабо-централизиран модел за акредитация

Един възможен подход за автоматизация на акредитационните процедури е чрез *изграждане на централизирана система на акредитационната агенция и НАЦИД и самостоятелни университетски системи за осигуряване на качеството* (фиг. 45). Всяка университетска QA система автоматизирано извлича необходимите акредитационни данни от университетските хранилища [99]. Друга част от информацията, касаеща обосновки, обяснения и изводи, се въвежда от университетски отговорници и експерти по качеството. В определен момент от време, информацията от университетската QA система се експортира към QA системата на акредитиращата институция. Предимствата при този подход са възможността всяко висше училище да изгради собствена QA система според нуждите си и в съответствие със стандартите и протоколите, зададени от акредитиращата агенция. Улеснение е предлагането на стандартизирани университетски QA системи, получаващи автоматизирани обновления и които институциите биха могли самостоятелно да инсталират, конфигурират и използват. Наличието на централизирана система предполага университетите да ползват предоставените от НАОА и НАЦИД възможности.



Фиг. 45. Слабо-централизиран модел за акредитация

а) Основни системи и комуникационни интерфейси в процеса на акредитация на едно висше училище

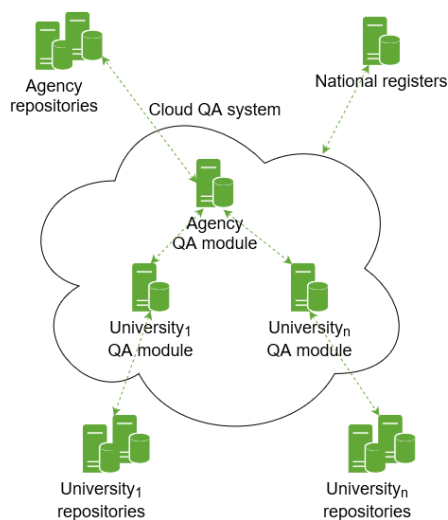
б) Основни системи и комуникационни интерфейси в процеса на акредитация в системата на висшето образование

Предимствата на този подход определят и неговите недостатъци. Създаването на собствено решение или поддръжката на стандартизирано приложение от страна на университетите изисква наличието на специализирани ИТ специалисти във всяко висше училище. Създаването на стандартизирана университетска QA система е трудоемка задача, която трябва да се съобрази с наличието на множество различни университетски хранилища и други софтуерни системи.

4.6.2. Силно-централизиран модел за акредитация

Създаването на национална централизирана софтуерна система за осигуряване на качеството ще позволи по-ефективна софтуерна поддръжка, ще осигури прозрачност и непрекъснат мониторинг на качеството на обучението. Силно-централизираният модел (фиг. 46) предлага *използването на централизирана cloud система за акредитация*. Системата поддържа модул както за НАОА и НАЦИД, така и конфигурируеми модули, обслужващи акредитационните процеси за отделните висши училища. Използването на единна система предполага улеснена интеграция и комуникация между основните модули, лесна за потребителите актуализация на софтуера, възможности за лесна интеграция на бъдещи софтуерни разширения, създаване на динамични анализи и справки върху данните, предоставени от всички институции [99].

Докато в слабо-централизирания модел има не-автоматизирани връзки и процеси, то силно централизираният модел, освен че премахва множество от връзките между различни несигурни компоненти в софтуерната среда, също така и създава по-големи възможности за автоматизация на процесите по осигуряване на качеството и акредитацията.



Фигура 46. Централизиран модел за акредитация

Основна цел на описаните модели е предлагане на *платформа*, улесняваща работата на всички участници в процедурите по оценяване и акредитация – от страна на обучаващите, и от страна на акредитиращите институции. Те дават възможност за автоматизиране на процесите, свързани с различните етапи на акредитационните процедури. Използването на софтуерни системи, базирани на предложените модели има много предимства: централизиране на процесите по оценяване, което улеснява тяхното

администриране и мониторинг; осигуряване на обща платформа за екипна асинхронна работа на потребители с различни права и отговорности; възможност за работа по всяко време и от всяко място; улеснена комуникация и прозрачност; автоматизирано генериране на документация; оптимизиране на времеви и материални ресурси; намаляване вероятността от допускане на грешки, и др. Слабо-централизираният модел предоставя по-голяма възможност за интеграция на университетската QA система с други вътрешни системи на институцията и собствен контрол върху бизнес данните. Основните недостатъци са необходимостта от допълнителни ИТ специалисти за поддръжка и надграждане на QA системата, както и дублиране на голям обем информация в централизирания модул на агенцията. Силно-централизираният модел е по-лесен за използване от висшето училище, минимизира нуждата от административна поддръжка от страна на висшето училище, премахва необходимостта от дублиране на информация, но усложнява възможностите за интеграцията му с наличните университетски системи.

4.6.3. Обобщеномрежов модел на процесите на обработка на данни при самооценяване във висше училище

Модел на централизирана система за осигуряване на качеството на висшето образование е създаден в [101]. Обобщеномрежов модел на система за осигуряване на качеството във висше училище е представен в [100]. Тук се създава минимален редуциран обобщеномрежов модел, описващ процесите на обработка на данни при самооценяваща процедура във висшето учебно заведение, който съдържа следния набор от преходи:

$$A = \{Z_1, Z_2, Z_3, Z_4\},$$

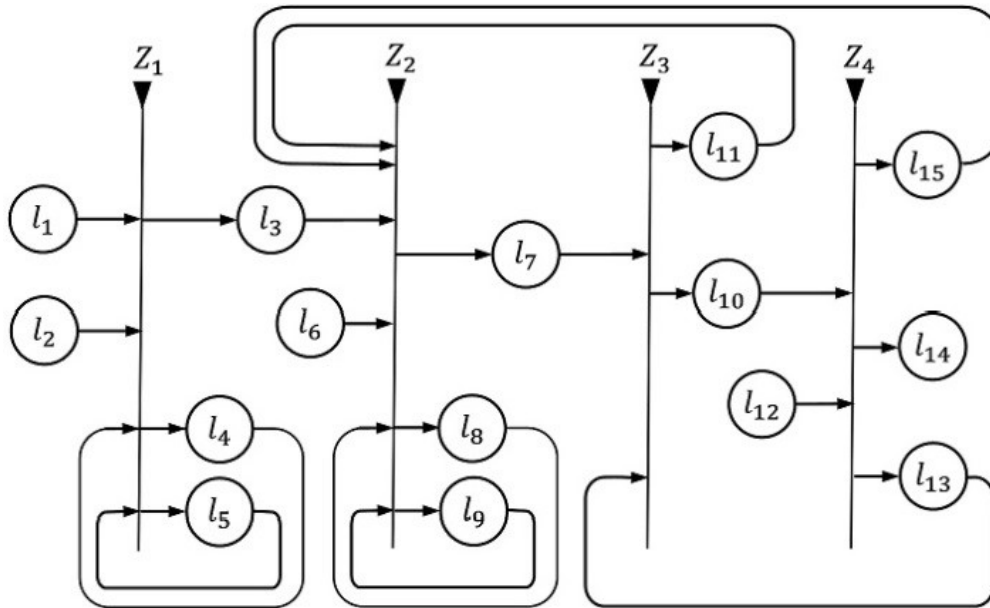
където:

- Z_1 – Събиране на данни от различни източници;
- Z_2 – Интегриране на данни;
- Z_3 – Обработка на данни за изискванията, свързани с критериалната система;
- Z_4 – Изготвяне на доклад за самооценка и съответните приложения.

Следните ядра се използват за описание на процесите:

- α -ядра – данни за дейността на висшето училище;
- β -ядра – членове на административния и академичен персонал, членове на комисията по качеството и групата за изготвяне на доклад за самооценка;
- μ -ядра – критерии и критериална система на акредитиращата институция;
- η -ядро – хранилище на данни с актуална информация за дейността на висшето училище, необходима за изготвяне на доклад за самооценка;
- γ -ядра – инструменти за обработка на данни;
- σ -ядра – заявки за данни от потребител/приложение.

Обобщеномрежовият модел, описващ процесите при обработка на данни при самооценяваща процедура във висшето учебно заведение е представен на фиг. 47.



Фигура 47. OM модел на процесите на обработка на данни при самооценяване във ВУЗ

Преход 1: Събиране на данни от различни източници

Информацията се събира от различни административни и образователни звена. В някои случаи няма установени стандарти и често един и същи тип информация се структурира по различен начин. Част от информацията се обработва ръчно, като актуализирането на вече обработената информация изисква повторна обработка и обобщаване. Ако учебното заведение не разполага с единно хранилище на административни и образователни документи, съществува риск от предоставяне на дублирана информация, използване на остарели данни и документи и др.

Процесът се стартира, когато α -ядро постъпи в обобщената мрежа в позиция l_1 с начална характеристика:

„key-data, data-value“.

Чрез позиция l_2 , нови членове на административния и академичен персонал могат да постъпват в мрежата чрез β -ядра, с начална характеристика:

“ID user, name”.

В позиция l_4 стои α -ядро с характеристика:

“база от данни за висшето училище“.

$$Z_1 = \langle \{ l_1, l_2, l_4, l_5 \}, \{ l_3, l_4, l_5 \}, r_1 \rangle,$$

| $r_1 =$ | l_3 | l_4 | l_5 |
|---------|-----------|-------|-----------|
| l_1 | $W_{1,3}$ | true | false |
| l_2 | false | false | $W_{2,5}$ |
| l_4 | $W_{4,3}$ | true | false |
| l_5 | false | false | true |

където:

$W_{1,3}$ = “нови данни (множество от данни) са постъпили за обработка и съхранение”;

$W_{2,5}$ = “нов потребител (член на административния и академичен състав) е постъпил;

$W_{4,3}$ = “данни са извлечени от базата от данни за висшето училище”.

В позиция l_5 стои β -ядро с характеристика:

“*списък потребители - членове на административния и академичен състав*”.

Потребителите членове на административния и академичен състав са с различни компетенции, участващи в процедурите по оценяване. Те са от различни звена, имат различни права и отговорности, различен достъп до информация, различно мнение и виждане за качеството на образованието, умения за работа в екип и др.

По време на активирането на преход Z_1 , α -ядро в позиция l_5 се разделя на две ядра. Едното остава в позиция l_5 а другото ядро се слива с α -ядрото от позиция l_1 и постъпва в позиция l_3 получавайки характеристика:

“*ID user, key-data, data-value*”.

Преход 2: Интегриране на данни в хранилището.

$$Z_2 = \langle \{ l_3, l_6, l_8, l_9, l_{11}, l_{15} \}, \{ l_7, l_8, l_9 \}, r_2 \rangle,$$

| | | | | |
|---------|----------|--------------|--------------|--------------|
| $r_2 =$ | | l_7 | l_8 | l_9 |
| | l_3 | <i>false</i> | $W_{3,8}$ | $W_{3,9}$ |
| | l_6 | <i>true</i> | <i>false</i> | <i>true</i> |
| | l_8 | $W_{8,7}$ | <i>true</i> | <i>false</i> |
| | l_9 | $W_{9,7}$ | <i>false</i> | <i>true</i> |
| | l_{11} | <i>false</i> | $W_{11,8}$ | $W_{11,9}$ |
| | l_{15} | <i>false</i> | $W_{11,8}$ | $W_{15,9}$ |

където:

$W_{3,8}$ = “данни са постъпили в хранилището”;

$W_{3,9}$ = “обработката на данни е извършена от софтуерни инструменти”;

$W_{8,7}$ = “данни (група от данни) са извлечени от хранилището”;

$W_{9,7} = W_{3,9}$;

$W_{11,8}$ = “постъпили са нови данни в резултат на аналитична обработка на данни”;

$W_{15,8}$ = “получена е заявка за данни от хранилището”;

$W_{11,9}$ = “имплементиран е инструмент за обработка на данни”;

$W_{15,9} = W_{11,9}$.

По време на активиране на прехода Z_2 , μ -ядро постъпва през позиция l_6 с характеристика:

“*критериална система на акредитиращата институция*”.

В позиция l_8 цикли η -ядро с характеристика:

“*Хранилище с актуална информация за дейността на висшето училище, необходима за изготвяне на доклад за самооценка*”.

α -ядро от позиция l_3 постъпва в позиция l_8 и при следващо активиране на прехода преминава в позиция l_7 като получава характеристиката:

“*ID user, key-data, data-value, mun обработка на данните*”.

В практиката обикновено се използват различни приложения, които частично автоматизират различни дейности по осигуряване на качеството. Те автоматично извличат необходимите данни за акредитация от университетските хранилища (студенти и обучение, учебни планове и програми, управление на човешките ресурси, хранилище на документи, научни постижения, публикации, проекти и др.). Тези инструменти могат да извършват предварителна обработка на получените данни, както и необходимите трансформации на формата, и впоследствие тези данни се интегрират в хранилището на данни. За целта в позиция l_9 цикли γ -ядро с характеристика:

“*Data processing tools*”.

Преход 3: Обработка на данни в съответствие с изискванията, свързани с критериалната система

За изготвянето на доклада за самооценка, е необходимо да се събере голямо количество разнообразна информация и документи, както и да се изготвят различни приложения. Това включва например доклади за структурата на академичния състав, резултати от анкети, проведени сред студентите, участие в национални и международни научни и образователни форуми, брой международни научни договори, информация за успешни докторанти, брой публикации на академичен състав, публикации в чужбина и на международни форуми, публикации в рецензирани научни списания, брой цитирания на научни статии и др.

$$Z_3 = \langle \{ l_7, l_{13} \}, \{ l_{10}, l_{11} \}, r_{13} \rangle,$$

$$r_3 = \begin{array}{c|cc} & l_{10} & l_{11} \\ \hline l_7 & W_{7,10} & W_{7,11} \\ l_{13} & false & W_{13,11} \end{array}$$

където:

$W_{7,10}$ = “процесът на анализ на данни не е завършен”;

$W_{7,11}$ = “процесът за аналитична обработка на данните е завършен”;

$W_{13,11} = W_{7,11}$.

По време на активиране на прехода Z_3 α -ядро постъпва в позиция l_{10} или позиция l_{11} като в позиция l_{10} получава характеристика:

“*key-data, data-value, приложен инструмент за анализ*”,

докато в позиция l_{11} не получава нова характеристика.

Преход 4: Изготвяне на доклад за самооценка и съответните приложения

$$Z_4 = \langle \{ l_{10}, l_{12} \}, \{ l_{13}, l_{14}, l_{15} \}, r_4 \rangle,$$

$$r_4 = \begin{array}{c|ccc} & l_{13} & l_{14} & l_{15} \\ \hline l_{10} & W_{10,13} & W_{10,14} & W_{10,15} \\ l_{12} & true & true & false \end{array}$$

където:

$W_{10,13}$ = “изискана е заявка за изпълнение на аналитична обработка на данни”;

$W_{10,14}$ = “исканите данни са включени в доклада за самооценка на институцията”;

$W_{10,15}$ = “изискана е заявка за данни от хранилището”.

По време на активиране на прехода Z_4 , σ -ядро постъпва през позиция l_{12} с характеристика:

“*ID user, заявка за данни от потребител/приложение*”,

α -ядрото от позиция l_{10} постъпва в позиция l_{13} , l_{14} или l_{15} като получава характеристика съответно:

“*ID user, заявка за аналитична обработка на данни*”;

“*key-data, data-value, включени в доклада за самооценка на институцията*”;

“*ID user, заявка за данни от хранилището*”.

ИЗВОД: Предложен е слабо-централизиран и силно-централизиран модел за осигуряване на качеството и акредитацията във висшето образование. Моделът отчита йерархичната структура на организацията и позволява оптимизиране на ресурсите. Създаден е формален модел на информационните потоци при обработката на данни при самооценяване във висшето училище. Този модел може да бъде развит и чрез използване на йерархичен оператор (НЗ от теорията на ОМ [31, 32]), който замества даден преход или позиция с подмрежа, която има същото, но описано по-детайлно поведение на елементите. Въз основа на създадения обобщеномрежов модел и събраните данни от реални процеси могат да бъдат открити поведенчески модели и анализ на производителността на отделните компоненти.

ГЛАВА 5. РЕШЕНИЯ НА НАУЧНОПРИЛОЖНИ ПРОБЛЕМИ В ОБУЧЕНИЕТО

Резултати от приложения на направените изследвания и анализи в образователна среда са представени в няколко публикации (от списъка на публикациите по дисертационния труд). Методически техники за въвеждане на базови концепции от науката за данните в курсове за студенти, адаптирани към знанията и специалностите на обучаемите са предложени в [148, 167]. Как да следваме съвременните тенденции в курсовете за анализ на данни, чрез решаване на подходящо подобрени реални проблеми и задачи се обсъжда в [151]. Подход за представяне на концепции и стандарти за информационна сигурност във висшето образование се предлага в [157]. Програмиране с реактивни блокове се представя в [150] за курсове по Интернет на нещата. В [203] се анализират трудностите при обучението по дедуктивни методи за проверка и синтез на програми. Тенденции в обучението на софтуерни инженери се представят в [88, 89]. В тази глава от дисертационния труд се обобщават резултати от тези публикации като се предлагат основни елементи на учебни програми и подходи за въвеждане на базови модули и инструменти на науката за данните във Висшите училища.

5.1. Учебна програма за анализ на данните

През последните години има огромен напредък в развитието на технологиите, позволяващи анализ на големи данни, идентифициране на модели и сложни техники за извод. Това налага необходимост от въвеждане на науката за данните във висшето образование. За определяне на съдържанието на преподавания курс, всяко направление и конкретна специалност трябва да се фокусират върху целите, които си поставят с обучението на студентите. От една страна приложните специалности акцентират върху способността на обучаваните да могат да избират подходящ софтуер за анализ на данни. В други специалности акцентът е обучаемите да усвоят основни алгоритми за анализ на данни, машинно обучение и изкуствен интелект. Студентът трябва да може да конструира модели за анализ на съществуваща ситуация и бъдеща прогноза, да се научава как да използва различни техники на изкуствения интелект с цел откриване на аномалии и създаване на оптимални модели. В най-амбициозния план, **след завършване на обучението студентът трябва да може да управлява целия цикъл от данни и да извлича от тях знания с помощта на методи за събиране и анализ на данни и използване на подходящи алгоритми и системи за управление на данните.**

Отчитайки спецификата на обучението в направлението и специалността, учебната програма на дисциплината, свързана с анализ на данните може да акцентира върху различни аспекти. Следва предложение на **базови модули**, които могат да намерят различно място в обучението по дисциплината (независимо от конкретното наименование).

