Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

FIIT-5220-56347

Bc. Petra Vrablecová

OBJAVOVANIE VZŤAHOV VO VÝUČBOVOM OBSAHU

Diplomová práca

Vedúci práce: Ing. Marián Šimko, PhD.

máj 2013

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

FIIT-5220-56347

Bc. Petra Vrablecová

OBJAVOVANIE VZŤAHOV VO VÝUČBOVOM OBSAHU

Diplomová práca

Študijný program: Softvérové inžinierstvo
Študijný odbor: 9.2.5 Softvérové inžinierstvo
Miesto vypracovania: Ústav informatiky a softvérového inžinierstva, FIIT
Vedúci práce: Ing. Marián Šimko, PhD.

máj 2013

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
Fakulta informatiky a informačných technológií

Zadanie diplomovej práce

Meno študenta: **Bc. Vrablecová Petra**

Študijný program: Softvérové inžinierstvo

Študijný odbor: Softvérové inžinierstvo

Názov práce: **Objavovanie vzťahov vo výučbovom obsahu**

Samostatnou výskumnou a vývojovou činnosťou v rámci predmetov Diplomový projekt I, II, III vypracujte diplomovú prácu na tému, vyjadrenú vyššie uvedeným názvom tak, aby ste dosiahli tieto ciele:

Všeobecný cieľ:

Vypracovaním diplomovej práce preukážte, ako ste si osvojili metódy a postupy riešenia relatívne rozsiahlych projektov, schopnosť samostatne a tvorivo riešiť zložité úlohy aj výskumného charakteru v súlade so súčasnými metódami a postupmi študovaného odboru využívanými v príslušnej oblasti a schopnosť samostatne, tvorivo a kriticky pristupovať k analýze možných riešení a k tvorbe modelov.

Špecifický cieľ:

Vytvorte riešenie, zodpovedúce návrhu textu zadania, ktorý je prílohou tohto zadania. Návrh bližšie opisuje tému vyjadrenú názvom. Tento opis je záväzný, má však rámcový charakter, aby vznikol dostatočný priestor pre Vašu tvorivosť.

Riaďte sa pokynmi Vášho vedúceho.

Pokiaľ v priebehu riešenia, opierajúc sa o hlbšie poznanie súčasného stavu v príslušnej oblasti alebo o priebežné výsledky Vášho riešenia alebo o iné závažné skutočnosti, dospejete spoločne s Vaším vedúcim k presvedčeniu, že niečo v texte zadania a/alebo v názve by sa malo zmeniť, navrhnite zmenu. Zmena je spravidla možná len pri dosiahnutí kontrolného bodu.

Miesto vypracovania: Ústav informatiky a softvérového inžinierstva FIIT STU v Bratislave

Vedúci práce: **Ing. Marián Šimko**

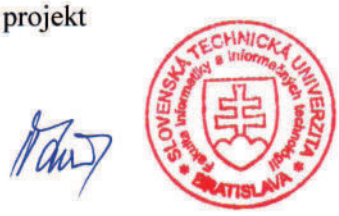
Termíny odovzdania:

podľa harmonogramu štúdia platného pre semester, v ktorom máte príslušný predmet (Diplomový projekt I, II, III) absolvovať podľa Vášho študijného plánu

Predmety odovzdania:

V každom predmete dokument podľa pokynov na www.fiit.stuba.sk v časti:
home > Informácie o > štúdiu > organizácia štúdia > diplomový projekt

V Bratislave dňa 13. 2. 2012



prof. Ing. Pavol Návrat, PhD.
riaditeľ Ústavu informatiky a softvérového inžinierstva

S T U
F I I T
SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
Fakulta informatiky a informačných technológií

Návrh zadania diplomovej práce

Finálna verzia do diplomovej práce [1]

Študent:

Meno, priezvisko, tituly:	Petra Vrablecová, Bc.
Študijný program:	Softvérové inžinierstvo
Kontakt:	petra.vrablecova@gmail.com

Výskumník:

Meno, priezvisko, tituly:	Marián Šimko, Ing.