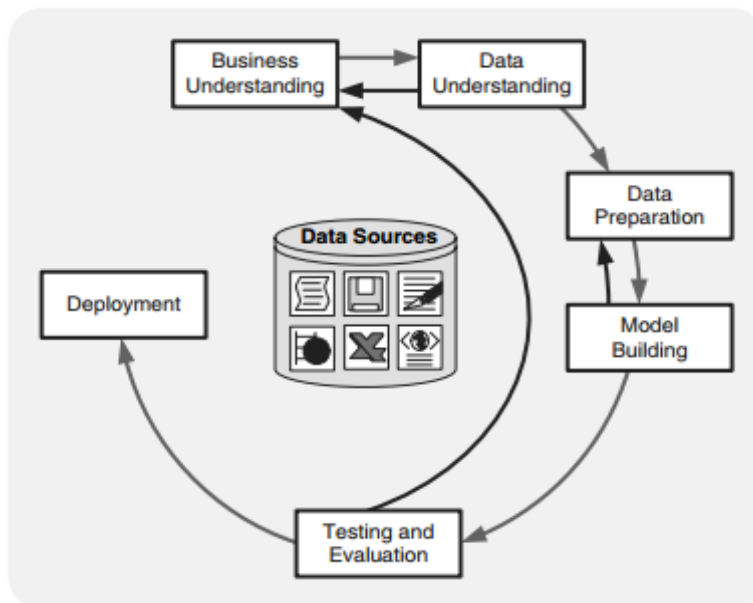
(1) Предварителна подготовка. Студентите трябва да са преминали основните курсове по програмиране, математика, теория на вероятностите и статистически методи, бази от данни, изкуствен интелект, дискретна математика, структури от данни и комуникация, за да се даде възможност на обучаемия да свързва методологиите с

въвежданите нови знания по дисциплината. Въпроси, свързани с постепенното включване на базовите модули в курса по изкуствен интелект са представени в [137].

(2) Встъпителна част. В този модул трябва да се представят основите етапи в цикъла на анализ и извличане на данни и да се направи преглед на различни инструменти, които поддържат всеки етап от цикъла. Целта на модула е да помогне на студентите в бъдеще успешно да се реализират като анализатори на данни или да провеждат научни изследвания и развиват академична кариера.

Всеки процес на извличане на знания се основава на цикъл на работа с данни, завършващ с вземането на решение, като преминава през няколко етапа. Етапите на цикъла са почти еднакви за всяка степен на сложност във всеки сектор и дисциплина. Фиг. 48 представя цикъла на данните, според един от най-широко разпространените стандарти за извличане на знания: Cross-Industry Standard Process for Data Mining - (CRISP-DM) [110]. Процесът на извличане на закономерности от данни се състои от шест основни етапа (фази):

- *Разбиране на проблемната област (Business understanding)* - включва определяне на бизнес цели, оценка на настоящата ситуация, дефиниране на целта и разработване на план за действие. Тук изследователят трябва да разбере, какво точно иска от него потребителя. Често потребителят има множество конкуриращи се цели и ограничения, които трябва да бъдат балансирани по подходящ начин. В края на този етап трябва да се дефинират задачите за търсене на данни и да се направи план за изпълнението им. *Потенциални инструменти:* количествени модели, качествени подходи, математически инструменти и други.



Фигура. 48. Цикъл на данните на процеса Data Mining според стандарта CRISP-DM

- *Разбиране на данните (Data understanding)* - първоначално събиране на данни и след това поредица от дейности, свързани със задълбочено разбиране и анализиране на естеството на данните. На този етап трябва да се идентифицират проблеми по качеството на данните, да бъдат намерени интересните подмножества на данните и да се формират първоначални хипотези за скрита в данните информация. Тук трябва да бъдат

намерени отговори на такива въпроси като: дали данните са пълни? Дали те са коректни или съдържат грешки? Има ли липсващите данни и ако е така, как те са представени? *Потенциални инструменти:* браузъри, индекси, търсачки, международни организации, статистически бюра, техники за извличане на данни и други.

- *Подготовката на данните (Data preparation)* - извличане на данни от различни източници и съхраняването им на достъпно място; валидиране и почистване на данни. Включват се всички дейности по преобразуване на първоначалните „сурови“ данни в крайното множество от данни, които ще бъдат подадени за работа на средствата за моделиране. Този етап често се налага да бъде прилаган многократно. *Потенциални инструменти:* технология за пренос на данни – комуникации, облаци, софтуер за управление на бази от данни, инструменти за валидиране на данни и други.

- *Моделиране (Modeling)* - този етап се състои от избор и прилагане на различни техники за моделиране, целящи извличане на закономерности от данните. *Data mining* е процесът на извличане на валидна и предварително неизвестна информация от различни хранилища с данни и използването ѝ при взимане на решения. *Data Mining* също се разглежда като процес на откриване на шаблони, корелации и тенденции в големи обеми от данни с помощта на статистически и математически средства и методи на изкуствения интелект. Използва методи от различни области като: статистика, невронни мрежи, машинно обучение, генетични алгоритми и др. Този процес е ефективен при наличие на обобщени данни, каквито се съхраняват в складовете от данни (*Data Warehouse*). При работата с малки множества от данни обикновено се използват класически статистически и емпирични методи на изследователски анализ. В случая на големи обеми от данни трябва да се решават редица проблеми, свързани с ефективно съхраняване и търсене на данни или редица въпроси, като: как да бъдат избрани характерни представители на данните, как тези данни могат да бъдат анализирани за кратко време, как да се определи, дали открита връзка между данните отразява действителната реалност или е случайно съвпадение в определена част от данните. *Потенциални инструменти:* алгоритми и инструменти на изкуствения интелект, машинно обучение, програми за обработка на данни, евристики и др.

- *Оценка на модела (Model evaluation)* - на тази стъпка се оценява степента, до която моделът отговаря на целите, поставени от потребителя и се прави опит да се определи, дали съществуват някакви причини от проблемната област, правещи модела неприложим. Резултатите трябва да бъдат сравнени с оценъчния критерий, дефиниран в началото на изследванията. Етапът се състои във внимателно преглеждане на всички стъпки, изпълнени при създаването на конкретния модел, за да се потвърди, че те постигат поставените цели. В края на този етап се приема решение за използване или не на получените резултати. *Потенциални инструменти:* системи за докладване, инструменти за разработване и дизайн, машинно обучение и оценяване и други.

- *Експлоатация на модела (Deployment)* – трябва да се определи процедура по наблюдение и изготвяне на стратегия за експлоатация. Освен това при какви условия и кога да се поднови процедурата по извличане на знания от данни. *Потенциални инструменти:* инструменти за подпомагане на вземането на решения, инструменти за симулация, инструменти за визуализация и други.

Не е необходимо точно спазване на този **шест-фазен процес**, но CRISP-DM осигурява полезна рамка за извличане на данни.

Друга добре известна методика за извличане на знания от данни е разработена от института SAS, наречена SEMMA - Sample, Explore, Modify, Model, Assess. Независимо от модела, нуждите във всеки етап (фаза) са различни и има голямо разнообразие на инструменти, които поддържат процесите.

(3) Основна част на курса. Този модул трябва да осигури задълбочено разбиране на анализа на големи данни, използвайки статистически модели, разпределени изчисления, машинно обучение и евристично програмиране. Основните умения, които трябва да получат обучаваните са свързани с анализиране на данни в отговор на заявки от различни предметни области. Теоретичното обучение представя базовите алгоритми и подходи за анализ на данни като: класификационен анализ, регресионен анализ, асоциативен анализ, клъстерния анализ, анализ на шумове и др. Основен акцент в процеса е прилагането на съответен Data Mining алгоритъм или алгоритми, позволяващи получаването на знания, описващи: връзка между свойства на данните, модели на данните, резултатите от класификацията и клъстеризацията на данните и др. Интересно е какви типове зависимости могат да бъдат открити в процеса на извличане на закономерности от данни. За целта се предлага разглеждане на шест типа задачи, решаването, на които води до получаване на нови зависимости:

- **Задачата за описание и обобщение на данни** (Data Description and Summarisation) цели намирането на описание на основните характеристики на данни, обикновено в обобщена форма. Това позволява на потребителя да получи представа за структурата на данните. В повечето от случаите описанието и обобщението на данни е само една подзадача, изпълнявана на ранните стадии на едно изследване. Началният анализ на данните е свързан предимно с разбиране на природата на данните и построяване на първоначални хипотези за скрита в данните информация. Резултатите от описанието или обобщението на данните могат да бъдат представени във вид на графики, диаграми, таблици, правила и др.

- **Задачата за клъстеризация (сегментация)** търси интересни и смислени подгрупи на данни - клъстери. Елементите в една подгрупа трябва да споделят общи характеристики. Клъстеризацията се прилага и при създаване на таксономии, т.е. организацията на обекти в йерархия от клъстери, групиращи сходните обекти. Сегментацията може да се прилага като самостоятелна задача за определяне на сегменти и да бъде основна цел на *Data Mining* процеса. Но обикновено тя е стъпка от решаването на други задачи и целта е да се намерят еднородни подмножества от данни, които след това да бъдат анализирани. Обикновено при големи обеми от данни сегментиране на данните значително улеснява основната задача за търсене на интересни закономерности.

- **Анализ на крайностите (outlier detection)** - данните могат да съдържат определени обекти, които не отговарят на общия модел на данни. Такива обекти са крайни или екстремни (outliers). Повечето от методите за анализ разглеждат екстремните обекти като шум, който се игнорира. Обаче, в някои приложения, такива като диагностиката или разкриването на измами, рядко срещаните обекти са значително по-интересни от данните подчиняващи се на общи закономерности. Екстремните стойности могат да бъдат намерени чрез използване на статистическите тестове, базираци се на определен вероятностен модел на данни или чрез използване на мерки за разстояние, смятайки че екстремните са тези обекти, които се намират на значително разстояние от всички клъстери. Методите, базираци на отклонение, опитват да идентифицират крайностите чрез анализ на разликите в основните характеристики на обекти в групата.

○ **Задачата за описание на понятия** (*concept description*) цели създаване на разбираеми описания на понятия или класове. Тук целта не е да бъдат създадени пълни модели с висока предсказваща точност, а да бъдат създадени описания на отделни класове, позволяващи да бъде разбрана тяхната структура. Описанията на понятия могат да бъдат използвани и за целите на класификацията. От друга страна, някои класификационни техники създават класификационни модели, които могат да се разглеждат като описание на понятия. Основно различие е, че класификацията цели създаването на пълни модели, т.е. модели, които могат да бъдат приложени към всички данни. Описанието на понятия не е задължително да бъде пълно – достатъчно е то да описва важни части от понятия или класове. Описанията на понятия най-често се представят във вид на правила или прототипи.

○ **Класификация и регресия.** Класификацията предполага наличието на множеството от обекти, описвани чрез определени атрибути и принадлежащи към различни класове. Целевият атрибут приема дискретни (символни) стойности и е известен за всеки обект. Целта е да бъдат построени класификационни модели (класификатори), които назначават коректен клас на по-рано неизвестни обекти - обекти с предварително неизвестна стойност на целевия атрибут. Класификационните модели предимно се използват за предсказващото моделиране. Стойността на целевия атрибут може да бъде зададена предварително (например от потребителя) или изведена от сегментацията. Преди построяване на класификационни модели желателно е да бъде прилаган анализ на изключения, който позволява те да бъдат премахнати и по този начин да бъде подобрен класификационният модел. Целта на предсказването е да се намери числовата стойност на целевия атрибут за по-рано неизвестни обекти и този тип задачи наричаме регресия. Когато предсказването работи с данни за времеви серии, тази задача носи название предвиждане (*forecasting*) [110]. Регресионните модели могат да бъдат представени във вид на регресионни дървета, невронни мрежи, генетични алгоритми и т.н.

○ **Анализът на зависимостите** се състои в намирането на модел, който описва важни зависимости (или асоциации) между отделни елементи от данни или събития. Зависимостите могат да бъдат точни или вероятности [131].

- Асоциациите са специален случай на зависимостите, те описват сходството между елементите на данни, т.е. тези елементи: двойки атрибут-стойност, които често се срещат заедно. Типичното приложение на асоциациите е анализ на потребителските кошници. Така следното асоциативно правило “В 30% от всички покупки бира и фъстъци са били закупени заедно” е типичен пример на една асоциация. Алгоритмите за намиране на асоциации са много бързи и извеждат много асоциации. Избор на най-интересните от тях е предмет на интензивна изследователска работа.

- Времеви серии (редове) са специален вид зависимости, извлечени от данни, представляващи последователности от събития във времето. Така например в областта на анализа на потребителските кошници последователните шаблони описват покупките на определен потребител или група от потребители във времето.

За извличане и представяне на моделите данни се използват различни математически средства и методи на изкуствения интелект като: невронни мрежи, машинно обучение, генетични алгоритми, дървета на решенията и др. Най-често използваните статистически средства са: групиране на данни, статистика, пресмятане на средни величини и отклонение, честотни разпределения и др. Моделите създавани от съответните средства представят компресиран образ за входните данни и съдържат

основните познания, извлечени от данните на базата на определящия алгоритъм [118]. Основни методи, използвани от техниките за извличане на закономерности от данни са: метод на поддържащите вектори (Support Vector Machines SVM), метод на наивния Бейс (Naïve Bayes), разлагане по метода на главните компоненти (Principal Component Analysis), метод на изкуствена невронна мрежа за анализ на данни (Artificial Neural Networks) и др.

(4) Практическо обучение – обучаемите експериментират с извличане на данни, управление на данни и анализ, подходи за визуализиране на данни, чрез прилагане на различни системи с отворен код. В курса следва да се представят популярни системи за обработване на данни и извличане на знания като например:

- *WEKA* – Waikato Environment for Knowledge Analysis е система за свободно ползване, лицензиран под GNU General Public License [213].

- *RapidMiner* - софтуерна платформа, която предоставя интегрирана среда за машинно обучение, извличане на данни, извличане на текст, средства за прогнозен и бизнес анализ, визуализация на резултатите, валидиране и оптимизация [176].

- *KNIME* - платформа за анализ на данни с отворен код. Интегрира различни компоненти за машинно обучение и извличане на данни като позволява на потребителите да създават визуално потоци (или конвейри) [125].

- *Orange* - софтуер с отворен код за машинно обучение и извличане на данни, написан на Python [139]. Orange се поддържа на MAC-OS, Windows и Linux. Предлага богат набор от инструменти за извличане анализирани и визуализирани на данни.

- *Excel* се използва успешно за подготовка на данни и за визуализиране на резултатите чрез различни графики. Статистическите функции на Excel дават богати възможности за анализ на данните. Използването на макроси позволява бърза и многократна обработка на групи от данни.

(5) Курсов проект. Студентите с по-слаби умения по програмиране и недостатъчни математически знания ще изпитват редица трудности при разбирането на същността на анализа на данните и прилагането на избраната техника. За преодоляване на трудностите се предлага като последен модул на курса включване на проектно-базирано обучение. Стратегическата цел в тези часове е въвличането на студентите в изследване и овладяване на редица умения като: да извършват изследователска и практическа работа в конкретна предметна област; да се научат да идентифицират етапите в разработването на проект; да планират дейността си по часове; да спазват планираните срокове; да работят съвместно с други членове на екипа; да самооценяват действията си; да участват в дискусии по темата на проекта, като формират и защитават аргументирано собствени идеи и умения и др.

Курсовите проекти могат да бъдат индивидуални или екипни, но със специфична ориентация към определена бизнес област. Например:

- организационния и бизнес анализ на данните: финансови данни, застраховане, банкиране, управление на инвестиции, управление на риска, пазарни данни и др.;

- анализа на данните за общественото здраве: данни за качеството на живот на човека, изграждане на модели за качествен живот, предотвратяване на бъдещи заболявания и епидемии и формулиране на програми за управление на здравето в общността;

- управлението на публичните институции, болници, държавни институции и др.;

- анализ на данни в интелигентни градове и кибер-сигурност.

Независимо дали целта е да се открият интересни взаимовръзки, да се категоризират обекти в групи, да се оптимизира планирането на ресурси или да се определят тарифи за таксуване, работата ще помогне на студентите да получат основното разбиране на техниките за анализ на данни и да усвоят умения за извличане на знания с конкретни задачи.

ИЗВОД: Развитие на такива академични програми е от решаващо значение за успеха на обучаемите като бъдещи изследователи в дигиталния свят. Интересът на обучаемите и активното им участие в процеса на обучение могат да бъдат повлияни от качеството на обучаващия процес и използваната учебна програма.

5.2. Инструменти за извличане на знания чрез примери

Обучението в курсовете по Базис на данни на студентите от направление *Информатика и компютърни науки* в повечето български университети се провежда през втори или трети семестър. За целта последователно се въвеждат: основни модели на данните; процес на проектиране на релационни бази от данни, анализ и нормализация на релационни схеми, езици за описание и обработка на данни. Студентите се запознават с проблемите при разпределени бази от данни и обектно-ориентирани бази от данни и новите перспективи за развитие на големите данни (BigData). След като реализират редица приложения под формата на курсови проекти, студентите преминават към анализ на хранилища от данни и Data mining техники. Тези техники предлагат богати възможности за изследване на данните, но изучаването им предполага силна математическа основа. Имайки предвид ограниченото време, в курсовете тук се предлага разкриване на възможностите на тези техники чрез конкретни примери върху предварително подготвени данни.

Студентите могат да анализират данните, следвайки фазите на CRISP-DM модела, като в последователност от шест стъпки се следват шестте основни етапа в процеса на извличане на закономерности от данни. В резултат могат да се достигнат до изводи, като например:

- да анализират данните за потребителите и да предскажат кредитен риск за нови потребители, базирайки се на техния доход, възраст и предходна кредитна информация;
- да определят множеството от стоки, които често се продават заедно;
- да правят прогнози за потребителското търсене на определени продукти при намаление на цените; за обема на продажбите за даден период от време;
- да разкрият отклонения, например стоки, чиито продажби са извън очакванията в сравнение с предходна година и др.

Следва описание на методологията за запознаване на студентите с основни Data mining функционалности, базирайки се на примери [151]. За целта се използват подходящо подбрани реални задачи за анализ на данни, които са адаптирани с цел по-лесно прилагане на съответен алгоритъм. Решението на всеки проблем се представя като последователност от шест стъпки, съответстващи на фазите на стандарта за извличане на знания: Cross-Industry Standard Process for Data Mining (CRISP-DM).

5.2.1. Класификация и прогнозиране

При задачата за класификация е зададено крайно множество от обекти, за които е известно към кои класове принадлежат, както и крайно множество от обекти, за които класовата принадлежност е неизвестна. Трябва да се построи модел, който определя произволен обект към съответен клас, като укаже стойността на търсения атрибут (целевия атрибут). Ако целевият атрибут има само две възможности стойности е “бинарен” класификационен проблем, в случаите на повече възможни стойности е “многокласов” класификационен проблем. Най-общо алгоритъмът създава серия от случайни правила, работейки върху тренировъчните данни и се избират тези, които отговарят най-добре на данните. Процесът се повтаря определен брой пъти, след това правилата се проверяват чрез множеството на тестовите данни.

Следва примерен проблем за прилагане на класификационен шест стъпков алгоритъм, като обобщение на пример от [110].

(1) стъпка. Дефиниране на проблема: “*Как банка да намали загубата на свои клиенти?*”.

Банката иска да научи повече за тези клиенти, които закриват своите сметки и да задържи съществуващите клиенти чрез подходящи мерки. В случая целевият атрибут, чиято стойност ще се предвижда е: “вероятен за изгубване клиент (*Attrite*)” с възможни стойности – *true/false*.

(2) стъпка. Тук се избират данните за клиентите, събрани от техните транзакции. Използвайки класификационна функция може да се определи профилът на потребителите. По-доброто разбиране на поведението на клиентите, позволява на банката да разработи бизнес план за задържането им.

(3) стъпка. Определяне на данните – използваме таблица CUSTOMER от базата от данни. Тук се определят атрибутите, които ще се използват от алгоритъма. Може да се въведе и подмножество от клиенти. При изследване на стойностите на атрибутите могат да се открият отклонения в данните, например има няколко клиента с печалби над 1 милион долара и тези стойности могат да изкривят резултатите, затова се взима решение как да се третират – като липсващи или като действителни данни.

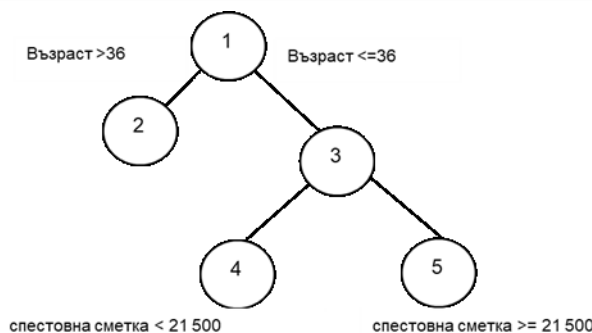
(4) стъпка. На тази стъпка първо се описват няколко подходящи моделиращи техники, а след това работата с тях.

(4.1) *Дърво на решенията* – графичен метод за избор на алтернатива чрез изследване на последователни и взаимно свързани решения и резултатите от тях. Разглежда се подробно в курса по изкуствен интелект. Тук техниката се прилага като всеки възел разделя данните на база условие върху атрибута. В примера, коренът представя всички n-потребители от таблицата CUSTOMER. За тези потребители относно първия атрибут “възраст”, алгоритъмът намира Мах(възраст), за която целевият атрибут има стойност “Non-attrite” (не е вероятен за изгубване клиент). Така първият възел разделя данните на групи във възли 2 и 3, на база условие върху стойностите на атрибута възраст на потребителя. После възел 3 разделя данните на групи във възли 4 и 5, базирани на

условие, относно стойностите на атрибута “баланс на спестовната сметка” и т.н. Така използвайки данните от таблицата, алгоритъмът построява дърво на решенията.

С всеки възел в дървото има свързано правило, което предсказва целевата стойност с определени “сигурност” (confidence) и “подкрепа” (support). “Сигурност” - мярка на вероятността, с която възела на дървото предсказва целевата стойност. Това е съотношението на броя на случаите във възела, в които условието е предсказано вярно към общия брой случаи във възела. “Подкрепа” - съотношението на броя на случаите във възела, в които условието е предсказано вярно към общия брой случаи в таблицата, т.е. мярка – колко случаи от базата са свързани с условието. Например:

| Възел | Правило | Целева стойност | Брой случаи | Сигурност | Подкрепа |
|-------|--------------------------------|-----------------|-------------|-----------|-----------|
| 1 | | Attrite | 10 | 5/10=0.5 | 5/10=0.5 |
| 2 | Възраст > 36 | Attrite | 3 | 3/3 = 1 | 3/10=0.3 |
| 3 | Възраст ≤ 36 | Non-attrite | 7 | 5/7=0.71 | 7/10=0.7 |
| 4 | Възраст ≤ 36 и сметка < 21 500 | Non-attrite | 5 | 5/5 = 1 | 5/10=0.5 |
| 5 | Възраст ≤ 36 и сметка ≥ 21 500 | Attrite | 2 | 2/2 = 1 | 2/10 =0.2 |



Фиг. 49. Примерна извадка от данни и съответно дърво на решението

За дървото на решението от фигура 49, възел 2 има 3 случая, които удовлетворяват условието “възраст”>36 и всички те имат целева стойност „Attrite”, сигурност = 3/3=1 или 100%, затова няма ново разделяне на възел 2.

Възел 3 може да бъде разделен, защото сигурност = 5/7 < 1 и сигурността може да бъде подобрена чрез използване на друг атрибут като “баланс на спестовната сметка”.

Възли 2, 4 и 5 са листа, сигурност = 1.

Могат да се стопират критериите чрез задаване на максимална дълбочина или др.

Студентите трябва да извеждат справка за всеки възел на дървото: свързаното с него правило, стойността на целевата функция, броя на случаите във възела и мерките “сигурност” (confidence) и “подкрепа” (support).

(4.2) *Класификационен алгоритъм, базиран на теоремата на Bayes.* Разглеждат се комбинациите на целевия атрибут с другите атрибути, например разглеждат се двата атрибута “възраст” и “баланс на спестовна сметка” от таблица CUSTOMER и целевият атрибут “Attrite” (възможен за изгубване клиент). На базата на класификационните правила, данните за клиентите се разделят на групи: група-1 съдържа данни за клиенти на

възраст по-малка от 35, а група-2 – за клиенти на възраст по-голяма или равна на 35 години.

По отношение на атрибута “баланс на спестовна сметка”, данните също се разделят в две групи: група-1 съдържа данни за клиенти със стойност на баланса на спестовна им сметка по-малка от 20 000, а група-2 за клиенти със стойност на баланса на спестовната им сметка по-голяма или равна на 20000.

Алгоритъмът изчислява първо вероятността за удовлетворяване на целевата стойност за всеки отделен атрибут, на базата на наличните данни в таблицата. А за всеки конкретен случай за предвиждане се изчислява произведение на условните вероятности. Така в примера за двата атрибута се пресмятат 8-те възможни вероятности:

$$\begin{array}{ll}
 P(\text{възраст} < 35 / \text{Attrite}); & P(\text{сметка} < 20\,000 / \text{Attrite}); \\
 P(\text{възраст} < 35 / \text{Non-Attrite}); & P(\text{сметка} < 20\,000 / \text{Non-Attrite}); \\
 P(\text{възраст} \geq 35 / \text{Attrite}); & P(\text{сметка} \geq 20\,000 / \text{Attrite}); \\
 P(\text{възраст} \geq 35 / \text{Non-Attrite}); & P(\text{сметка} \geq 20\,000 / \text{Non-Attrite});
 \end{array}$$

За един нов потребител, който е на 25 години и има спестовна сметка = 13 300 се изчисляват вероятностите за целевия атрибут като:

$$\begin{aligned}
 &P(\text{Attrite}/\text{възраст}=25 \text{ and } \text{сметка}=13300) = \\
 &\quad P(\text{възраст}<35/\text{Attrite}) \times P(\text{сметка}<20000/\text{Attrite}) \times P(\text{Attrite}) \\
 &P(\text{Non-Attrite}/\text{възраст}=25 \text{ and } \text{сметка}=13300) = \\
 &\quad P(\text{възраст}<35/\text{Non-Attrite}) \times P(\text{сметка}<20000/\text{Non-Attrite}) \times P(\text{Non-Attrite})
 \end{aligned}$$

тогава в зависимост от получените при пресмятането стойности се определя:

$$P(\text{Non-Attrite}) > P(\text{Attrite}) \text{ е в сила или не е в сила.}$$

(4.3) *Невронна мрежа* – представя идеята за математическия аналог на биологичната невронна мрежа, чрез взаимосвързани изчислителни елементи – неврони. Всеки неврон приема сигнали под формата на числа, сумира ги и сумата минава през активационна функция, която изчислява състоянието на всеки неврон. В примера използваме сигмоидалната функция $y=f(x)=1/(1+e^{-x})$. Така се определя активацията на неврона (степен на възбуда), която се предава по изходящите връзки към другите неврони. Невронната мрежа [39] се представя като свързана група от симулационни неврони, които представят изчислителния модел на информационен процес. Един *симулационен неврон* е математически модел, който може да получава един или няколко входа и произвежда изход. Изходът се изчислява чрез умножаване на всеки вход по съответно “тегло” и ги комбинира в съответен изход като се прилага определената функция. Входното ниво има толкова неврона, колкото са входните атрибути, а изходното ниво – колкото е броят на целевите стойности. За да бъде една невронна мрежа приложима към даден проблем, тя трябва да бъде предварително обучена.

Невронната мрежа се обучава чрез промяна на теглата на връзките между невроните. Това се реализира чрез правила, които определят как се променят теглата. Най-използван е *методът на обратното разпространение на сигнала за грешка* (back-propagation), като при този метод за всеки изходен неврон се изчислява разликата от

полученото с желаното състояние и се формира сигнал за грешка. Сигналът се движи назад към входния слой и променя теглата на връзките. При следващата активация на мрежата грешката е по-малка. Обучението на мрежата е чрез итеративен процес, сравнявайки предикатите на невроните за всеки един случай с действителните целеви стойности от данните за обучение.

(5) стъпка. Важно е да се оцени качеството на модела, преди използването му за прогнозиране. С конкретни примери се въвеждат тестови метрики при класификационните модели:

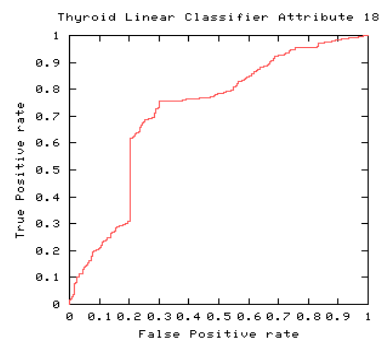
Confusion matrix (матрица на хауса). В примера допускаме, че от 910 коректни предвиждания – 750 са “Non-Attrite”, а 160 – “Attrite”, а от грешни 90 случая – 30 са предвидени като “Non-Attrite”, но не са; а 60 са предвидени като “Attrite”, но не са. Построяваме матрицата на хауса като двумерна NxN таблица, в която записваме броя на верните и неверни предвиждания на класификационния модел върху тестовите данни. N е броят на целевите стойности – в случая Attrite и Non-Attrite:

| | Attrite | Non- Attrite |
|-------------|---------|--------------|
| Attrite | 160 | 30 (FN) |
| Non-Attrite | 60 (FP) | 750 |

така матрицата на хауса показва къде моделът предвижда неправилно: FP – false positive, FN - false negative. *Тегловата матрица* е таблицата, в която се определят разходите, свързани с неправилни прогнози.

Прогнозна точност – изчисляваме я чрез сравняване на предвижданията и действителните целеви стойности. Нека в примера имаме 1000 случая и класификационният алгоритъм предвиди правилно 910 от случаите. Точността на предвиждането е $910/1000 = 0.91 = 91\%$. Точността представя вероятността, предвиждането да е правилно и се свързва с модела.

Приемник на функционалните характеристики ROC (receiver operating characteristic) - друг начин за оценка на качеството на класификационния модел. Тази графика представя FP (false positive) състоянията по X-оста, TP (true positive) състоянията по Y-оста. Колкото по-голяма е площта под кривата ROC, като цяло, толкова по-голяма е ефективността на модела [39].



(6) стъпка. След оценяването на модела, използвайки тестови данни, се избира най-добрият модел (спрямо точност и цена) и се прилага за предвиждане на целеви стойности. На тази стъпка студентите пресмятат цената и качеството на модела като например: разглеждат случая (Confusion matrix), където за целевият атрибут “Attrite” има 60 грешни предвиждания (FP - предвидени като “Attrite”, но не са) и 30 грешни предвиждания (FN – предвидени като “Non-Attrite”, но не са). Нека в тегловата матрица е зададено, че цената на пресмятане на FN е три пъти по-висока от тази на FP, тогава общата цена за грешните предвиждания (FN и FP) $3 \times 30 + 1 \times 60 = 150$.

Прилагането на модела може да става при различни случаи, например при нов клиент или когато клиент извърши трансфер на голяма сума от своята сметка в сметка на друга банка. Тогава приложението може да преизчисли предвиждането при новите данни.

5.2.2. Регресионен анализ

Този анализ изучава и оценява възможните функционални зависимости между две или повече величини. Основни въпроси са дали съществува функционална зависимост между величини и ако е така, да се намери функция $y = f(x_1, x_2, \dots, x_p)$, която дава израз на ефектите на отделните променливи x_1, x_2, \dots, x_p върху променливата y .

Следва обобщен шест стъпков алгоритъм, за работа със студентите в часовете [110].

(1) стъпка. Дефиниране на проблем: Банка иска да автоматизира процедурата по оценка на недвижими имоти. За тази цел се търси цената на недвижим имот, използвайки регресионен анализ върху данните за имотите. Това ще намали времето при отпускане на ипотечни заеми.

(2) стъпка. Използване на данни, събрани в години за оценки на недвижими имоти.

(3) стъпка. Определяне на данните: таблица “Real-estate-data” от базата от данни. Тази таблица има следните атрибути: city, country, state, house_size, year_build, land_size, number_of_rooms, garages, pool_type, number_of_schools, school_rating_avg, crime_rate. Целевият атрибут е *цена на имота*.

(4) стъпка. Алгоритмите, използвани при класификационния анализ могат да бъдат прилагани и при регресия, като например: *невронни мрежи* и *дърво на решенията*.

(5) стъпка. За оценка на качеството на регресионните модели се пресмята нарастването на грешките. Има различни математически метрики, които могат да бъдат използвани за определяне на грешката. В часовете се прилага: средна квадратична грешка на предвиждането, като изчислява корен квадратен от сумата на квадратите на грешките, разделена на броя им и абсолютна грешка, като сума на абсолютните стойности на грешките, разделени на броя им.

(6) стъпка. След намиране на модел с минимална грешка, той може да се прилага към нови данни за предвиждане и се пресмятат: предвиждаща точност и степен на доверие, свързани с модела. Степента на доверие се представя като число между 0 и 1 или като число в интервала 0 – 100, като най-ниската степен на доверие е 0, а на-високата е 1 или 100. За разлика от класификационния анализ, (който може да изведе повече от една стойност), регресията произвежда една стойност, защото целевият атрибут е непрекъснат.

5.2.3. Асоциативен анализ

Асоциативният анализ изучава честотата на едновременно появяване на факти. Връзките между фактите се откриват и измерват чрез обучаващи шаблони от по-рано натрупани данни. Този анализ е свързан с откриване на “асоциативни правила”, задаващи условия за стойностите на атрибутите, които се явяват често заедно в дадено множество от данни. Асоциативните правила имат вида:

$X \Rightarrow Y$, където X е условна (предшестваща) част, а Y е логическото следствие (резултатна част).

Асоционното правило $X \Rightarrow Y$ се интерпретира като: “записите в БД, които удовлетворяват условията в X , вероятно удовлетворяват и условията в Y ”. Отново се въвеждат мерките:

Сигурност (confidence) на едно правило: $P(X \text{ and } Y) / P(X)$.

Подкрепа (support) на правилото: $P(X \text{ and } Y)$ / брой записи.

Lift е отношението между сигурността на правилото (confidence) и неговата очаквана сигурност (expected confidence). Отражава честотата на логическата последователност в данните.

Очакваната сигурност се измерва като брой случаи, в които следствието е удовлетворено към общия брой на случаите в базата, т.е. $P(Y)$ / брой записи в базата.

Следва шест стъпков обобщен алгоритъм за прилагане на асоциативен анализ към данни, адаптация на пример [110].

(1) стъпка. Дефиниране на проблем: “Как да се определят свързаните услуги за клиентите?”. Когато клиентът ползва комплекс от услуги, то е по-малко вероятно той да се прехвърли в друга банка, ако няма друга причина. Затова нека банката се интересува кое множество от услуги се ползва заедно от клиентите и да ги препоръчва комплексно или да допълни липсващите услуги в предлагането си.

(2) стъпка. Банката има събрани данни за услугите, които всеки клиент ползва и техните характеристики като: “среден баланс на разплащателната сметка”, “баланс на спестовна сметка” и др. Търсят се свързани услуги, като например: ако клиент има една сметка в банката, то какви други сметки или финансови услуги биха могли да го интересуват.

(3) стъпка. Определяне на данните в таблиците “Customer” и “Products” от базата от данни. Нека разгледаме пример с потребител и няколко вида сметки: разплащателна (Checking Account), спестовна (Saving Account), кредитен сертификат (Certificate Account) и др.

(4) стъпка. Извеждане на асоциативни правила. Ако един клиент ползва няколко услуги, то за него има няколко отделни записа в таблицата в транзакционен формат. Примерно асоциативно правило:

Потребителят X има *Saving Account*

→ Потребителят X има *Checking Account*.

(5 и 6) стъпки. Оценяване и прилагане на модела. На практика данните могат да съдържат хиляди транзакции и моделът може да получи голям брой правила. Обикновено се използва праг, който ограничава броя им и позволява на потребителя да се фокусира върху подмножество от правила. Праговете могат да се задават за една или повече метрики като: *support*, *confidence*, *lift*.

Примери за асоциации между атрибути, които студентите извеждат върху друга примерната база от данни за продажби на стоки са:

Age(X , “20-29”) and Income(X , “20-30K”)

\Rightarrow Buys(X , “Gaming Mouse”), [support = 2%, confidence = 60%]

тази асоциация означава, че клиенти на възраст между 20 и 29 години, с доход от 20-30 хиляди, които са 2% от общия брой клиенти в базата от данни, са купувачи на “ Gaming Mouse”. В базата имаме 60% вероятност, че клиент в тази възраст и доход е в групата на купувачите на “ Gaming Mouse”.

Докато асоциацията:

Contains(T, “computer”) => Contains(T, “software”), [support = 1%; confidence = 50%], означава, че ако транзакция Т съдържа “computer”, 50% е вероятността тя да съдържа и “software”, което е 1% от всички транзакции.