Projekt:

Názov:	Objavovanie vzťahov vo výučbovom obsahu
Názov v angličtine:	Relationship discovery from educational content
Miesto vypracovania:	Ústav informatiky a softvérového inžinierstva, FIIT STU, Bratislava
Oblasť problematiky:	Dolovanie v texte, spracovanie prirodzeného jazyka

Text návrhu zadania[2]

V posledných rokoch rapídne vzrástol počet výskumných prác zaoberajúcich sa extrakciou vzťahov z textu písanom v prirodzenom jazyku. Výskum sa orientuje hlavne na texty na Webe, ktorý poskytuje obrovské množstvo dát. Vývoj nových metód extrakcie vzťahov v oblastiach objavovania znalostí z textu, akou je automatické zostavovanie ontológií, predstavuje prínos pre reprezentáciu obsahu, čo sa pozitívne odráža do vývoja Webu so sémantikou.

Analyzujte možnosti tzv. odľahčenej reprezentácie sémantiky obsahu v kontexte existujúcich prístupov k extrakcii vzťahov z textu. Sústreďte sa na metódy pre extrakciu vzťahov v doméne vzdelávania. Navrhnite metódu pre automatizovanú extrakciu vybraného typu vzťahu na základe spracovania textu v prirodzenom jazyku. Uvažujte väzby medzi dokumentmi predstavujúcimi vzdelávací obsah aj anotácie, ktoré vytvorili používatelia. Zamerajte sa na vzťahy medzi pojmami slúžiacimi na reprezentáciu obsahu, ktorý je predmetom odporúčania. Pri návrhu vychádzajte z existujúcich metód.

Navrhnutú metódu overte integrovaním do nástroja pre podporu tvorby adaptívneho obsahu a experimentujte s navrhnutým riešením v doméne adaptívneho webového výučbového portálu ALEF.

---

[1] Vytlačiť obojstranne na jeden list papiera

[2] 150-200 slov (1200-1700 znakov), ktoré opisujú výskumný problém v kontexte súčasného stavu vrátane motivácie a smerov riešenia

Literatúra[3]

- WONG, W., LIU, W., BENNAMOUN, M.: Ontology Learning from Text : A Look back and into the Future. In: ACM Computing Surveys, 2011, 36 s.
- CIMIANO, P., VÖLKER, J., STUDER, R.: Ontologies on Demand? A Description of the State-of-the-Art, Applications, Challenges and Trends for Ontology Learning from Text. In: Information, Wissenschaft und Praxis, vol. 57, 2006, no. 6-7., pp. 315-320.
- CVITAS, A.: Relation extraction from text documents. In: MIPRO, 2011, s. 1565-1570.

Vyššie je uvedený návrh diplomového projektu, ktorý vypracoval(a) Bc. Petra Vrablecová, konzultoval(a) a osvojil(a) si ho Ing. Marián Šimko a súhlasí, že bude takýto projekt viesť v prípade, že bude pridelený tomuto študentovi.

V Bratislave dňa 14.1.2012

Vrablecová
Podpis študenta

Podpis výskumníka

Vyjadrenie garanta predmetov Diplomový projekt I, II, III

Návrh zadania schválený: áno / ~~nie~~[4]

Dňa: 7.2.2012

Podpis garanta predmetov

[3] 2-3 vedecké zdroje, každý na samostatnom riadku a s údajmi zodpovedajúcimi bibliografickým odkazom podľa normy STN ISO 690, ktoré sa viažu k téme zadania a preukazujú výskumnú povahu problému a jeho aktuálnosť (uveďte všetky potrebné údaje na identifikáciu zdroja, pričom uprednostnite vedecké príspevky v časopisoch a medzinárodných konferenciách)

[4] Nehodiace sa prečiarknite

STU FIIT
SLOVENSKÁ TECHNICKÁ
UNIVERZITA V BRATISLAVE
FAKULTA INFORMATIKY
A INFORMAČNÝCH TECHNOLÓGIÍ

Návrh zadania diplomovej práce

Revízia č.: 1 [1]

Študent:

Meno, priezvisko, tituly:	Petra Vrablecová, Bc.
Študijný program:	Softvérové inžinierstvo
Kontakt:	petra.vrablecova@gmail.com

Výskumník:

Meno, priezvisko, tituly:	Marián Šimko, Ing. PhD.