5.2.4. Клъстерен анализ

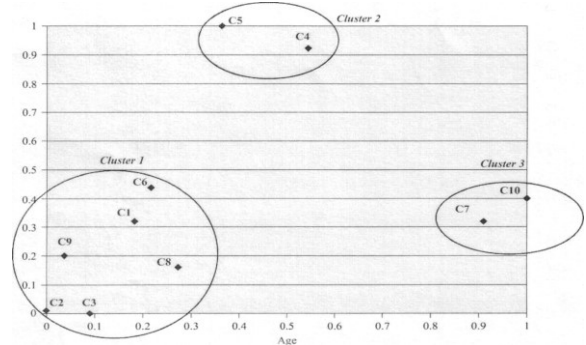
При клъстерният анализ целта е n на брой обекта да се групират в k на брой групи (клъстери), като се използват p на брой признаци (променливи). Така клъстерът се формира от подобни обекти, независимо от техните класове. Стремещт е разкриване на евентуално скрита групировка на обектите. Важно деление на клъстеризационните процедури е в зависимост от това, дали предварително се задава или не се задава броя на клъстерите. Голямото разнообразие на процедурите се поражда от използваните правила за създаване на клъстерите [214].

Шест стъпков обобщен алгоритъм:

(1) стъпка. Дефиниране на проблем: Банка иска да раздели потребителите на групи според нуждите им, за да ги използва в кампаниите за увеличаване на приходите и пазарното си участие. Целта е намиране на групи от сходни помежду си клиенти.

(2) стъпка. Използвайки данните за потребителите, от таблицата “Customer” от базата от данни, се вземат двата атрибута: възраст (age) и доход (income).

(3) стъпка. Как да се измерят сходните помежду им случаи? Числовите атрибути могат да имат различни скали, например възрастта на клиентите е от 15 до 100 години, а доходът им е от \$5 000 до \$130 000. Техните стойности могат да бъдат нормализирани и доведени до една и съща скала.



В примера, който представяме на студентите за пресмятане на разстоянията между случаите се използва евклидово разстояние, а като функция за сравнение – абсолютна стойност на разликата.

(4) стъпка. В случая с анализ на два атрибута, резултатът се визуализира като двумерна диаграма. Но клъстеризацията може да включи десетки атрибути, което изисква алтернативни техники за представяне.

Всеки клъстер в клъстерния модел има свързани с него статистики и правила за описание на клъстера. Правилото за описание на клъстер 1 от примера има вида:

Rule: IF age >= 15 and age <= 31.5 and income >= 5000 and income <= 67500
THEN Cluster = 1 [Support = 6/10 = 0.6]

(5 и 6) стъпки. Оценяване и прилагане на модела. При въвеждане на нови случаи се изчислява разстоянието до центъра на всеки клъстер и чрез сравняване на разстоянията се определя клъстерът, към който случаят принадлежи.

Други примери, подходящи при обучението на студентите:

- разполагане на няколко АТМ-автомата в един град;
- изследване на архитектурните типове и стилове на жилищни сгради, според културата и географските особености на регионите и др.

5.2.5. Анализ на шумове

В данните обикновено има обекти, които се отклоняват от основното поведение или модел на данните. Те се явяват отклонения (Outliers) и се определят от Data Mining системите като шумове.

Отклоненията са разликите между измерените стойности и съответните очаквания на базата на предишни или нормативни стойности. Една Data Mining система може да се интересува от откриване на множество от отклонения и да опише характеристиките им. Освен това може да обясни причината и дори да предложи действия за довеждане на стойностите към техните очаквания.

Примерен проблем: намалението на общите продажби за последния месец в сравнение с продажбите за този месец през предходната година е шаблон на отклонение. След като открият зависимост, студентите се опитват да обяснят отклонението, базирайки се на данните (например, компанията е имала по-голям персонал по продажбите през предходната година в сравнение със същия период през текущата година).

Други примерни теми, разработвани от студентите по време на обучението:

- да се направи описание на общите характеристики на потребителите, които харчат повече от 1000\$ годишно за електроника (от примерна база от данни). В резултат трябва да се изведе общ профил на потребителите като например: 40-50 годишна възраст, щатен служител на заплата с отличен кредитен рейтинг и т.н.
- да се сравнят две групи клиенти: тези които пазаруват компютърни продукти повече от 4 пъти на месец и тези, които рядко купуват такива продукти (по-малко от 3 пъти годишно). Очакван резултат от описанието е сравнителен профил на потребителите като например: 80% от редовните клиенти са между 20 – 40 годишна възраст и са с висше образование. От друга страна 60% от клиентите, които рядко купуват такива продукти са или по-възрастни или по-млади, нямат висше образование и т.н.
- да се сравнят общите характеристики на софтуерните продукти, чиито продажби са се увеличили с 10% през последната година с тези, чиито продажби са намалели най-малко с 30% през същия период.

ИЗВОД: Анализът на резултатите от проведеното обучение показва, повишаване степента на усвоеност на учебното съдържание и повишаване на интереса към дисциплината [151]. Обучението по тематика не предизвика сериозни трудности сред студентите, които притежават необходимата предварителна подготовка. Студентите с по-слаби умения по програмиране и не достатъчни математически знания изпитват редица трудности при разбирането на същността на анализа и прилагането на избраната техника.

5.3. Въвеждане на онтологии в обучението на студенти

Обучението на студентите от направление „Информатика и компютърни науки“ в областта на онтологията се провежда главно в дисциплините, свързани с Изкуствен интелект и представяне на знанията през третия, а за някои специалности в четвъртия семестър.

Въвеждането на понятието „онтология“ започва със запознаване на неговия произход от философията и отразява природата на нещата, които реално съществуват. С този термин в съвременната философска литература се **означава система от категории, които са следствие от определена система от възгледи (определена гледна точка) за света**. В зависимост от различни класификационни признаци се представя тяхна класификация. В [95] се представя класификация на онтологията на: общи онтологии, ориентирани към конкретна област, ориентирани към конкретна задача и приложни онтологии.

- Общите онтологии са от най-високо ниво на абстракция и са предметно-независими. Те отразяват общите категории (концепти), които са независими от конкретната задача или област като: пространство, време, обект, събитие, действие, количество, мярка и т.н. Тези онтологии трябва да са приложими за всяка специализирана област и това ги различава от другите типове онтологии.

- Предметните онтологии и онтологията на задачи описват речник, свързан с предметната област (търговия, медицина и др.) или речник, свързан с конкретната дейност (продажби, диагностика и др.) и обикновено са специализация на термините, въведени в обща онтология.

- Приложните онтологии представят концепти, зависещи от конкретна предметна област и от дейностите и решаваните задачи. Обикновено концептите в тези онтологии съответстват на ролята, които обектите заемат в предметната област при изпълнение на дейност.

Може да заключим, че с онтологията се описва реалността и са свързани със специфична област (домейн на интерес). Разработват се за отделяне на знанията за домейна от оперативните знания, с цел анализ и преизползване на знанията [205].

По отношение на структурата на онтологията явно се отделят два компонента: имена на съществуващи концепции и връзки в домейна. Редица ограничения могат да бъдат наложени над домейните. Онтологията заедно с множество конкретни инстанции на класовете съставя база от знания (knowledge base). Така разработването на една онтология преминава през следните фази [91]: дефиниране на класовете в онтологията; изграждане на йерархия от класовете; дефиниране на характеристики и описание на възможните стойности за тях; попълване на конкретни стойности за характеристиките, за да се получат конкретни инстанции.

Разработването на онтология е итеративен процес. Основно правило е концепциите в онтологията да са близки до обектите и връзките в конкретния домейн. В началото на процеса, свързан с изграждане на онтология трябва да се определи каква е целта, как ще се използва - за какви въпроси се предполага, че ще се търсят отговори, с помощта на онтологията. Обучението продължава с изучаване на: методология за проектиране на онтология; изучаване на Web Ontology Language (OWL) и прилагане на езика за изграждане на примерна онтология; използване на Reasoner за работа с онтологии.

5.3.1. Методология за проектиране на онтология

За по-доброто разбиране на онтологиите от студентите, при проектирането на онтология първо следва да се разясняват основните принципи:

- формализация – описание на предметната област в единни, строго определени термини и модели;
- използване на ограничено количество базови понятия, на основата, на които се конструират останалите понятия;
- пълнота и непротиворечивост на понятията в онтологията;
- прилагане на инструменти, осигуряващи възможности за редактиране, визуализация, документиране, импорт и експорт на онтологии.

Основен подход, използван в процеса на обучението по проектиране и създаване на онтологии е чрез примерни задачи. В часовете, студентите **разработват конкретен пример, като следват последователно стъпките:**

1. Определяне на целите и областта на приложение на онтологията;
2. Разглеждане и обсъждане на съществуващи преизползваеми онтологии;
3. Изброяване на основните термини (концепции) в онтологията;
4. Дефиниране на класовете и на йерархията от класове;
5. Дефиниране на характеристиките (slots) на класовете;
6. Дефиниране на аспектите на характеристиките;
7. Създаване на инстанции на класовете.

5.3.2. Прилагане на езика OWL за изграждане структурата на онтология

За изграждане на онтологията се използва OWL – език за дефиниране на Web онтологии [127]. Онтология, изградена чрез OWL се означава като OWL онтология и включва описание на класове, свойства на класове и техните инстанции. Следва кратко описание на компонентите на онтология, изградена чрез OWL:

- *Individuals*: индивидуалностите или обектите са инстанции на клас;
- *Classes*: класовете са конкретно представяне на концепциите;
- *Attributes (properties, slots)*: характеристики на обектите и класовете. Известни са още като роли в дескриптивната логика;
- *Relations*: начините, по които класовете и обектите могат да бъдат свързани;
- *Function terms*: сложни структури, формирани от конкретни връзки, които могат да бъдат използвани на мястото на индивидуален член в твърдение;
- *Restrictions*: формално описание на това какво трябва да е истина в дадено твърдение, за да бъде прието като вход;
- *Rules*: *if-then* изрази, описващи логическото заключение, което може да бъде извлечено от твърдение в конкретна форма;
- *Axioms*: твърдения (включително правила) в логическа форма, които заедно обхващат цялата теория, която онтологията описва в нейния домейн на приложение;
- *Events*: задават промяната на атрибути или връзки (*attributes, relations*).

OWL спецификацията дефинира подход за извършване на логически изводи и получаване на факти, които не са явно представени в онтологията, а са следствие от

семантиката. Чрез OWL механизмите могат да се правят изводи, чрез комбиниране на множество факти и правила.

За целите на обучението се разработва конкретен пример за OWL онтология като се използва софтуера Protégé [174]. Този софтуер е създаден от специалисти по медицинска информатика в Станфордския университет. Той е свободно разпространявана Java програма, предназначена за създаване на онтологии. Включва редактор на онтологии, предлага графичен интерфейс, възможности за извеждане на справки и примери. Поддържа езика OWL за представяне на структурата на онтологията.

Конкретни дейности, изпълнявани от студентите чрез софтуера Protégé са:

1. Създаване на класове. След като бъдат създадени класовете например: *Student* и *Teacher*, можем да ги направим *disjoint* (така обект не може да бъде инстанция на повече от един от тези два класа).

2. Създаване на йерархия от класове. Създаване на подкласовете на *Teacher* – *Lectures* и *Assistants*.

3. Дефиниране на OWL *Properties*, които представят връзки на индивидуалности. Следва представяне на двата основни вида *properties*:

- *Object properties* – свързват индивидуалност с индивидуалност;

- *Data properties* – свързват индивидуалност със стойност (XML или RDF literal).

4. Следващата стъпка е въвеждане на OWL характеристики на *object properties*:

- Всяко *object property* може да има инверсно (*inverse*) *property*. Ако някое *property* свързва A с B, то тогава неговото инверсно свързва B с A.

- Функционално (*functional*) *property* – за дадена индивидуалност има най-много една индивидуалност, свързана чрез това *property*.

Пример: Ако *hasBirthMother* е *functional* и

Ivan *hasBirthMother* Maria и Ivan *hasBirthMother* Neli => Maria \equiv Neli

- Ако едно *Property* е инверсно функционално (*inverse functional*), то инверсното *property* е функционално.

Пример: Neli *isBirthMotherOf* Ivan и

Maria *isBirthMotherOf* Ivan => Maria \equiv Neli

- Ако *Property* е транзитивно и свързва A с B и B с C => то свързва A с C. Ако едно *property* е транзитивно, то и неговото инверсно е транзитивно. Ако едно *property* е транзитивно, то не може да е функционално.

- Ако *Property* свързва A с B и е симетрично, то свързва B с A.

- Ако *Property* свързва A с B и е асиметрично, то индивидуалността B не може да се свърже с индивидуалността A чрез това *Property*.

- *Property* е рефлексивно, ако свързва индивидуалността A със себе си.

- Ако *Property* е иррефлексно (*irreflexive*), то може да се опише като свързващо A с B, където A и B не са една и съща индивидуалност.

- *Property Domains* и *Ranges* свързват индивидуалности от домейна с индивидуалности от обхвата.

- *Property Restriction* е клас от индивидуалности, определен чрез връзките, в които тези индивидуалности участват. Описват анонимен (неименуван) клас.

Една от ключовите възможности на онтологиите, описани с OWL, е че могат да бъдат обработени с *Reasoner*. Една от основните услуги, които предлага *Reasoner* е да тества, дали един клас е подклас на друг или не (subsumption testing). Друга вградена възможност е да провери онтологията за съгласуваност (consistency).

5. Следваща стъпка в процеса на обучение е дефинирането на примитивни и определени класове, на базата на необходими и достатъчни условия:

- класове, които имат само необходими условия се наричат примитивни (primitive). Това означава, че ако нещо е член на даден клас, то е необходимо да изпълни дадени условия. Не можем да кажем обаче, че ако дадена индивидуалност изпълнява необходимите условия, то тя е член на този клас, ако не сме дефинирали достатъчно условие за това.
- класове, които имат най-малко едно множество от необходими и достатъчни условия се наричат определени класове (defined classes).

OWL позволява всички елементи на онтологията, както и самата онтология да бъдат документирани с мета-данни.

6. Разработва се съвместно със студентите примерна онтология, преминавайки през всички стъпки в процеса на създаването.

В резултат се постига въвеждане на онтологиите, чрез разясняване на базовите понятия и принципи в областта, базирайки се на примери.

- В процеса на обучение се предоставят възможности за допълване на изградената онтология – с нови понятия, отношения или атрибути. Механизмът на езика OWL позволява първоначално създаване на онтология с базови концепции и развитие на съществуващи онтологии.

- По време на обучението се прилага проектно-базирания подход като педагогически модел на междупредметни дейности, насочени към реално съществуваща проблематика.

- За по-доброто разбиране на онтологиите от студентите и преодоляване на трудностите, са подготвени и предоставени допълнителни обучаващи ресурси и примери в средата за електронно обучение към курсовете.

ИЗВОД: Предлага се метод за въвеждане на онтологии в курсовете на специалностите в направление “Информатика и компютърни науки” чрез реализация на три основни задачи, които се решават в процеса на обучение [148]. *Първата задача* е свързана с описание на природата на онтологията. Представя се произхода и значението на онтологията и различните видове онтологии. Студентите се запознават с основните компоненти на структурата на онтологията. *Втората задача* е свързана с представяне на схеми, които реализират онтологии. Тази част от обучението се базира на примери, сравнение между различни примерни онтологии, както и анализ на всяка от тях. За да се направи това, се въвеждат средствата на езика OWL. *Третата задача* е разработването на пример за OWL онтология, с помощта на софтуера Protégé. Освен това се обсъждат въпроси, възникнали по време на обучението и възможности за подобряване на резултатите от обучението в областта.

5.4. Виртуалното образователно пространство и науката за данните

Виртуалното образователно пространство и науката за данните са две тясно свързани области, които взаимно се допълват и съчетават. От една страна **науката за данните** улеснява достъпа до знания и ресурси, предоставяйки средства и инструменти към средите за обучение, с които допринася за тяхното развитие и повишаване на качеството на провежданото обучение. От друга страна **виртуалното образователно пространство** предоставя подходяща среда за обучение и разнообразни ресурси, които подкрепят обучението и развитието на умения в областта на науката „Data Science“. Тази среда позволява на студентите да имат достъп до различни инструменти за анализ на данни, както и платформи за изпълнение на проекти и практически упражнения в областта, прилагайки теоретични знания в реални сценарии с реални данни. Виртуалните образователни ресурси могат да бъдат представени чрез интерактивни материали като симулации, уеб базирани приложения и видео лекции, които визуализират сложни концепции от областта на науката за данните в по-достъпен и лесен за разбиране начин.

Виртуалното образователно пространство създава уникална възможност за обучаемите да развият практически умения за събиране, обработка, анализ и визуализация на данни и да подобрят своите умения в областта на анализа на данни и машинното обучение, което е от съществено значение в съвременния свят, където данните играят все по-голяма роля във вземането на решения и развитието на технологиите. Освен това виртуалното образователно пространство предоставя платформи и форуми, където обучаемите, студентите и професионалистите в областта на науката за данните могат да се свързват, споделят знания, проблеми и предизвикателства. Това създава възможности за колаборация и обмен на идеи между обучаемите.

В българските университети и училища няма традиция да се използва науката за данните, за анализ и оптимизиране на информационните системи в образователната институция както и да има работни позиции, които да изискват такава научна експертиза, докато в други държави съществуват:

- *Бизнес анализатор в образованието* - специалист, който се занимава със събирането, анализа и интерпретацията на данните, свързани с образователния процес и успеха на обучаемите. Целта им е да проследяват тенденциите, да идентифицират проблемни области и да предоставят данни и анализи, които помагат на ръководство да взема информирани решения.

- *Системен анализатор в образованието* - специалист отговорен за анализирането и оптимизирането на информационните системи в институция. Занимава се със събиране, обработката и анализа на данните от различни източници, обучаващи среди и системи.

- *Дигитален мениджър на образованието* – анализира данните от учебните системи и управлява дигиталните ресурси и комуникационния процес, с цел да оптимизира процесите на обучение и оценяване.

Тенденцията е изкуственият интелект и машинното обучение да продължават да играят все по-голяма роля във всички сфери на съвременния живот, според данните на Google Trends (<https://trends.google.com/trends/>). Алгоритмите за машинно обучение и автоматичното вземане на решения ще се развиват и ще се използват все повече за автоматизиране на процесите на анализ и прогнозиране.

ЗАКЛЮЧЕНИЕ

Науката за данните ще подпомага висшите училища при анализа на данните за успеваемостта, предпочитанията и интересите на студентите, ще даде възможност за персонализиране на обучението и развитие на по-ефективни програми. Използването на данните от различни източници, включително социалните мрежи, ще позволи да се разбират по-добре нуждите и предпочитанията на обучаемите. Ще помогне на учебните институции да оптимизират управлението на своите ресурси, като финанси, персонал и инфраструктура. Анализът на данни и прогнозите ще подпомогнат вземането на решения относно бюджетирането, планирането на капацитета и оптимизацията на процесите. Ще стимулира иновации в образователния процес, което отговаря на съвременните нужди и предизвикателства в образованието.

В процеса на изследването и анализа на резултатите се оформиха някои констатации и възникнаха въпроси:

- При наличието на огромен обем достъпна информация, както и технологии за нейното съхранение, обработване и предаване една от важните задачи продължава да бъде: ***търсене на най-ефективните начини за трансформиране на информацията в личностно достояние и подходи за лесно възприемане.*** Различните среди реализират новаторски подходи и търсят потенциални възможности за решаване на тази задача.

- Не можем да твърдим, че са установени трайни и работещи връзки между теорията и практиката, които да правят изучаването на фундаменталните теории реалистично и ефективно. Теорията и практиката все още не са удовлетворително свързани. Как да се представи учебния материал, така че да дава ясна представа ***за специфичното предназначение на теоретичните резултати и свързаните с тях потенциални практически предимства?***

- Друг въпрос е ***до колко работата в конкретна обучаваща среда може да повлияе на начина и качеството на учене.*** Предстоят изследвания за ефекта на различни практически подходи, предлагани в образователното пространство върху процеса на обучение и има ли значително повишаване на успеваемостта на студентите, когато обучението е основно в областта на фактологическите знания и какво е равнището на тяхното възпроизвеждане. Какви са възможностите на различни практически подходи върху развитието на творческото и критичното мислене на студентите?

- Колко силна е ролята на практическото обучение? ***Дали в бъдеще то ще продължи да допълва и подсилва традиционното обучение в рамките на съществуващите педагогически парадигми, или може да промени фундаментално начина, по който студентите учат и преподавателите преподават?***

Отговорите на тези въпроси и произтичащите от тях могат да бъдат търсени чрез комбиниране на педагогически подходи и нови изследвания относно ролята на обучаващите среди и техните инструменти в конкретен контекст.

Направления за бъдещи изследвания

Извличането на данни в образователни среди (Educational Data Mining) и анализът на обучението (Learning Analytics) са сравнително нови области, които имат за цел да подобрят образователния опит, да помагат на заинтересованите страни (преподаватели, студенти, администратори, изследователи), да вземат по-добри решения, използвайки

натрупаните данни. Въпреки големите очаквания и нарасналия обем на публикациите в областта на извличането на данни в образователни среди, все още стоят бариери и предизвикателства пред изследователите в областта, като **липсата на изчерпателни и лесни за използване и разбиране инструменти, които могат да бъдат интегрирани в най-популярните среди за управление на обучението.**

Съвременните обучаващи среди се базират на софтуерни платформи за управление на учебното съдържание, осигуряващи технологичен апарат, при който паралелно с публикуване на учебното съдържание може да се събира и съхранява информация и за активността на обучаемите и преподавателите. Насока за бъдещо развитие на работата е създаване на лесни за ползване софтуерни инструменти, с възможности за интегриране в средите за управление на обучението, за ранното откриване на обучаеми в риск и своевременно уведомяване на преподавателите кои обучаеми се нуждаят от допълнителна помощ и кои практики на преподаване оказват най-голямо въздействие. Освен това трябва да продължат изследванията, относно възможностите за персонализиране на обучаващите среди, така че всеки обучаем да получава необходимите ресурси спрямо текущото ниво на познания и нагласа за обучаващия процес.

Предизвикателство е разработване на адаптивни курсове, които се персонализират автоматично според профилите на обучаемите (нужди, цели, произход, държава, стил на учене и т.н.). В дисертационния труд фокусът е върху наблюдение и идентифициране на обучаеми в риск, за които има вероятност да отпаднат или да се провалят по време на обучението. Но проведените анализи на данни са върху сравнително малки групи обучаеми. По-добри резултати могат да бъдат получени при анализиране на голям брой студенти, курсове и институции. Например при MOOCs (Massive Open Online Courses) могат да бъдат използвани данни от хиляди студенти. Участниците имат различни произход, зрялост, опит, нива на образование, езикови умения, цели, нужди и стилове на учене. Но значителна бариера, [94] е свързана с **етиката при използване на личните данни, която трябва да се взема предвид на всички етапи - от събирането на данни до тълкуването на резултатите и вземането на решения.** Например данни свързани с пол, социален статус, раса, религиозни вярвания, идеология или увреждане биха могли да доведат до дискриминационно отношение. От друга страна трябва да се вземе предвид и поредицата мерки, свързани със защитата на личните данни.

Следваща стъпка на изследванията в областта е създаване и прилагане на методологии за използване на Big Data Analytics в образователни среди. В ерата на големите данни, възможностите за съхраняване, управление и обработка на данни от онлайн среди за обучение дават възможност да се изследват по-добре процесите на обучение и да се търсят ефективни начини за тяхното подобряване. **Комбинацията от Big Data и анализът на обучението (Learning Analytics) е обещаваща област на бъдещи изследвания.**

Приноси на дисертационния труд

А. Научни приноси (в класа обогатяване на съществуващите знания).

Предложени са методи и модели, като резултат от теоретични обобщения на процесите на наблюдение и анализ на дейностите на обучаемите:

1. Петстъпков метод за оценяване и прогнозиране на знанията, уменията и компетентностите на обучаемите във виртуалното образователно пространство, с възможност да се прилагат изчислителни модели и критерии за динамично оценяване, като окончателната оценка се формира от множество оценъчни компоненти. Проведените емпирични изследвания потвърждават, че предложеният метод е перспективен за разработване на система за ранно предупреждение за различни заинтересовани страни от обучаващия процес.
2. Модели чрез размита логика на йерархично многокомпонентно оценяване на различни мисловни умения от висок и нисък порядък, теоретични знания и практически умения: *Модел 1* на йерархична дървовидна организация на оценъчните компоненти и *Модел 2* на йерархична графова (недървовидна) организация на оценъчните компоненти. Представените модели могат да се използват при множество различни подходи за оценяване, при които крайната оценката е функция на множество оценъчни компоненти.
3. Метод с уеб метрики и индуктивна размита класификация за оценка на степента на използване на уеб-ресурси от обучаемите, при анализ на поведението им в обучаващата среда и уеб пространството.
4. Метод за анализ на типа на документ, на базата на класификационни алгоритми, като целевия атрибут е дали документа е свързан с областта на обучение. Анализът на данните дава възможност да се търсят трудностите, които обучаемите срещат при работа с литературни източници и да се проектират модули, които да отговарят на индивидуалните им нужди при търсене и използване на документи.
5. Трифакторен модел на обучаемия, който включва: фактори на компетентност, емоционални фактори, фактори на социалната среда. Чрез моделът на обучаемия може да се проследи промяната на знанията, уменията и компетентностите и да се прави предвиждане, според натрупаните данни от работата на среди и системи за електронно обучение с различни потребители.
6. Обобщеномрежови модели:
 - с възможност за проследяване на процесите на използване на различни инструменти в среди за електронно обучение;
 - на процеса на прилагане на *Data Mining* средства в среди за обучение;
 - на процеса на многокомпонентно оценяване в шест етапа с възможност за персонализация на начина за формиране на тестове чрез дефиниране на мета-модели, задаващи рамки за създаване на тестове и подходи за оценяването им;
 - на процесите в проектното обучение и възможностите за интеграция с електронното и уеб-базираното обучение;
 - за игровизация на *E-learning* курс, за анализиране на възможностите и проблемите при обучение с игрови ситуации.

Създадените модели предлагат частично решение на проблемите, формулиран в книгата на акад. Красимир Атанасов „*Generalized Nets and Intuitionistic Fuzziness in Data Mining*“ [34].

7. Слабо-централизиран и силно-централизиран модел за осигуряване на качеството и акредитацията във висшето образование, като се отчита йерархичната структура на организацията и позволява управление и оптимизиране на ресурсите. Базов обощеномрежов модел на информационните потоци при обработката на данни при самооценяване във висшето училище.

Б. Научно-приложни приноси (в класа приложение на научните постижения в практиката)

Предложени са:

1. Многостъпкови методи за създаване на модели чрез инструментите за машинно обучение на системата *Orange Data Mining*. Методите могат да бъдат използвани за изследване, анализ и обучение.
2. Софтуер на инструменти за анализиране и сравняване на поведението на два агента за машинно обучение: *Rule-Based System* (система базирана на правила) и *Reinforcement Learning* (подсилващо обучение).
3. Софтуерни инструменти за анализ на звуковите честоти и преобразуването им в цветове в RGB модела. Създаден е модел на автоматизирана система, която реализира преобразуване на звуковите честоти в цвят.
4. Основни модули на учебни програми на науката за данните в дисциплини на Висшите училища.
5. Методически техники за обучение на студенти с инструменти за извличане на знания чрез примери от реални проблеми и задачи за проектиране и въвеждане на онтологии в обучението на студенти.

Декларация за оригиналност на резултатите

Декларирам, че настоящата дисертация съдържа оригинални резултати, получени при проведени от мен научни изследвания. Резултатите, които са получени, описани и/или публикувани от други учени, са надлежно и подробно цитирани в библиографията.

Настоящата дисертация не е прилагана за придобиване на научна степен в друго висше училище, университет или научен институт.

Подпис:



Списък на научните публикации по темата на дисертационния труд

в издания, които са реферирани и индексирани в световноизвестни бази данни с научна информация (Web of Science и Scopus), след процедурата „Професор“ (2012 г.)

1. **Orozova D.**, Appropriate E-Test System Selection Model, Comptes rendus de l'Academie bulgare des Sciences, 2019, Vol 72, No 6, pp. 811-820, ISSN 1310-1331, DOI: 10.7546/CRABS.2019.06.14, IF=0.343 (2019), **WoS** Q4, SJR=0.218, (Scopus Q2)

<https://www.webofscience.com/wos/woscc/full-record/WOS:000477784300014>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85070988936&partnerID=MN8TOARS>

2. **Orozova D.**, K. Atanassov, Generalized Net Model of Processes Related to Big Data, Comptes rendus de l'Académie bulgare des Sciences, book No 12, vol. 71, 2018, pp. 1679 -1686 ISSN 13101331, DOI: 10.7546/CRABS.2018.12.13, IF=0.321 (2018), **WoS** Q4, Scopus SJR=0.205, (Scopus Q2) <https://www.webofscience.com/wos/woscc/full-record/WOS:000456750000013>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85063227040&partnerID=MN8TOARS>

3. **Orozova D.**, K. Atanassov, Model of Big Data Map/Reduce Processing, Comptes rendus de l'Academie bulgare des Sciences, 2019, Vol 72, No11, pp.1537-1545, ISSN 1310-1331, DOI: 10.7546/CRABS.2019.11.11, IF=0.343 (2019), **WoS** Q4, SJR=0.218, (Scopus Q2)

<https://www.webofscience.com/wos/woscc/full-record/WOS:000502809100011>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85109014572&partnerID=MN8TOARS>

4. **D. Orozova**, K. Atanassov, M. Todorova, Generalized Net Model of the Process of Personalization and Usage of an e-Learning Environment, Proceedings of the Jangjeon Mathematical Society 19 (2016), No. 4, pp. 615 – 624. ISSN (print):1598-7264, ISSN (online):2508-7916, ISBN:89-87809-15-3, SJR=0.508, (Scopus Q2)

URL <http://www.scopus.com/inward/record.url?eid=2-s2.0-84996565258&partnerID=MN8TOARS>

5. **Orozova D.**, M. Todorova, Ontology concept in courses on students, TEM Journal. Volume 7, Issue 3, Pages 693-697, ISSN 2217-8309, DOI:10.18421/TEM73-29, 2018. SJR=0.148. (Scopus Q4) <https://www.webofscience.com/wos/woscc/full-record/WOS:000442699500029>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85052218825&partnerID=MN8TOARS>

6. Popchev I., **D. Orozova**, Towards Big Data Analytics in the E-learning Space. Cybernetics and Information Technologies, Vol. 19(3), 2019, pp.16-25, ISSN: 1311-9702, DOI: 10.2478/cait-2019-0023, SJR=0.31, (Scopus Q2)

<https://www.webofscience.com/wos/woscc/full-record/WOS:000486349500002>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85073873211&partnerID=MN8TOARS>

7. **Orozova D.**, K. Kaloyanova, M. Todorova, Introducing Information Security Concepts and Standards in Higher Education, TEM Journal. Volume 8, Issue 3, Pages 1017-1024, ISSN 2217-8309, DOI: 10.18421/TEM83-46, 2019, SJR=0.167, (Scopus Q2)

<https://www.webofscience.com/wos/woscc/full-record/WOS:000483978200046>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85072752036&partnerID=MN8TOARS>

8. Popchev I., **D. Orozova**, Towards a multistep method for assessment in e-learning of emerging technologies, Cybernetics and Information Technologies, Volume 20, No 3, Sofia, 2020, pp.116-129, ISSN: 1311-9702, DOI: 10.2478/cait-2020-0032, SJR=0.272, (Scopus Q2)

<https://www.webofscience.com/wos/woscc/full-record/WOS:000572532500009>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85093080750&partnerID=MN8TOARS>

9. Hadzhikolev, E., S. Hazdhikoleva, K. Yotov, **D. Orozova**, Models for Multicomponent Fuzzy Evaluation, with a Focus on the Assessment of Higher-Order Thinking Skills, TEM Journal, Vol.9, No.4, pp.22-28, 2020, ISSN: 2217-8309, DOI: 10.18421/TEM94-43, SJR=0.199, (Scopus Q4)

<https://www.webofscience.com/wos/woscc/full-record/WOS:000596297300043>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85092629596&partnerID=MN8TOARS>

10. Todorov, J., V. Valkanov, S. Stoyanov, B. Daskalov, I. Popchev, **D. Orozova**. Chapter 6: Personal Assistants in a Virtual Education Space, Sgurev V., Jotsov V., Kacprzyk J. (eds) Practical Issues of Intelligent Innovations. Studies in Systems, Decision and Control, 140, Springer International Publishing AG part of Springer Nature, 2018, Library of Congress Control Number 2011893667, ISSN 2198-4182, DOI: 10.1007/978-3-319-78437-3_6, SJR=0.131 (Scopus Q4)

<https://www.webofscience.com/wos/woscc/full-record/WOS:000466552900007>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85049662711&partnerID=MN8TOARS>

11. Todorov J., E. Doychev, **D. Orozova**, A. Stoyanova-Doycheva, IoT Multiagent Assistant in Virtual Educational Space, 2018, AIP Conference Proceedings Volume 2048, pp. 020031; ISSN: 0094243X, ISBN: 978-073541774-8, doi: 10.1063/1.5082049, <https://doi.org/10.1063/1.5082049>, SJR=0.182 (за 2018).

<https://www.webofscience.com/wos/woscc/full-record/WOS:000468108800031>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85058786246&partnerID=MN8TOARS>

12. Stoyanova-Doycheva A., T. Glushkova, S. Stoyanov, **D. Orozova**, Lifelong Learning Supported by an Intelligent Tourist Guide, 2018, AIP Conference Proceedings, Volume 2048, pp. 020038; ISSN: 0094243X, ISBN: 978-073541774-8, SJR=0.182, DOI: 10.1063/1.5082056

<https://www.webofscience.com/wos/woscc/full-record/WOS:000468108800038>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85058819893&partnerID=MN8TOARS>

13. Hadzhikolev E., S. Hadzhikoleva, **D. Orozova**, K. Yotov, A Comprehensive Approach to Assessing Higher and Lower Order Thinking Skills, Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, Lecture Notes in Networks and Systems, Volume 338, pp. 164–177, Lecture Notes in Networks and Systems, Volume 338, pp. 164–177, ISSN 2367-3370, 2022, DOI: 10.1007/978-3-030-95929-6_13, SJR=0.151 (Scopus Q4)
<https://www.webofscience.com/wos/woscc/full-record/WOS:000775291100013>
<http://www.scopus.com/inward/record.url?eid=2-s2.0-85126177886&partnerID=MN8TOARS>
14. Hadzhikoleva S., **D. Orozova**, N. Andonov, E. Hadzhikolev, Generalized Net Model of a System for Quality Assurance in Higher Education, 45th International Conference Applications of Mathematics in Engineering and Economics (AMEE'19), Sozopol, June, 2019, pp. 040005-1 - 040005-8, ISBN: 978-0-7354-1919-3. AIP Conference Proceedings, 2019, DOI: 10.1063/1.5133515, Part of ISSN: 15517616 0094243X, SJR=0.19 (за 2019).
<https://www.webofscience.com/wos/woscc/full-record/WOS:000521744400033>
<http://www.scopus.com/inward/record.url?eid=2-s2.0-85075796767&partnerID=MN8TOARS>
15. Orozov B., **D. Orozova**, An AI application in E-Learning, CEUR Workshop Proceedings 3061, pp. 76-80, September 27–28, 2021, Plovdiv, Bulgaria, ISSN 1613-0073, SJR=0.228.
<http://www.scopus.com/inward/record.url?eid=2-s2.0-85122281151&partnerID=MN8TOARS>
16. Ivan Popchev, **Daniela Orozova**, Algorithms for Machine Learning with Orange System, International Journal of Online and Biomedical Engineering (iJOE), Vol. 19, No. 04, 2023, pp. 109 - 123, ISSN: 2626-8493, SJR=0. SJR=0.304 (за 2022) (Scopus Q2), DOI:
<https://doi.org/10.3991/ijoe.v19i04.36897>.
<https://www.webofscience.com/wos/woscc/full-record/WOS:000967834800007>
<https://www.scopus.com/record/display.uri?eid=2-s2.0-85152704594&origin=resultslist&sort=plf-f>
17. **D. Orozova**, N. Hristova, An application of analytical data research in e-learning system, ICTTE 2020, International Conference on Technics, Technologies and Education, Journal IOP Conference Series: Materials Science and Engineering Volume 1031 (2021) 012058, IOP Publishing, Online ISSN: 1757-899X, DOI: 10.1088/1757-899X/1031/1/012058.
<http://www.scopus.com/inward/record.url?eid=2-s2.0-85101662489&partnerID=MN8TOARS>
18. Popchev I., **D. Orozova**, DataScience: Experience and Trends, ICTTE 2020, International Conference on Technics, Technologies and Education, Journal IOP Conference Series: Materials Science and Engineering, Volume 1031 (2021) 012057, IOP Publishing, Online ISSN: 1757-899X, Print ISSN: 1757-8981, doi:10.1088/1757-899X/1031/1/012057.
<http://www.scopus.com/inward/record.url?eid=2-s2.0-85101718405&partnerID=MN8TOARS>

19. M. Todorova, **D. Orozova**, Training Difficulties in Deductive Methods of Verification and Synthesis of Program, International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 9, No. 7, 2018, pp. 18-22, ISSN: 2158107X ISSN (online):2156-5570, doi:10.14569/IJACSA.2018.090703.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000441330600003>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85054031652&partnerID=MN8TOARS>

20. **D. Orozova**, SQL User Queries Execution Model, Advances in Intelligent Systems and Computing book series (AISC, volume 1081), Uncertainty and Imprecision in Decision Making and Decision Support: New Challenges, Solutions and Perspectives, Volume 1081 AISC, 2021, Pages 184-189, ISSN 2194-5357, DOI: 10.1007/978-3-030-47024-1_20

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85087754542&partnerID=MN8TOARS>

21. Orozov B., **D. Orozova**, Rule Based System Against Reinforcement Learning, CompSysTech'21: ACM International Conference Proceeding Series on Computer Systems and Technologies '21, June 2021, Pages 67–70, ISBN: 978-145038982-2, DOI: 10.1145/3472410.3472437.