Projekt:

Názov:	Objavovanie vzťahov vo výučbovom obsahu
Názov v angličtine:	Relationship discovery from educational content
Miesto vypracovania:	Ústav informatiky a softvérového inžinierstva, FIIT STU, Bratislava
Oblasť problematiky:	Dolovanie v texte, spracovanie prirodzeného jazyka

Text návrhu zadania[2]

V posledných rokoch rapídne vzrástol počet výskumných prác zaoberajúcich sa extrakciou vzťahov z textu písanom v prirodzenom jazyku. Výskum sa orientuje hlavne na texty na Webe, ktorý poskytuje obrovské množstvo dát. Vývoj nových metód extrakcie vzťahov v oblastiach objavovania znalostí z textu, akou je automatické zostavovanie ontológií, predstavuje prínos pre reprezentáciu obsahu, čo sa pozitívne odráža do vývoja Webu so sémantikou.

Analyzujte možnosti tzv. odľahčenej reprezentácie sémantiky obsahu v kontexte existujúcich prístupov k extrakcii vzťahov z textu. Sústreďte sa na metódy pre extrakciu vzťahov v doméne vzdelávania. Navrhnite metódu pre automatizovanú extrakciu vybraného typu vzťahu na základe spracovania textu v prirodzenom jazyku. Uvažujte aj väzby medzi dokumentmi predstavujúcimi vzdelávací obsah. Zamerajte sa na vzťahy medzi pojmami slúžiacimi na reprezentáciu obsahu, ktorý je predmetom odporúčania. Pri návrhu vychádzajte z existujúcich metód.

Navrhnutú metódu overte integrovaním do nástroja pre podporu tvorby adaptívneho obsahu a experimentujte s navrhnutým riešením v doméne adaptívneho webového výučbového portálu ALEF.

---

[1] Vytlačiť obojstranne na jeden list papiera

[2] 150-200 slov (1200-1700 znakov), ktoré opisujú výskumný problém v kontexte súčasného stavu vrátane motivácie a smerov riešenia

Literatúra[3]

- WONG, W., LIU, W., BENNAMOUN, M.: Ontology Learning from Text : A Look back and into the Future. In: ACM Computing Surveys, 2011, 36 s.
- CIMIANO, P., VÖLKER, J., STUDER, R.: Ontologies on Demand? A Description of the State-of-the-Art, Applications, Challenges and Trends for Ontology Learning from Text. In: Information, Wissenschaft und Praxis, vol. 57, 2006, no. 6-7., pp. 315-320.
- CVITAS, A.: Relation extraction from text documents. In: MIPRO, 2011, s. 1565-1570.

Vyššie je uvedený návrh diplomového projektu, ktorý vypracoval(a) Bc. Petra Vrablecová, konzultoval(a) a osvojil(a) si ho Ing. Marián Šimko, PhD. a súhlasí, že bude takýto projekt viesť.

V Bratislave dňa 13.2.2013

Podpis študenta

Podpis výskumníka

Vyjadrenie garanta predmetov Diplomový projekt I, II, III

Návrh zadania schválený: áno / ~~nie~~[4]

Dňa: 1.2.2013

Podpis garanta predmetov

[3] 2 vedecké zdroje, každý v samostatnej rubrike a s údajmi zodpovedajúcimi bibliografickým odkazom podľa normy STN ISO 690, ktoré sa viažu k téme zadania a preukazujú výskumnú povahu problému a jeho aktuálnosť (uveďte všetky potrebné údaje na identifikáciu zdroja, pričom uprednostnite vedecké príspevky v časopisoch a medzinárodných konferenciách)

[4] Nehodiace sa prečiarknite

Anotácia

Slovenská technická univerzita v Bratislave

FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLÓGIÍ

Študijný program: SOFTVÉROVÉ INŽINIERSTVO

Autor: Bc. Petra Vrablecová

Diplomová práca: Objavovanie vzťahov vo výučbovom obsahu

Vedúci diplomovej práce: Ing. Marián Šimko, PhD.

máj, 2013

Doménový model je základná súčasť adaptívneho výučbového systému. Slúži na opis sémantiky výučbového obsahu formou metadát. My uvažujeme, že je to „ľahká“ ontológia skladajúca sa z relevantných doménových termov a vzťahov medzi nimi. Jeho manuálne vytvorenie je náročná úloha pre učiteľa, preto je snaha ju automatizovať.