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85117607164&partnerID=MN8TOARS>

22. **Daniela Orozova**, Milena Georgieva, A model of the process of Big Data with generalized net, 2016, IEEE 8th International Conference on Intelligent Systems, IS 2016 – Proceedings, 7 2016, Article number 7737487, pp. 599-603, ISBN: 978-150901353-1

<https://www.scopus.com/record/display.uri?eid=2-s2.0-85006035368&origin=resultslist&sort=plf-f>

23. Hadzhikolev E., S. Hadzhikoleva, **D. Orozova**, Digital Model of a Document in a University Document Repository, IEEE XX-th International Symposium on Electrical Apparatus and Technologies SIELA 2018, 3-6 June 2018, Bulgaria, ISBN 978-1-5386-3418-9, DOI: 10.1109/SIELA.2018.8447089.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000454837100037>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85053840788&partnerID=MN8TOARS>

24. **D. Orozova**, A. Ivanov, Generalized Net Model of Virtual Collaboration Space, IEEE XX-th International Symposium on Electrical Apparatus and Technologies SIELA 2018, 3-6 June 2018, Bourgas, Bulgaria, ISBN 978-1-5386-3418-9, DOI: 10.1109/SIELA.2018.8447090.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000454837100038>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85053825957&partnerID=MN8TOARS>

25. **Orozova D.**, E. Sotirova, Modeling of a Learning Management System, IEEE XVIII-th International Symposium on Electrical Apparatus and Technologies SIELA 2014, 2014, Bulgaria, pp. 95-98, ISBN: 978-147995817-7, DOI: 10.1109/SIELA.2014.6871879.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000345744800038>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-84906751001&partnerID=MN8TOARS>

26. **Daniela Orozova**, Milena Georgieva, A model of the process of Big Data with Generalized Net, Electrical Apparatus and Technologies (SIELA), 2016, IEEE 19th International Symposium, Bourgas, 238-241, ISSN: 978-1-4673-9521-2, ISBN 978-619-160-648-1, DOI: 10.1109/SIELA.2016.7543030.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000382936800059>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-84985945018&partnerID=MN8TOARS>

27. I. Popchev, **D. Orozova**, S. Stoyanov, IoT and Big Data Analytics in E-Learning, 2019 Big Data, Knowledge and Control Systems Engineering (BdKCSE), IEEE 2019, pp. 1-5, ISBN 978-172816481-6, DOI: 10.1109/BdKCSE48644.2019.9010666.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000563467600027>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85082305981&partnerID=MN8TOARS>

28. **Orozova D.**, I. Popchev, Cyber-Physical-Social Systems for Big Data, IEEE XXI International Symposium on Electrical Apparatus and Technologies SIELA 2020, 3-6 June 2020, Bourgas, Bulgaria, pp. 334 – 337, ISBN 978-1-7281-4345-3, DOI: 10.1109/SIELA49118.2020.9167161.

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85091338714&partnerID=MN8TOARS>

29. **Orozova D.**, N. Hristova, Generalized Net Model for dynamic decision making and prognoses, IEEE XXI International Symposium on Electrical Apparatus and Technologies SIELA 2020, Bulgaria, 2020, pp.330 – 333, ISBN 978-1-7281-4345-3, DOI: 10.1109/SIELA49118.2020.9167077.

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85091339727&partnerID=MN8TOARS>

30. Popchev I., **D. Orozova**, Text Mining in the Domain of Plant Genetic Resources, Proceedings of 2020 IEEE 10th International Conference on Intelligent Systems, 2020, pp. 596-600, ISBN 978-172815456-5, DOI: 10.1109/IS48319.2020.9200174.

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85092747530&partnerID=MN8TOARS>

31. Hadzhikoleva S., **D. Orozova**, E. Hadzhikolev, N. Andonov, Model of a Centralized System for Quality Assurance in Higher Education, Proceedings of 2020 IEEE 10th International Conference on Intelligent Systems 2020, pp. 87-92, ISBN 978-172815456-5, DOI: 10.1109/IS48319.2020.9199951.

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85092696472&partnerID=MN8TOARS>

32. Georgieva P., E. Nikolova, **D. Orozova**, Data Cleaning Techniques in Detecting Tendencies in Software Engineering, MIPRO 2020, IEEE 43rd International Convention, 2020, Croatia, Opatija, pp.1272- 1277, ISSN: 2623-8764, DOI: 10.23919/MIPRO48935.2020.9245416.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000790326400188>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85097229443&partnerID=MN8TOARS>

33. **D. Orozova**, S. Hadzhikoleva, E. Hadzhikolev, A Course Gamification Model for the Development of Higher Order Thinking Skills, MIPRO 2021, IEEE 44th International Convention, September 27, 2021 – October 1, 202, Opatija, Croatia, pp. 1772-1777, ISSN 1847-3946, DOI: 10.23919/MIPRO52101.2021.9597151.

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85123052448&partnerID=MN8TOARS>

34. I. Popchev, **D. Orozova**, Data Analytics Approach in Virtual Educational Space, 2021 IEEE Big Data, Knowledge and Control Systems Engineering (BdKCSE), 2021, pp. 1-5, doi: 10.1109/BdKCSE53180.2021.9627247, ISBN 978-166541042-7.

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85123994089&partnerID=MN8TOARS>

35. **D. Orozova**, M. Todorova, Applying Reactive Blocks in Courses of Internet of Things, Proceedings of the 10th Annual International Conference of Education, Research and Innovation, Seville, 16-18 November, 2017, стр. 6960-6967, ISSN (print):2340-1095.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000429975307011>

36. **D. Orozova**, M. Todorova, How to Follow Modern Trends in Courses in "Databases" - Introduction of Data Mining Techniques by Example, Proceedings of the 11th Annual International Technology, Education and Development Conference (INTED), Valencia, 2017, стр.8186-8194, ISSN:2340-1079.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000427401303026>

37. Georgieva P., E. Nikolova, **D. Orozova**, Tendencies in Software Engineering Education, ICERI2019, 12th International Conference of Education, Research and Innovation, Seville (Spain) 11-13 November, 2019, pp. 3601- 3609, ISBN: 978-84-09-14755-7.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000530109203108>

38. Georgieva P., E. Nikolova, **D. Orozova**, V. Jecheva, Certification In Teaching Information Technologies For Engineering Graduates, ICERI2019, 12th International Conference of Education, Research and Innovation, Seville (Spain) 11-13 November, 2019, pp. 4564-4573.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000530109204100>

39. D. Tuparova, G. Tuparov, **D. Orozova**, Educational Computer Games and Gamification in the Higher Education – Students’ Points of View, 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), 2020, Croatia, Opatija, pp.1879-1884 Electronic ISSN: 2623-8764, DOI: 10.23919/MIPRO48935.2020.9245251.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000790326400284>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85097241244&partnerID=MN8TOARS>

40. Magdalena Todorova, **Daniela Orozova**, Generalized Net Model of Sequential Programs, In proc of the 20th International Symposium on Electrical Apparatus and Technologies (SIELA 2018), 3-6 June 2018, Bourgas, Bulgaria, DOI: 10.1109/SIELA.2018.8447068, Electronic ISBN: 978-1-5386-3419-6, ISBN: 978-1-5386-3420-2.

<https://www.webofscience.com/wos/woscc/full-record/WOS:000454837100016>

<http://www.scopus.com/inward/record.url?eid=2-s2.0-85053816562&partnerID=MN8TOARS>

41. Zlatev Z., J. Ilieva, **D. Orozova**, G. Shivacheva, N. Angelova, Design and Research of a Sound-to-RGB smart acoustic device, Multimodal Technologies and Interaction, 2023, 7(8), 79; ISSN: 2414-4088, WoS, Scopus Q2 SJR=0.504 (за 2022)

<https://doi.org/10.3390/mti7080079>

<https://www.mdpi.com/journal/mti/stats>

Издаден учебник:

42. Иван Попчев, **Даниела Орозова**, Представяне на знания в системи с изкуствен интелект, издателство Божич, Бургас, 2018, ISBN 978-619-7181-59-3.

БИБЛИОГРАФИЯ

1. Барендрегт Х., Лямбда-исчисление. Его синтаксис и семантика. М., Мир, 1985.
2. БДС ISO/IEC 27001:2006. Информационни технологии. Методи за сигурност. Системи за управление на сигурността на информацията. Изисквания (ISO/IEC 27001:2005)
3. Вълканова, В., М. Сандалски, И. Попчев, С. Стоянов. Образователен портал за подготовка на ученици за зрелостни изпити. - Взаимодействието теория - практика: Ключови проблеми и решения. Международна научна конференция, том трети, БСУ, Бургас, 2011, 287-294. ISBN: 978-954-9370-80-5.
4. Мендельсон Э., Введение в математическую логику. Москва, Наука, 1971.
5. НАОА: Националната агенция за оценяване и акредитация <https://www.neaa.government.bg/>
6. Орозова, Д., С. Стоянов и И. Попчев, "Виртуално образователно пространство" в Научна конференция с международно участие "Знанието - източник на иновация", БСУ, Бургас, 2013, pp. 153-159, ISBN 978-954-9370-99-7.
7. Орозова Д., Предизвикателства на Big Data и Big Data Analytics, Годушник на Бургаски свободен университет, том XXXVIII, 2018, стр. 47-52.
8. Орозова Д., Интегриране на data mining средствата в среди за електронно обучение, VII Национална конференция „Образованието и изследванията в информационното общество”, май, 2014, Пловдив, България, стр. 154-160. ISSN 1314-0752 (print), ISBN 978-954-8986-39-7.
9. Орозова Д., Електронни курсове чрез SCORM стандарт, Новации в прилагането на електронни форми в дистанционното обучение, Сборник доклади от конференция на БСУ, 28-29 ноември, 2014, стр. 71-80. ISBN 978-954-8468-96-1
10. Попчев И., Д. Орозова, Представяне на знания в системи с изкуствен интелект, издателство Божич, Бургас, 2018, ISBN 978-619-7181-59-3.
11. Попчев И., Д. Орозова, Съвременност: Emerging Technologies и Рискове, Международна научна конференция "Съвременни управленски практики XI - Интелигентната специализация в десетилетието на свързаността и автоматизацията“, Бургаски свободен университет, 4-5 юни 2021 г. стр. 402-410, ISSN: 1313-8758.
12. Попчев, И., И. Радева, Четвъртата индустриална революция и новите рискове – Техносфера, бр. 2(44), 2019, 69 - 73. Издателство на БАН „Проф. М. Дринов. ISSN 1313-3861
13. Попчев И., И. Радева, Новата парадигма и рискът в релацията „човек – цифрова среда“ - Списание на БАН 5/2019, pp. 72-77, ISSN 0007-3989
14. Притам, В. В. Защитни стени и сигурност в Интернет. София: Duo Design, 2005. 352 с. ISBN 954-8396-22-X.
15. Рос, Алек. Индустриите на бъдещето. София, НСМ Медиа, 2017, стр.317, ISBN 978-954-9913-61-3.
16. Стоянов С., В. Табакова-Комсалова, Л. Дуковска. Изкуствен интелект в 24..., Книга първа, Издателство на БАН „Проф. Марин Дринов“, 2023, ISBN: 978-619-245-305-3.
17. Стоянов С., Д. Орозова, И. Попчев, Е. Дойчев Виртуално пространство за продължаващо обучение, БСУ Научна конференция с международно участие “Хоризонти в развитието на човешките ресурси и знанието” том 2, 2015, стр. 419-425
18. Стоянов С., Д. Орозова, И. Попчев. Виртуално образователно пространство като интернет на нещата екосистема, Международна научна конференция „Синя икономика и синьо развитие“, БСУ 2018, стр. 494-503.

19. Стоянов, С., М. Сандалски, И. Попчев, Г. Чолаков, Е. Дойчев. Персонализирана и проактивна доставка на електронни услуги в образователния портал на DeLC. - Образованието в информационното общество. IV национална конференция, Пловдив, 26-27 май 2011. Пловдив, 2011, 119-128.
20. Тодорова М., Подходи, програмни среди и езици за проверка на коректността на програми и прилагането им при подготовката на софтуерни специалисти, София, 2013.
21. Фейс Р., Модална логика. Москва, Наука, 1974.
22. Христова Н., И. Попчев, Д. Орозова, Приложение на анализа на данни за моделиране на обучението, Годишник на Бургаския свободен университет, БСУ, 2019, ISSN 1311-221 X.
23. Най-добри 12 инструмента и софтуер за уеб анализ за стартиращи фирми <https://squeezegrowth.com/bg/best-web-analytics-tools-for-startups/>
24. Abt, C. C. (1987). *Serious Games*. University Press of America. Изтеглено на 11. 02. 2023 г. от <https://books.google.com/books?id=axUs9HA-hF8C>
25. Ahituv N., What Should be Taught in an Academic Program of Data Sciences, Digital Presentation and Preservation of Cultural and Scientific Heritage. Conference Proceedings. Vol. 9, Bulgaria: Institute of Mathematics and Informatics – BAS, 2019. ISSN: 1314-4006.
26. Ahlers, R., Garris, R., & Driskell, J. (2014). *Games, Motivation, and Learning: A Research and Practice Model*. Computer Science Applications. Florida Maxima Corporation.
27. Angelova N., M. Todorova, D. Oroszova, K. Atanassov, Software Protection via Generalized Nets, *Advanced Studies in Contemporary Mathematics*, 29 (2019), No. 2, pp. 225 – 236, ISSN 1229-3067
28. Anderson L. W., D. R. Krathwohl (Eds). *A taxonomy for learning, teaching and assessing: A revision of Bloom's Taxonomy of educational objectives: Complete edition*, New York, Longman, 2001.
29. Andrea De Mauro, Marco Greco and Michele Grimaldi. "What is Big Data? A Consensual Definition and a Review of Key Research Topics". In "AIP Proceedings"2014, "4th International Conference on Integrated Information".
30. Antonova, A., Stefanova, E., Nikolova, N., Mihnev, P., Stamenkova, R., Zafirova-Malcheva, T. (2022) Evaluating the Outcomes of Competence-Based Learning Course: The Field Experience for Training 21st Century Skills in Sofia University, *Lecture Notes in Networks and Systems*, vol. 469, 502-511, https://doi.org/10.1007/978-3-031-04819-7_48.
31. Atanassov K., *On Generalized Nets Theory*. "Prof. M. Drinov" Academic Publishing House, Sofia, 2007.
32. Atanassov K., *Generalized nets* (World Scientific, Singapore), 1991.
33. Atanassov, K., *Intuitionistic Fuzzy Sets-Theory and Applications*, Springer-Physica Verlag, Berlin, 1999.
34. Atanassov, K., *Generalized Nets and Intuitionistic Fuzziness in Data Mining*, "Prof. M. Drinov" Publishing House of Bulgarian Academy of Sciences, Sofia, 2020, ISBN 978-619-245-022-9.
35. Atanassov, K., Bureva, V., Index Matrix Representation of Big Data Structures, 2022, 75 (5), *Comptes rendus de l'Academie bulgare des Sciences*, pp.719-725, ISSN 13101331, DOI 10.7546/CRABS.2022.05.12
36. Baker, R. S. J. D., & Inventado, P. S. (2014). Educational Data Mining and Learning Analytics. In J. A. Larusson, & B. White (Eds.), *Learning Analytics: from Research to Practice* (pp. 61-75). New York, NY: Springer.