V tejto práci sme navrhli a overili metódu pre extrakciu metadát z výučbového obsahu, zameranú na vzťahy medzi relevantnými doménovými termami. Využili sme existujúce metódy a upravili sme ich pre doménu vzdelávania. Naša metóda je založená na štatistickom prístupe. Je jazykovo nezávislá a aplikovateľná na akýkoľvek výučbový obsah. Integrovali sme ju do systému pre správu výučbového obsahu, kde ju môžu využívať učitelia. Metóda dokázala objaviť vzťahy, ktoré sa nepodarilo objaviť metódou založenou na lingvistickom spracovaní. Naša práca je prínosom pre rozrastajúcu sa oblasť automatizovaného získavania doménového modelu.

Annotation

Slovak University of Technology Bratislava

FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES

Degree Course: SOFTWARE ENGINEERING

Author: Bc. Petra Vrablecová

Master's Thesis: Relationship Discovery from Educational Content

Supervisor: Ing. Marián Šimko, PhD.

2013, May

The domain model is an essential part of adaptive learning system. It expresses the semantics of educational content in the form of metadata. We consider it to be a lightweight ontology, i.e., a set of terms and relations. Manual domain model building is a challenging task for teachers, hence there is an effort to automate it.

We designed and verified a method for automated acquisition of metadata from educational content, aimed at relationships discovery between terms. We exploit existing methods for relationship discovery from text and adopt them for the educational domain. Our method is based on statistical approach. It is language independent and it can be applied on any educational content. It was able to discover relationships which was not discovered by method based on linguistic processing. Our work is promising contribution to the growing field of automated domain model acquisition.

Ďakujem môjmu vedúcemu práce, Ing. Mariánovi Šimkovi, PhD., za cenné rady, pripomienky, námety a motiváciu počas celého vypracovávania projektu.

Obsah

1 Úvod

Aby človek dokázal poňať realitu v celej jej zložitosti, vytvorili sa uňho prirodzene schopnosti, ktoré mu umožňujú realitu zjednodušiť a opísať. Je to najmä schopnosť abstrahovať spoločné vlastnosti skupiny objektov, vytvárať ich zjednodušené reprezentácie a zoraďovať tieto reprezentácie do hierarchií. Vďaka tejto schopnosti je možné potom opísať realitu pojmami, ktoré môžeme klasifikovať podľa úrovne abstrakcie od najvšeobecnejších (napr. tekutina) po najšpecifickejšie (napr. Coca-Cola). Navyše človek sa naučil odvodzovať pravidlá, ktoré mu uľahčujú klasifikovať objekty a opisovať ich pojmami na správnej úrovni abstrakcie, tzv. axiómy (napr. Tekutina nedokáže udržať tvar.).

Vďaka svojim kognitívnym schopnostiam človek na rozdiel od stroja automaticky porozumie opisu reality – textu, ktorý je vytvorený niekým iným. Bohužiaľ človek nie je schopný si zapamätať a orientovať sa v obrovskom množstve informácií, ktoré dnes existuje a neustále sa zväčšuje. Preto sú informácie strojovo spracované. Aby kooperácia medzi človekom a technológiami narábajúcimi s informáciami bola pre človeka čo najefektívnejšia a najzrozumiteľnejšia, snažíme sa, aby informácie boli spracovávané na základe ich významu (napr. vo vyhľadávačoch, odporúčačoch a pod.). Preto je nutné dať formálnu podobu znalostiam, ktoré človek bežne využíva pri vnímaní. To znamená, že musí byť vytvorený formálny opis domény – oblasti, z ktorej sa majú spracovávať informácie.

Formálny opis domény predstavuje štruktúra skladajúca sa z pojmov (konceptov), medzi ktorými existujú vzťahy. Súčasťou opisu domény môžu byť aj k nej sa vzťahujúce axiómy. Zostavenie úplného opisu domény je však aj pre doménového odborníka náročné, a preto je snaha tento proces automatizovať a sú vyvíjané metódy pre objavovanie znalostí. V tejto práci sa zaoberáme objavovaním vzťahov.

Zameriavame sa na doménu vzdelávania, konkrétne adaptívne výučbové systémy. Ich výučbový obsah tvoria väčšinou výučbové texty, ale môže obsahovať napr. aj multimediálne časti alebo ľubovoľné entity slúžiace na výučbu. Aby adaptívny systém mohol sledovať progres používateľov v učení a prispôsobovať im náležite svoj obsah, potrebuje opis sémantiky výučbového obsahu vo forme metadát. Metadáta predstavujú doménový model skladajúci sa z konceptov prepojených vzťahmi. Všetok výučbový obsah vrátane doménového modelu je vytváraný pedagógmi. Manuálne vytvorenie doménového modelu vyžaduje vynaloženie veľkého úsilia od pedagóga, preto je snaha automatizovať tento proces alebo aspoň jeho časť. Sústredili sme sa na extrakciu metadát z textového obsahu adaptívneho výučbového portálu s orientáciou na vzťahy medzi konceptmi.

Analyzovali sme výučbový obsah a doménový model v adaptívnych výučbových systémoch a možnosti objavovania vzťahov z tohto obsahu (kapitoly 2, 3 a 4). Následne sme navrhli metódu pre objavovanie vzťahov medzi konceptmi z výučbového obsahu (kapitola 5). Metódu sme overili na dátach z adaptívneho výučbového systému ALEF [37], ktorý je využívaný na fakulte v rámci výučby. Jej výsledky sme porovnali s výsledkami existujúcej metódy pre objavovanie vzťahov [35] (kapitola 6). Metódu sme úspešne integrovali do systému COME^{2}T [14], ktorý slúži pre správu výučbového obsahu systému ALEF. Zhodnotenie práce sa nachádza v kapitole 7.

2 Reprezentácia obsahu

Pre reprezentovanie obsahu bola zavedená *ontológia*, ktorá predstavuje *„formálnu explicitnú špecifikáciu zdieľanej konceptualizácie“* [17], t.j. zjednodušeného opisu reality. Je to množina pojmov, ktoré môžu byť hierarchicky usporiadané, prípadne môžu medzi nimi existovať iné vzťahy; a sú jednoznačne definované a spoločne používané pre opis domény, ku ktorej sa vzťahujú. V závislosti od zložitosti formalizmu ontológie je možné rozlíšiť tzv. *„ľahké“* (angl. *lightweight*) a *„ťažké“* (angl. *heavyweight*) ontológie [35, 38, 40].

„Ľahké“ ontológie sa skladajú zvyčajne iba z konceptov, medzi ktorými existujú vzťahy. Formálne je takáto ontológia definovaná ako trojica

$$O = \{N, E, C\}$$

kde *N* je množina uzlov a *E* je množina hrán medzi uzlami. Spolu vytvárajú strom. *C* je množina konceptov a platí, že jeden koncept z množiny C je reprezentovaný práve jedným uzlom z množiny N a ak je uzol z množiny N rodič iného uzla z tejto množiny, potom tento vzťah platí aj pre koncepty z množiny C, ktoré tieto uzly reprezentujú [16].

Na rozdiel od klasickej definície konceptov [3], podľa ktorej koncept je abstraktná trieda, ktorá je reprezentovaná v korpuse konkrétnymi lexikálnymi formami, koncepty „ľahkej“ ontológie sú zvyčajne len špecifické pojmy z domény. Keďže opisujú istú doménovú oblasť, nazývajú sa aj relevantné doménové termy (angl. *relevant domain term*, skratka *RDT*). „Ľahká“ ontológia môže predstavovať iba súbor pojmov - termov, slovník, ktorý má striktne definovanú slovnú zásobu, tezaurus alebo taxonómiu, v ktorých existujú medzi termami aj vzťahy [16].