37. Barr A., Feigenbaum E., The Handbook of Artificial Intelligence, vol.2, 1984.
38. Bavelier, D., C. S. Green, D. H. Han, P. F. Renshaw, M. M. Merzenich, and D. A. Gentile. Brains on Video Games, Nature Reviews. – Neuroscience, 12, December 2011, 763-768.
39. Berry M., Linoff, G. (1997). Data Mining Techniques: For Marketing, Sales, and Customer Support, John Wiley & Sons, Inc. New York, NY, USA, 1997, ISBN:0471179809.
40. Berkeley Bootcamp: <https://sites.google.com/view/deep-rl-bootcamp/lectures>
41. Bianchi, A., I. Oakley, D. Kwon. Open Sesame: Design Guidelines for Invisible Passwords. Computer, IEEE Computer Society Publ., April 2012, 58-65.
42. Bienkowski, M., Feng, M., & Means, B. (2012). Enhancing Teaching and Learning Through Educational Data Mining and Learning Analytics: An Issue Brief. Retrieved from <http://tech.ed.gov/wp-content/uploads/2014/03/edm-la-brief.pdf>
43. Big Data Definition, Teradata, <https://www.teradata.com/Glossary/What-is-Big-Data>, последен достъп: 01.02.2023
44. Bloom B., M. Engelhart, E. Furst, et. al., “Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain”. New York: David McKay Company, 1956.
45. Borissova D., D. Keremedchiev, Group Decision Making in Evaluation and Ranking of Students by Extended Simple Multi-Attribute Rating Technique, Cybernetics and Information Technologies, Volume 19, No 3, Sofia, 2019, pp.45-56.
46. Bradley, J., Atkins, E. Coupled Cyber-Physical System Modeling and Coregulation of a CubeSat, IEEE Transactions on Robotics, 2015, Vol. 31, No. 2, 443-456.
47. Brookhart C., “How to Assess Higher-Order Thinking Skills in Your Classroom”, publ. ASCD, 2010, ISBN: 978-1-4166-1048-9.
48. Buckland, B., F. Schreier & T. H. Winkler. Democratic Governance Challenges of Cyber Security. DCAF Horizon 2015 Working Paper No. 1, Geneva, Geneva Centre for the Democratic Control of Armed Forces.
49. Egger, N. etc., Themes in Science & Technology Education, 2014, Vol. 7 Issue 1, 3-17, 15p.
50. [ESG] Standards and Guidelines for Quality Assurance in the European Higher Education Area (ESG). (2015). Brussels, Belgium. ISBN: 978-9-08-168672-3. Available at: https://enqa.eu/wp-content/uploads/2015/11/ESG_2015.pdf.
51. BDI4JADE <http://www.inf.ufrgs.br/prosoft/bdi4jade/>
52. Balzarotti, D. (Editor). Second Report on Threats on the Future Internet and Research Roadmap. – SysSec Consortia, August 2012, Available at <http://www.syssec-project.eu/media/page-media/3/syssec-d4.2-future-threats-roadmap-2012.pdf>.
53. Blanco-Fernández, Yolanda, Martín López-Nores, Alberto Gil-Solla, Manuel Ramos-Cabrer и José J. Pazos-Arias, "Exploring synergies between content-based filtering and Spreading Activation techniques in knowledge-based recommender systems", Information Sciences, vol. 181, no. 21, pp. 4823-4846, November 2011.
54. Bower, J., C. Christensen. Disruptive Technologies: Catching the Wave. Harvard Business Review, January-February 1995, pp. 43-53.
55. Brown, B. The Psychology of Gamification: Why It Works. (Heyswift Pte. Ltd) Изтеглено на 15.02. 2023 г. от <https://www.bitcatcha.com/blog/gamify-website-increase-engagement/>
56. Brynjolfsson E., A. McAfee, The Second Machine Age: Work, Progress, and Prosperity in a Time of Brilliant Technologies, W.W. Norton & Company, 2014.
57. Bradfield J., C. Stirling, Modal Mu-Calculi, 2005, <http://homepages.inf.ed.ac.uk/jcb/Research/MLH-bradstir.pdf>

58. Cao L. Data science: a comprehensive overview. *ACM Comput Surv (CSUR)*. 2017;50 (3) pp.1–42.
59. Chappell, D. Introducing SCA (pp. 3-9). Chappell & Associates, (2007). Retrieved March 14, 2015 from David Chappell & Associates: http://www.davidchappell.com/writing/Introducing_SCA.pdf
60. Chapman A., “Principles and Methods of Data Cleaning - Primary Species and Species-Occurrence Data.”, Report for the Global Biodiversity Information Facility, Copenhagen, 2005.
61. Chatti, M, A, Dychhoff, A, L, Schroeder, U., and Thus, H. A reference model for learning analytics, *International journal of Technology Enhanced learning*, 2012.
62. Chemchem, A., Drias, H. From data mining to knowledge mining: Application to intelligent agents. *International Joint Conference on Artificial Intelligence: Expert Systems with Applications*, (2014).
63. Chounta, IA. и N. Avouris, "Towards the real-time evaluation of collaborative activities: Integration of an automatic rater of collaboration quality in the classroom from the teacher’s perspective", *Education and Information Technologies*, vol. 21, no. 4, pp. 815–835, July 2016.
64. Chourabi. H., T. Nam, S. Walker, J. R. Gil-Garcia, S. Mellouli, K. Nahon, T. Pard. and H. J. Scholl. Understanding Smart Cities: An Integrative Framework. *System Science (HICSS)*. 45th Hawaii International Conference on System Sciences. 2012, 2289-2297.
65. Clifton, B., 2008: *Advanced Web Metrics with Google Analytics*, Wiley, New York, USA.
66. Codd, E. F., *The Relational Model for Database Management (Version 2 ed.)*, Addison Wesley Publishing Company, ISBN 0-201-14192-2, 1990.
67. Common Cartridge, <http://www.imslobal.org/commoncartridge.html>
68. Czibula G., A.-M. Guran, I. G. Czibula, and G. S. Cojocar, “IPA - An intelligent personal assistant agent for task performance support,” in *IEEE 5th International Conference on Intelligent Computer Communication and Processing. ICCP 2009*, aug. 2009, pp. 31–34.
69. Davenport, T. (2017). *The Analytics Supply Chain*. Retrieved from The Analytics Supply Chain: <https://www.linkedin.com/pulse/analytics-supply-chain-tomdavenport/>
70. Ding C., J. C. Patra, and F. C. Peng, “Personalized web search with self-organizing map,” in *IEEE International Conference on e-Technology, e-Commerce and e-Service. EEE'05*, mar. 2005, pp. 144–147.
71. Dean, J., Ghemawat, S. MapReduce: Simplified data processing on large clusters. In *Proceedings of the Sixth Conference on Symposium on Operating Systems Design & Implementation (San Francisco, Dec. 6-8, 2004)*. 137--150.
72. Dey K., Understanding and Using Context, *Personal and Ubiquitous Computing Journal*, Vol.5, No.1, pp.4-7. 2001.
73. Dey A.K., G.D. Abowd. Towards a better understanding of context and context-awareness, In: *Proceedings of the Workshop on the What, Who, Where, When and How of Context-Awareness*, New York, ACM Press, 2000.
74. Dietrich D., B. Heller, B. Yang, *Data Science & Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data*, Published by John Wiley & Sons, Inc., 2015, EMC Education Services. ISBN: 978-1-118-87613-8
75. Di Pasquale, P. CHROMOSCALE Unique Language for Sounds Colors and Numbers. *Chromoscale and Chromotones, Sound & Color and Base 7* 2013. Available online: https://www.academia.edu/2401099/CHROMOSCALE_Unique_Language_for_Sounds_Colo

[rs and Numbers](#) (accessed on 3 June 2023).

76. Ekman P., W.V. Friesen, Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 1971, 17(2), 124–129, <https://doi.org/10.1037/h0030377>.
77. Escobar-Jeria, V. H., Martín-Bautista, M. J., Sánchez, D., Vila, M., 2007: Web Usage Mining Via Fuzzy Logic Techniques. In: Melin, P., Castillo, O., Aguilar, I. J., Kacprzyk, J., Pedrycz, W. (Eds.), 2007: *Lecture Notes In Artificial Intelligence*, Vol. 4529, Springer, New York, USA, pp. 243-252.
78. Estivill – Castro, V. & Yang, J. A. (2000). Fast and robust general purpose clustering algorithm. *Pacific Rim International Conference on Artificial Intelligence* (pp. 208-218).
79. Etheridge, D., *Excel Programming: Your Visual Blueprint for Creating Interactive Spreadsheets*, John Willey & Sons, Inc. 2010
80. Ethics Guidelines for Trustworthy AI [Online]. <https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai>
81. Floridi, Luciano, *The Fourth Revolution: How the Infosphere is Reshaping Human Reality*, 1st ed. Oxford, United Kingdom: Oxford University Press, 2014.
82. Fokkink W., *Introduction to Process Algebra*, 2nd edition, 2007, <http://www.few.vu.nl/~wanf/BOOKS/procalg.pdf>
83. Fraley C., Raftery A. E. (1998). How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. Technical Report No. 329. Department of Statistics University of Washington.
84. Garber, L. The Challenges of Securing the Virtualized Environment. – *Computer*, IEEE Computer Society Publ., January 2012, pp.17-20.
85. Gaff, B., R. Loren, E. Spinney. Intellectual Property, Part II – *Computer*, IEEE Computer Society Publ., February 2012b, 9-11.
86. Ghannam, J. *Social Media in the Arab World: Leading up to the Uprisings of 2011*, Washington, D.C.: Center for International Media Assistance. Retrieved from http://cima.ned.org/sites/default/files/CIMA-Arab_Social_Media-Report%20-%2010-25-11
87. Georgieva P., E. Nikolova, D. Orozova, Data Cleaning Techniques in Detecting Tendencies in Software Engineering, MIPRO 2020, 43rd International Convention, 2020, Croatia, Opatija, pp.1272- 1277, Electronic ISSN: 2623-8764
88. Georgieva P., E. Nikolova, D. Orozova, Tendencies in Software Engineering Education, ICERI2019, 12th International Conference of Education, Research and Innovation, Seville (Spain) 11-13 November, 2019, pp. 3601- 3609, ISBN: 978-84-09-14755-7
89. Georgieva P., E. Nikolova, D. Orozova, V. Jecheva, Certification In Teaching Information Technologies For Engineering Graduates, ICERI2019, 12th International Conference of Education, Research and Innovation, Seville (Spain) 11-13 November, 2019, pp. 4564-4573
90. *Global Risks Report 2012*. Seventh Edition, World Economic Forum. Available at: http://www3.weforum.org/docs/WEF_GlobalRisks_Report_2012.pdf.
91. Gomez-Perez A., M. Fernando-Lopez, O. Corcho. “Ontological Engineering with examples from the areas of Knowledge management,” *e-Commerce and the Semantic Web 2nd. Ed.* London, Springer-Verlag, 2004.
92. Gonçalves, Sérgio, Davide Carneiro, Javier Alfonso, Florentino Fdez-Riverola и Paulo Novais, "Analysis of student's context in e-Learning" в *International Symposium on Computers in Education (SIIE)*, DOI: 10.1109/SIIE.2014.7017726, Logrono, Spain, 2014.
93. Gramatova K., S. Stoyanov, E. Doychev, V. Valkanov, Integration of eTesting in an IoT eLearning ecosystem: Virtual eLearning Space, *BCI 2015*, 14:1-14:8

94. Greller, W., & Drachsler, H. (2012). Translating Learning into Numbers: A Generic Framework for Learning Analytics, *Educational Technology & Society*, 15(3), 42-57.
95. Gruber T. R., "Translation Approach to Portable Ontologies" Knowledge Acquisition, Journal, 1993. V.5. Pp.199-220.
96. Guo, B., Yu, Z., Zhou, X. A Data-Centric Framework for Cyber-Physical-Social Systems, IT Pro, 2015, 4-7.
97. Guttag J., Algebraic Specification of Abstract Data Types, http://wwwsst.informatik.tu-cottbus.de/~db/doc/People/Broy/Software-Pioneers/Guttag_new.pdf
98. Hadzhikolev, E., S. Hadzhikoleva, K. Yotov, D. Orozova, Models for Multicomponent Fuzzy Evaluation, with a Focus on the Assessment of Higher-Order Thinking Skills, TEM Journal, Vol.9, No.4, pp.22-28, 2020, ISSN: 2217-8309.
99. Hadzhikolev E., S. Hadzhikoleva, D. Orozova, Digital Model of a Document in a University Document Repository, XX-th International Symposium on Electrical Apparatus and Technologies SIELA 2018, 3-6 June 2018, Bourgas, Bulgaria, ISBN 978-1-5386-3418-9
100. Hadzhikoleva S., D. Orozova, N. Andonov, E. Hadzhikolev, Generalized Net Model of a System for Quality Assurance in Higher Education, 45th International Conference Applications of Mathematics in Engineering and Economics (AMEE'19), Sozopol, June, 2019, pp. 040005-1 -040005-8, ISBN: 978-0-7354-1919-3. AIP Conference Proceedings.
101. Hadzhikoleva S., D. Orozova, E. Hadzhikolev, N. Andonov, Conceptual Model of a Centralized System for Quality Assurance in Higher Education, Proceedings of 2020 IEEE 10th International Conference on Intelligent Systems 2020, pp. 87-92, ISBN 978-172815456-5, DOI 10.1109/IS48319.2020.9200174
102. Hadzhikolev E., S. Hadzhikoleva, D. Orozova, K. Yotov, A Comprehensive Approach to Assessing Higher and Lower Order Thinking Skills, Uncertainty and Imprecision in Decision Making and Decision Support: New Advances, Challenges, and Perspectives, Lecture Notes in Networks and Systems, Volume 338, pp. 164–177, Lecture Notes in Networks and Systems, Volume 338, pp. 164–177, ISSN 2367-3370, 2022
103. Hayashi, C. (1998). What is Data Science? Fundamental Concepts and a Heuristic Example. In: Hayashi, C., Yajima, K., Bock, HH., Ohsumi, N., Tanaka, Y., Baba, Y. (eds) Data Science, Classification, and Related Methods. Studies in Classification, Data Analysis, and Knowledge Organization. Springer, Tokyo. https://doi.org/10.1007/978-4-431-65950-1_3
104. Halfaker, A, J. Riedl. Bots and Cyborgs: Wikipedia's Immune System. – Computer, IEEE Computer Society Publ, March 2012, pp.79-82.
105. Han J., Kamber M., Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, 2001.
106. Hassan M.A., "Relational and NoSQL Databases: The Appropriate Database Model Choice," 2021 22nd International Arab Conference on Information Technology (ACIT), Muscat, Oman, 2021, pp. 1-6, doi: 10.1109/ACIT53391.2021.9677042.
107. He, J. и W.W. Chu, "A Social Network-Based Recommender System (SNRS)" in Data Mining for Social Network Data. Annals of Information Systems, N. Memon et al., Eds.: Springer, Boston, MA, ch. Vol. 12, pp. 47-74.
108. Heizenberg, J., Herschel, G., Lo, T., Vashisth, S., & Jain, A. (2022). Gartner Magic Quadrant for Data and Analytics Service Providers. Gartner, Inc. <https://www.gartner.com/en/documents/4011116>
109. Himanshu Agarwal ,G. N. Pandey, "Impact of E-Learning in Education", International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064.

110. Hornick M., E. Marcade, S. Venkayala, Java data mining: strategy, standard and practice, A practical Guide for Architecture, Design, and Implementation, 2006
111. Humphreys, Ed., Implementing the ISO/IEC 27001 ISMS Standard, Second Edition, ISBN 13: 978-1-60807-930-8, 2016, ARTECH HOUSE
112. Hussain, S., Dahan, N. A., Ba-Alwi, F. M. & Ribata, N. (2018). Educational data mining and analysis of students' academic performance using WEKA. Indonesian Journal of Electrical Engineering and Computer Science, 9(2), 447-459.
113. Ishwarappa, J. Anuradha. A Brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology. – Procedia Computer Science, Vol. 48, 2015, pp. 319-324.
114. International Educational Data Mining Society. 2022, <https://educationaldatamining.org>.
115. IMS QTI Standard, www.imsglobal.org/question/index.htm
116. Jadex Active Components: AC Tutorial. 2011. Tutorial Introduction, Jadex Active Components: <http://www.activecomponents.org/bin/view/AC+Tutorial/01+Introduction>
117. Jeong, H., & Biswas, G. (2008). Mining student behavior models in Learning-by-teaching environments. In R. S. J. D. Baker, T. Barnes, & J. Beck (Eds.), Proceedings of the 1st International Conference on Educational Data Mining (pp. 127-136). Montreal, Quebec, Canada.
118. Jiawei Han and Micheline Kamber. 2007. Data Mining: Concepts and Techniques. San Fransisco, CA: Morgan Kaufmann publishers.
119. Jecheva V., D. Orozova, Ontology-Based Electronic Test Result Evaluation, Advances in Intelligent and Soft Computing, Third International Conference of Software, Services and Semantic Technologies S3T, Springer, 213-214, 2011. ISSN: 1867-5662
120. Kabakchieva, D. Predicting Student Performance by Using Data Mining Methods for Classification. – Cybernetics and Information Technologies, Vol. 13(1), 2013, pp. 61-72
121. Kapp, K. (2012). The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education. Pfeiffer.
122. Kaufmann, M., Meier, A., 2009: An Inductive Fuzzy Classification Approach applied to Individual Marketing, In: Proceedings of the 28th North American Fuzzy Information Processing Society Annual Conference, Ohio, USA.
123. Kaushik, A., Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity, Wiley, 2010. <https://www.wiley.com/en-gb/Web+Analytics+2+0%3A+The+Art+of+Online+Accountability+and+Science+of+Customer+Centricity-p-9780470596449>
124. Kinnebrew, J., & Biswas, G. (2012). Identifying learning behaviours by contextualizing differential sequence mining with action features and performance evolution. In K. Yacef, O. Zaïane, H. Hershkovitz, M. Yudelson, & J. Stamper (Eds.), *Proceedings of the 5th International Conference on Educational Data Mining*, pp. 57-64, Chania, Greece.
125. KNIME – Open for Innovation: <https://www.knime.com/>
126. Lee Chae Jang, Taekyun Kim, Dal-Won Park, Daniela Langova-Orozova, Modelling of an Intelligent Training System by a Generalized Net, Issues in Intuitionistic Fuzzy Sets and Generalized Nets, (K. Atanassov, J. Kacprzyk, M. Krawczak Eds.) Wyzsza szkola Informatyki Stosowanej i Zarzadvania, Warszawa, 2004, vol.2, 17-29.
127. Lee C.-S., M.-H. Wang, J.-J. Chen, and C.-Y. Hsu, "Ontology-based Intelligent Decision Support Agent for CMMI Project Monitoring and Control," in Annual meeting of the North American Fuzzy Information Processing Society. NAFIPS 2006, jun. 2006, pp. 627–632.
128. Lee, J. I. & Brunskill, E. (2012). The impact on individualizing student models on

- necessary practice opportunities. In K. Yacef, O. Zaïane, H. HersHKovitz, M. YudelsoN, & J. Stamper (Eds.), *Proceedings of the 5th International Conference on Educational Data Mining*, pp. 118-125, Chania, Greece.
129. Linear-time logic, <http://www.eecs.qmul.ac.uk/~pm/SaR/2004l1t1.pdf>
 130. Liu B. et. al., Intelligent Spaces: An Overview, IEEE International Conference on Vehicular Electronics and Safety, 13-15 Dec. 2007, Beijing, 978-1-4244-1266-2.
 131. Marr B., "Big Data: Using SMART Big Data, Analytics and Metrics To Make Better Decisions and Improve Performance". John Wiley & Sons Ltd, 2015.
 132. Mardani, A., A. Jusoh, K. MD Nor, Z. Khalifah, N. Zakwan, A. Valipour. Multiple criteria decision-making techniques and their applications – a review of the literature from 2000 to 2014. – Economic Research-Ekonomska Istraživanja, Vol. 28(1), 2015, pp. 516-571.
 133. Millington, Ian, 2006. Artificial Intelligence for Games. Elsevier Inc. Ch.7.6 Reinforcement Learning pp. 612-628
 134. Mobasher, B., 2007: Web Usage Mining, In: (Liu 2007, pp. 449-483).
 135. Moodle, Database schema introduction, http://docs.moodle.org/dev/Database_schema_introduction.
 136. Mu, Jin, Karsten Stegmann, Elijah Mayfield, Carolyn Rosé и Frank Fischer, "The ACODEA framework: Developing segmentation and classification schemes for fully automatic analysis of online discussions", International Journal of Computer-Supported Collaborative Learning, vol. 7, no. 2, April 2012.
 137. Nisheva-Pavlova, M. Towards Building an AI Curriculum for High School Students. CEUR Workshop Proceedings, ISSN 1613-0073, Vol. 3191, 2022, pp. 276-284.
 138. Oancea, Bogdan, Raluca Dragoescu и Stefan Ciucu, "Predicting students' results in higher education using neural networks" в International Conference on Applied Information and Communication Technologies (AICT2013), Jelgava, Latvia, 2013.
 139. Orange system [Online]: <https://orange.biolab.si/training/introduction-to-data-mining/>
 140. Orozov B., D. Orozova, An AI application in E-Learning, CEUR Workshop Proceedings 3061, pp. 76-80, September 27–28, 2021, Plovdiv, Bulgaria, ISSN 1613-0073.
 141. Orozov B., D. Orozova, Rule Based System Against Reinforcement Learning, CompSysTech'21: International Conference on Computer Systems and Technologies '21, June 2021, Pages 67–70, ISBN: 978-145038982-2
 142. Orozova D., Generalized Net Model of Tutoring System, Issues In Intuitionistic Fuzzy Sets and Generalized Nets, Vol. 5, 2007, pp. 25-34. ISBN 978-83-88311-90-1
 143. Orozova D., Appropriate E-Test System Selection Model, Comptes rendus de l'Acad'emie bulgare des Sciences, 2019, Vol 72, No 6, pp. 811-820, ISSN 1310-1331
 144. Orozova D., SQL User Queries Execution Model, Advances in Intelligent Systems and Computing book series (AISC, volume 1081), Uncertainty and Imprecision in Decision Making and Decision Support: New Challenges, Solutions and Perspectives, pp.184-189, 2020, ISSN 2194-5357, DOI: 10.1007/978-3-030-47024-1_20
 145. Orozova D., I. Popchev, Cyber-Physical-Social Systems for Big Data, XXI International Symposium on Electrical Apparatus and Technologies SIELA 2020, 3-6 June 2020, Bourgas, Bulgaria, pp. 334 – 337, ISBN 978-1-7281-4345-3
 146. Orozova D., K. Atanassov, Generalized Net Model of Processes Related to Big Data, Comptes rendus de l'Académie bulgare des Sciences, book No 12, vol. 71, 2018, 1679 -1686.
 147. Orozova D., K. Atanassov, Model of Big Data Map/Reduce Processing, Comptes rendus de l'Acad'emie bulgare des Sciences, 2019, Vol 72, No11, pp.1537-1545.