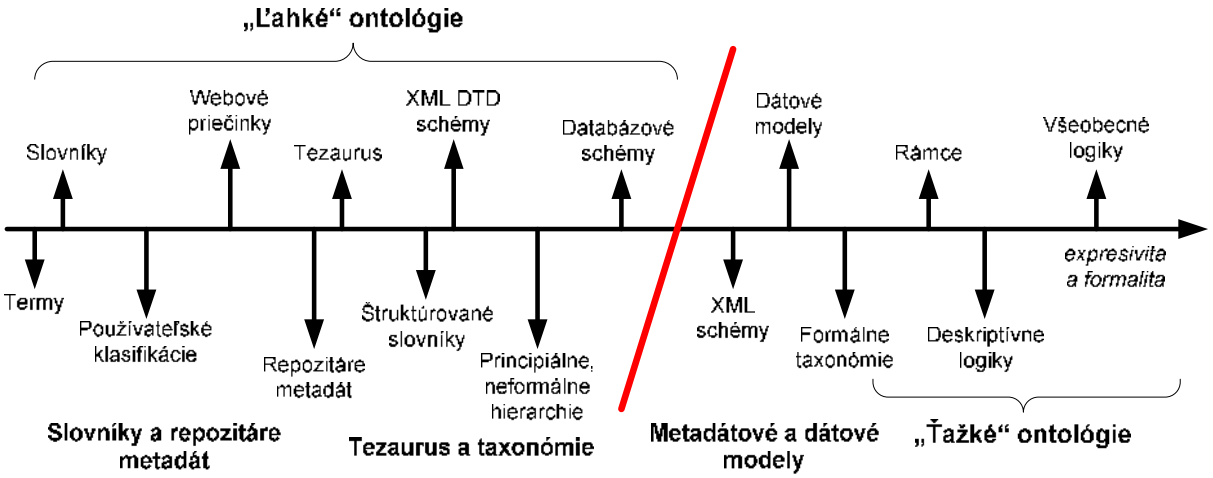
„Ťažké“ ontológie obsahujú aj axiómy, ktoré poskytujú bohatšie možnosti na sémantickú reprezentáciu obsahu a umožňujú automatizované odvodzovanie ďalších informácií o doméne [35, 40]. Axiómy jednoznačne definujú koncepty a vzťahy medzi konceptmi. Takéto ontológie sú využívané zvyčajne na modelovanie veľmi všeobecných konceptov (angl. *upper onotology*), ktoré sú zdieľané naprieč doménami, pretože modely postavené na týchto ontológiách sú konzistentné a presné [15, 29]. Formálne je *„ťažká“* ontológia definovaná ako [3]

$$O = \{C, \leq_C, R, \sigma_R, \leq_R, A, \sigma_A, T\}$$

kde *C*, *R*, *A* a *T* sú štyri disjunktné množiny, ktoré predstavujú množinu konceptov, množinu vzťahov, množinu atribútov opisujúcich koncepty a množinu dátových typov použitých na vyjadrenie atribútov. $\leq_C$ predstavuje hierarchiu konceptov, $\leq_R$ predstavuje hierarchiu vzťahov a σ_R a σ_A predstavujú funkcie na priradenie vzťahov medzi konceptmi a priradenie atribútov ku konceptom.

Na obrázku (Obr. 1) sú zobrazené možné formy ontológií.

Vytvorenie ontológie je náročná úloha a podieľa sa na nej zvyčajne súbor odborníkov z domény. V snahe uľahčiť tento proces vznikla výskumná oblasť učenie ontológií z textu (angl. *ontology learning from text*), ktorá sa zaoberá metódami pre automatické vytvorenie ontológií. Je to proces identifikovania pojmov, konceptov a vzťahov z textovej informácie a ich využitie pri zostavovaní ontológie [40].



Obr. 1. Rôzne druhy ontológií [40].

2.1 Vzťahy medzi konceptmi

Objavovanie vzťahov je samostatná disciplína v rámci učenia ontológií. Výskumom v tejto oblasti sa zaoberá čoraz viac odborníkov, čomu zodpovedá aj nárast publikovaných odborných prác [40].

Vzťahy medzi konceptmi je možné rozdeliť na:

- taxonomické
- a netaxonomické.

Pomocou *taxonomických* vzťahov sa modeluje hierarchia konceptov. Sú to teda vzťahy reprezentujúce nadradenosť/podradenosť konceptov (angl. *hypernymy/hyponymy*). Na modelovanie hierarchií sa najviac vyžíva vzťah `is-a` [40]. Tento vzťah klasifikuje koncepty podľa ich spoločných vlastností, ktoré podradené koncepty dedia od nadradených [29].

Netaxonomické vzťahy predstavujú všetky ostatné vzťahy okrem hierarchických reprezentujúce komplexnejšie väzby medzi konceptmi, napr. roly, vlastnosti, príčinné alebo významové súvislosti [40].

Patrí sem napríklad vzťah `part-of`. Na rozdiel od vzťahu `is-a`, ktorý sa zameriava na identitu konceptov (ich vlastnosti a dedičnosť), vzťah `part-of` sa zameriava na organizáciu konceptov do celkov (angl. *meronymy*, opačný vzťah *holonymy*). Správne pochopenie týchto dvoch vzťahov je nevyhnutné pre vytvorenie stabilnej zmysluplnej ontológie, ktorá môže byť využívaná širšou verejnosťou [29].

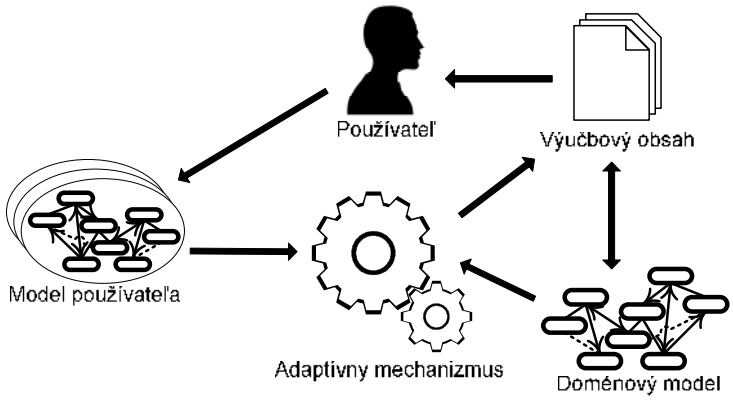
Ďalej sa sem radia vzťahy podobnosti. Tieto vzťahy vyjadrujú sémantickú podobnosť dvoch konceptov. Sú využívané napríklad pri hľadaní synoným (vzťah rovnosti, angl. *equality*) alebo pri hľadaní súvislostí medzi konceptmi, ktoré môžu byť vyjadrené napríklad vzťahom `related-to`.

Netaxonomické vzťahy sú väčšinou odvádzané analyzovaním syntaktických štruktúr a ich závislostí. Často sú identifikovateľné na základe slovies vo vetách, napr. zo spojenia “...rieka preteká cez Bratislavu...“ môže byť odvodený vzťah *pretekať_cez(Bratislava)*. Spolu s identifikáciou vzťahov sa teda objavuje aj ďalšia úloha, a to ich správne pomenovanie [40].

2.2 Reprezentácia obsahu vo výučbovom systéme

Obsah výučbového portálu je zvyčajne zložený z hierarchie dokumentov a doménového modelu, ktorý je prepojený na obsah dokumentov. Aby výučbový systém poskytol informácie používateľovi čo najefektívnejšie, snaží sa prispôsobiť poskytovaný obsah potrebám používateľa. Doménový model tvorí základ pre modelovanie používateľa a je využívaný mechanizmami pre prispôsobovanie obsahu. Základná schéma adaptívneho výučbového systému je znázornená na obrázku (Obr. 2).

Pre reprezentáciu doménového modelu sú najčastejšie využívané „ľahké" ontológie, pretože sú ľahko zostrojiteľné človekom, v porovnaní s „ťažkými" ontológiami ich je jednoduchšie naplniť pomocou metód pre objavovanie znalostí z textu, operácie nad nimi sú výpočtovo menej náročné [35].



Obr. 2. Schéma adaptívneho výučbového systému.

Pre účely adaptívneho výučbového systému sú okrem vzťahov medzi konceptmi z doménového modelu dôležité aj vzťahy mimo doménového modelu. Sú to vzťahy medzi:

- *konceptmi a dokumentmi*, ktoré slúžia na previazanie doménového modelu s výučbovým obsahom, vďaka týmto vzťahom je možné pre systém upravovať model používateľa, napríklad vyhodnotiť, ktoré koncepty sa už naučil a na základe týchto informácií odporučiť ďalší obsah,
- *dokumentmi navzájom*, ktoré sú užitočné pri tvorbe hierarchie dokumentov alebo pri odporúčaní dokumentov, lebo vzťahy môžu vyjadrovať sémantickú súvislosť obsahu dokumentov.

Príkladmi adaptívnych výučbových systémov, ktoré využívajú „ľahkú" ontológiu pre reprezentáciu doménového modelu sú systémy ELM-ART [42], LearnFit [11], ALEF [37]. Príklad systému, ktorý využíva pre opis doménového modelu bohatšiu sémantickú reprezentáciu, je systém Personal Reader [9]. Pri opise systémov sme sa zameriavali na to, ako majú reprezentovaný doménový model a ako je prepojený s obsahom.

2.2.1 ELM-ART

Systém ELM-ART (Episodic Learner Model - The Adaptive Remote Tutor) je interaktívny adaptívny systém pre výučbu programovacieho jazyka Lisp, ktorý vznikol v roku 1996 v Nemecku. Obsah tohto systému je reprezentovaný vo forme elektronickej učebnice. Dokumenty sú hierarchicky zoradené a každá časť je pokrytá úlohami, ktoré by mal používateľ vyriešiť, aby systém ohodnotil jeho vedomosti. Doménový model je v tomto systéme tvorený hierarchiou konceptov a hierarchiou

pravidiel aplikovateľných na koncepty. Pravidlá opisujú ako možno využiť koncepty pri riešení úloh. Opisujú teda ich správne používanie, ale aj neoptimálne alebo nesprávne používanie [41, 42].