148. Orozova D., M. Todorova, Ontology concept in courses on students, TEM Journal. Volume 7, Issue 3, Pages 693-697, ISSN 2217-8309, DOI: 10.18421/TEM73-29, 2018.
149. Orozova D., M. Krawczak, Project-based Learning Modelled with a Generalized Net, Issues in Intuitionistic Fuzzy Sets and Generalized Nets, Warsaw, Poland, Vol.12, 2015/2016, pp. 68–79, ISBN 978-83-61551-13-3.
150. Orozova D., M. Todorova, Applying Reactive Blocks in Courses of Internet of Things, Proceedings of the 10th Annual International Conference of Education, Research and Innovation, Seville, 16-18 November, 2017, стр.6960-6967, ISSN (print):2340-1095
151. Orozova D., M. Todorova, How to Follow Modern Trends in Courses in "Databases" - Introduction of Data Mining Techniques by Example, Proceedings of the 11th Annual International Technology, Education and Development Conference (INTED), Valencia, 2017, стр.8186-8194, ISSN:2340-1079
152. Orozova D., K. Atanassov, M. Todorova, Generalized Net Model of the Process of Personalization and Usage of an e-Learning Environment, Proceedings of the Jangjeon Mathematical Society 19 (2016), No. 4, pp. 615 – 624. ISSN (print):1598-7264, ISSN (online):2508-7916, ISBN:89-87809-15-3
153. Orozova, S. Hadzhikoleva, E. Hadzhikolev, A Course Gamification Model for the Development of Higher Order Thinking Skills, MIPRO 2021, 44th International Convention, September 27, 2021 – October 1, 202, Opatija, Croatia, pp. 1772-1777, ISSN 1847-3946.
154. Orozova D., E. Sotirova, Modeling of a Learning Management System, XVIII-th International Symposium on Electrical Apparatus and Technologies, Proceedings of digests, SIELA 2014, 29-31 May 2014, Bulgaria, pp. 95-98, ISBN: 978-147995817-7
155. Orozova D., M. Georgieva, A model of the process of Big Data with Generalized Net, Electrical Apparatus and Technologies (SIELA), 2016 19th International Symposium, Bourgas, 238-241, ISSN: 978-1-4673-9521-2, ISBN 978-619-160-648-1
156. Orozova D., A. Ivanov, Generalized Net Model of Virtual Collaboration Space, XX-th International Symposium on Electrical Apparatus and Technologies SIELA 2018, 3-6 June 2018, Bourgas, Bulgaria, ISBN 978-1-5386-3418-9
157. Orozova D., K. Kaloyanova, M. Todorova, Introducing Information Security Concepts and Standards in Higher Education, TEM Journal. Volume 8, Issue 3, Pages 1017-1024, ISSN 2217-8309, DOI: 10.18421/TEM83-46, August 2019
158. Orozova D., N. Hristova, Generalized Net Model for dynamic decision making and prognoses, XXI International Symposium on Electrical Apparatus and Technologies SIELA 2020, 3-6 June 2020, Bourgas, Bulgaria, pp. 330 – 333, ISBN ISBN 978-1-7281-4345-3.
159. Orozova D., N. Hristova, An application of analytical data research in e-learning system, ICTTE 2020, International Conference on Technics, Technologies and Education, Journal IOP Conference Series: Materials Science and Engineering Volume 1031 (2021) 012058, IOP Publishing, Online ISSN: 1757-899X
160. Peneva, V., I. Popchev. Fuzzy multicriteria decision-making. – Cybernetics and Information Technologies, v.2, 2002, No 1, 16-26. ISSN: 1311-9702.
161. Petri C.A., “Kommunikation mit Automaten”. Bonn:Institut fuer Instrumentelle Mathematik, Schriften des IIM Nr.3, 1962. Also, English translation: “Communication with Automata”, New York: Griffiss Air Force Base. Tech. Rep. RAD-TR-65-377, vol.1, Suppl. 1, 1966.
162. Popper, Karl Raimund (1972). Objective knowledge. Oxford, Clarendon Press.
163. Popchev I., D. Orozova, Towards a multistep method for assessment in e-learning of

- emerging technologies, Cybernetics and Information Technologies, Volume 20, No 3, Sofia, 2020, pp.116-129, Print ISSN: 1311-9702
164. Popchev I., D. Orozova. Towards Big Data Analytics in the E-learning Space. Cybernetics and Information Technologies, Vol. 19(3), 2019, pp. 16-25. ISSN: 1311-9702
 165. Popchev, D. Orozova, S. Stoyanov, IoT and Big Data Analytics in E-Learning, 2019 Big Data, Knowledge and Control Systems Engineering (BdKCSE), 2019, pp. 1-5, ISBN 978-172816481-6
 166. Popchev I., D. Orozova, Text Mining in the Domain of Plant Genetic Resources, Proceedings of 2020 IEEE 10th International Conference on Intelligent Systems, 2020, pp. 596-600
 167. Popchev I., D. Orozova, DataScience: Experience and Trends, ICTTE 2020, International Conference on Technics, Technologies and Education, Journal IOP Conference Series: Materials Science and Engineering, Volume 1031 (2021) 012057, IOP Publishing, Online ISSN: 1757-899X, Print ISSN: 1757-8981 doi:10.1088/1757-899X/1031/1/012057.
 168. Popchev I., D. Orozova, Data Analytics Approach in Virtual Educational Space, Big Data, Knowledge and Control Systems Engineering (BdKCSE), 2021, pp. 1-5, doi: 10.1109/BdKCSE53180.2021.9627247, ISBN 978-166541042-7.
 169. Popchev I., D. Orozova, Algorithms for Machine Learning with Orange System, International Journal of Online and Biomedical Engineering (iJOE), Vol. 19, No. 04, 2023, pp. 109 -123, ISSN: 2626-8493, DOI: <https://doi.org/10.3991/ijoe.v19i04.36897>
 170. Popchev, I., I. Radeva, Risk Analysis – an Instrument for Technology Selection. Engineering Sciences (2019) LVI (4) 5-20, ISSN1312-5702 (Print), ISSN 2603-3542 (Online), DOI: 10:7546/ Eng.Sci.LVI.19.04.01.
 171. Popchev, I., I. Radeva, L. Doukovska. Oracles Integration in Blockchain-Based Platform for Smart Crop Production Data Exchange. Electronics 2023, 12, 2244, 1-20. <https://doi.org/10.3390/electronics12102244>
 172. Policy and Investment Recommendations for Trustworthy Artificial Intelligence [Online] <https://ec.europa.eu/digital-single-market/en/news/policy-and-investment-recommendations-trustworthy-artificial-intelligence>
 173. Pfleeger Ch., Sh. Pfleeger, Analyzing Computer Security. A Threat/Vulnerability/Countermeasure Approach, Prentice Hall, ISBN-13:978-0-13-278946-2, 799 pages, 2012.
 174. Protégé Ontology Editor and Knowledge Acquisition System, <http://protege.stanford.edu/>
 175. Radeva, I. Multi-Criteria Models for Clusters Design. – Cybernetics and Information Technologies, Vol. 13(1), 2013, pp. 18-33.
 176. RapidMiner - Depth for Data Scientists: <https://rapidminer.com/>
 177. Reactive Blocks Tutorial: Build a Configurable Eclipse Application via Reactive Blocks, https://agrinode.github.io/docs/Bitreactive_tutorial
 178. Reimann, Peter, "Time is precious: Variable- and event-centred approaches to process analysis in CSCCL research", International Journal of Computer-Supported Collaborative Learning, vol. 4, no. 3, pp. 239–257, September 2009.
 179. Reinforcement Learning, Searching for optimal policies: <https://www.cs.upc.edu/~mmartin/Ag4-4x.pdf>
 180. Richard S. Sutton, Andrew G. Barto, Reinforcement Learning, An Introduction, Second edition, 2018, The MIT Press Cambridge, Massachusetts: <http://www.incompleteideas.net/book/bookdraft2018mar16.pdf>
 181. Russell, S and P. Norvig. Artificial Intelligence: A Modern Approach, Prentice Hall, Upper

- Saddle River, New Jersey, 1995.
182. Saraladevi, B., N. Pazhaniraja, P. Paul, M. S. Basha, P. Dhavachelvan. Big Data and Hadoop a Study in Security Perspective. *Procedia Computer Science*, Vol. 50, 2015, 596-601.
 183. Sarker, I.H. Data Science and Analytics: An Overview from Data-Driven Smart Computing, Decision-Making and Applications Perspective. *SN COMPUT. SCI.* 2, 377 (2021). <https://doi.org/10.1007/s42979-021-00765-8>
 184. Siemens, G, and Long, P, Penetrating the fog-analytics in learning and education. *Asynchronous Learning Networks*, 2011.
 185. SCORM 2004, <http://www.adlnet.gov/scorm/scorm-2004-4th>
 186. Schwab K., *The Fourth Industrial Revolution*. Crown Publishing Group, New York, 2017. ISBN: 978-5247-5886-8, eBook ISBN: 978-1-5247-5887-5
 187. Schaughency E., J. Smith, J. van der Meer, and D. Berg, *Classical Test Theory and Higher Education: Five Questions*, In: *Handbook on Measurement, Assessment, and Evaluation in Higher Education*, Edited by Charles Secolsky and D. Brian Denison, Publ. Routledge, 2017, ISBN: 1138892157.
 188. Shen, W., Hao, Q., Yoon, H. J., Norrie, D. H. Applications of agent-based systems in intelligent manufacturing: An updated review, *Advanced Engineering INFORMATICS*, 20(4), (2006), pp.415–431.
 189. Social Analytics 2022, <https://www.gartner.com/en/information-technology/glossary/social-analytics>.
 190. Sowa J. F., *Conceptual structures: information processing in mind and machine*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1984.
 191. Stancheva, N., A. Stoyanova-Doycheva, S. Stoyanov, I. Popchev, V. Ivanova. An Environment for Automatic Test Generation. – *Cybernetics and information Technologies*, Vol. 17, No. 2, Sofia, 2017, 183-196. Print ISSN: 1311-9702; Online ISSN: 1314-4081. DOI: 10.1515/cait-2007.0025.
 192. Stancheva, N., A. Stoyanova-Doycheva, S. Stoyanov, I. Popchev, V. Ivanova. A Model for Generation of Test Questions. – *Comptes rendus de l'Academie Bulgare des Sciences*. Vol. 70, No. 5, 2017, 619-630.
 193. Stoyanov S., *Context-Aware and Adaptable eLearning Systems*, Internal Report, Software Technology Research Laboratory, De Montfort University, Leicester, UK, August, 2012.
 194. Stoyanov, S., Cholakov, G., Valkanova, V. Sandalski, M., *Personalized, Reactive and Proactive Providing of e-Learning Services*, E-Learn 2011, Honolulu, USA.
 195. Stoyanov, S., I. Popchev, O. Rachneva, A. Rachnev. DeLC – technological environment supporting the transition from CBT to eLearning. - Intern. Scientific Conference “Informatics an the Scientific Knowledge” (Ed. N. Lyutov), Vol. 1, 28-30 June 2006, University Publicatio House, Varna Free University, “Chernorizets Hrabar”, 113-127.
 196. Stoyanov, S., I. Popchev, I., E. Doychev, D. Mitev, V. Valkanov, A. Stoyanova-Doycheva, V. Valkanova, I. Minov. DeLC Educational Portal. – *Cybernetics and Information Technologies*, Vol. 10, 2010, No. 3, 49-69, ISSN: 1311-9702
 197. Stoyanova-Doycheva A., T. Glushkova, S. Stoyanov, D. Orozova, *Lifelong Learning Supported by an Intelligent Tourist Guide*, 2018, AIP Conference Proceedings, Volume 2048, pp. 020038; ISSN: 0094243X, ISBN: 978-073541774-8, DOI: 10.1063/1.5082056
 198. Tarus, J.K., Z. Niu и G. Mustafa, "Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning", *Artificial Intelligence Review*, vol. 50, no. 1, pp. 21–48, June 2018.

199. Tarus, John K., Zhendong Niu и Dorothy Kalui, "A hybrid recommender system for e-learning based on context awareness and sequential pattern mining", *Soft Computing*, vol. 22, no. 8, pp. 2449–2461, April 2018.
200. Todorov, J., V. Valkanov, S. Stoyanov, B. Daskalov, I. Popchev, D. Orozova. Chapter 6: Personal Assistants in a Virtual Education Space, Sgurev V., Jotsov V., Kacprzyk J. (eds) *Practical Issues of Intelligent Innovations. Studies in Systems, Decision and Control*, 140, Springer International Publishing AG part of Springer Nature, 2018, ISSN 2198-4182, ISSN 2198-4190 (electronic), DOI:10.1007/978-3-319-78437-3, 22, 131-153
201. Todorov J., E. Doychev, D. Orozova, A. Stoyanova-Doycheva, IoT Multiagent Assistant in Virtual Educational Space, *AIP Conference Proceedings*, Vol. 2048, 020031; ISSN: 0094243X.
202. Todorova M., D. Orozova, Generalized Net Model of Sequential Programs, In proc of the 20th International Symposium on Electrical Apparatus and Technologies (SIELA 2018), 3-6 June 2018, Bourgas, Bulgaria, DOI: 10.1109/SIELA.2018.8447068, Electronic ISBN: 978-1-5386-3419-6, ISBN: 978-1-5386-3420-2.
203. Todorova M., D. Orozova, Training Difficulties in Deductive Methods of Verification and Synthesis of Program, *International Journal of Advanced Computer Science and Applications (IJACSA)*, Vol. 9, No. 7, 2018, pp. 18-22, ISSN (print):2158-107X, ISSN (online):2156-5570, doi:10.14569/IJACSA.2018.090703.
204. Tuparova D., G. Tuparov, D. Orozova, Educational Computer Games and Gamification in the Higher Education – Students’ Points of View, 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), 2020, Croatia, Opatija, pp.1879-1884 Electronic ISSN: 2623-8764, DOI: 10.23919/MIPRO48935.2020.9245251.
205. Ushold M., M. Gruninger, “Ontologies: Principles, Methods and Applications”, *Knowledge Engineering Review*, vol. 11, no. 2, pp. 93-136, 1996.
206. Unsupervised Learning and Data Clustering: <https://towardsdatascience.com/unsupervised-learning-and-data-clustering-eeecb78b422a>
207. Venema Y., Temporal Logic, <http://staff.science.uva.nl/~yde/papers/TempLog.pdf>
208. Venkatram K., Geetha Mary A., Review on Big Data & Analytics – Concepts, Philosophy, Process and Applications, *Cybernetics and Information Technologies*, Vol. 17(2), 2017, 3-27, ISSN: 1311-9702;
209. Wilges B., G. P. Mateus, R. A. Silveira, S. M. Nassar, “An animated pedagogical agent as a learning management system manipulating intelligent learning objects,” in *IEEE International Conference on Advanced Learning Technologies. ICALT 2007*, 2007, pp. 186–188.
210. Wang, F., "The Emergence of Intelligent Enterprises" From CPS to CPSS, *IEEE Intelligent Systems*, 2010, pp. 85-88.
211. Wang F-Y., Driving into the future with ITS, *IEEE Intelligent System*, Vol.21, No.3, 2006, pp.94–95.
212. Walkenbach, J., *Microsoft Excel 2016 Bible.*: Wiley, Indianapolis, IN, 2015.
213. WEKA Waikato Environment for Knowledge Analysis: <http://www.cs.waikato.ac.nz/ml/weka>
214. Witten, I. H., Frank, E. (2000). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann Publishers, ISBN 1-55860-552-5.
215. Wordclouds - free online generator: www.wordclouds.com/
216. Xhakaj, F., V. Alevan и B.M. McLaren, "Effects of a Teacher Dashboard for an Intelligent Tutoring System on Teacher Knowledge, Lesson Planning, Lessons and Student Learning" in

- Data Driven Approaches in Digital Education. EC-TEL 2017. Lecture Notes in Computer Scienc, É. Lavoué et al., Eds.: Springer, Cham, 2017, ch. vol. 10474, pp. 315-329.
217. Zadeh, L., The concept of a linguistic variable and its application to approximate reasoning, Published 1975, Information Sciences, v. 8, pp. 199–249.
218. Zhang Y., L.-L. Hou, Z.-L. Zhou, and H.-C. Ding, “Multi-agent paradigm and conceptual graphs in information retrieval model,” in Sixth International Conference on Intelligent Systems Design and Applications. ISDA'06, vol. 2, oct. 2006, pp. 875–880.
219. Zlatev Z., J. Ilieva, D. Orozova, G. Shivacheva, N. Angelova, Design and Research of a Sound-to-RGB smart acoustic device, Multimodal Technol. Interact. 2023, 7(8), 79.
220. Zumstein D., M. Kaufmann, A Fuzzy Web Analytics Model for Web Mining, IADIS European Conference on Data Mining, 2009, pp.59-66, ISBN: 978-972-8924-88-1.