2.2.2 LearnFit

Projekt LearnFit je doplnok k populárnemu výučbovému systému Moodle, ktorý prispôsobuje obsah používateľovi. Vznikol v roku 2008 v Egypte. Výučbový obsah systému je rozdelený na kurzy, ktoré majú kapitoly a tie sa až skladajú z dokumentov. Koncepty sa viažu na kapitoly a nie až na jednotlivé dokumenty. Doménový model je reprezentovaný sieťou konceptov [11, 12].

2.2.3 ALEF

Systém ALEF (Adaptive LEarning Framework) je adaptívny výučbový systém, ktorý sa používa na FIIT STU v Bratislave. Vznikol v roku 2009. Výučbový obsah je v tomto systéme rozdelený do kurzov. Každý kurz obsahuje súbor dokumentov. Dokumenty sú hierarchicky zoradené do kapitol a sekcií podľa knižnej predlohy. Okrem dokumentov obsahuje systém ešte obsah vytvorený používateľmi – poznámky k dokumentom (napr. upozornenia na chyby, poznámky vytvorené pri učení). Doménový model adaptívneho kurzu tvorí „ľahká“ ontológia zložená z relevantných doménových termov a vzťahov medzi nimi. Vzťahy sú rôznych typov, nie len hierarchické (napr. *related-to*, *prerequisity-to*). Doménový model je napojený na kurz pomocou vzťahov medzi termami a dokumentmi. Ku každému dokumentu je priradená množina relevantných doménových termov [37].

2.2.4 Personal Reader

Framework Personal Reader slúži na personalizáciu a odporúčanie obsahu používateľovi. Vznikol v roku 2004 v Nemecku. Pre použitie frameworku ako vzdelávací systém je nutný opis obsahu, ktorý má byť odporúčaný a opis domény, ku ktorej sa obsah viaže. Opisy sú vytvorené pomocou frameworku RDF a používajú metadáta Dublin Core. Doménový model je reprezentovaný ontológiou – hierarchiou konceptov poprepájaných vzťahom *subConceptOf*. Medzi konceptmi existuje ešte druhý typ vzťahu, a to *isPrerequisiteFor*. Koncepty sú naviazané na dokumenty cez entitu *ConceptRole*, čiže koncepty vystupujú v úlohách, ktoré hrajú v dokumentoch. Dokumenty, koncepty aj úlohy konceptov môžu formovať hierarchie [9, 20].

3 Objavovanie vzťahov

Úlohy v doméne objavovania vzťahov sú [3]:

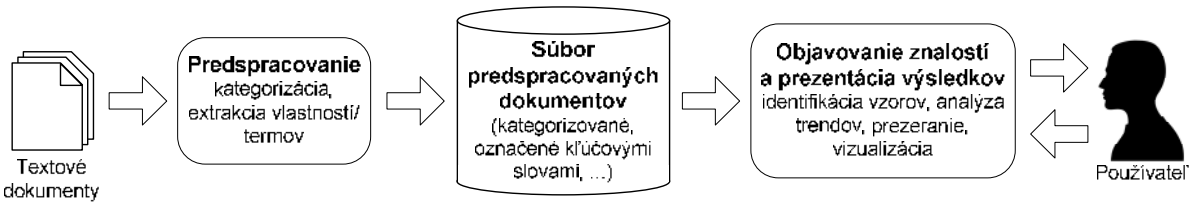
- hľadanie konceptov, ktoré sú v netaxonomickom vzťahu,
- hľadanie správneho označenia vzťahu,
- určovanie správnej úrovne abstrakcie vzťahu vzhľadom na hierarchiu konceptov a rozsah vzťahu,
- určenie hierarchického poradia vzťahov.

V tejto práci budeme riešiť prvú úlohu, čiže hľadanie konceptov, medzi ktorými existuje vzťah relevantný k doméne, ktorú chceme modelovať. Nemusíme sa však obmedzovať iba na netaxonomické vzťahy. Na základe zvoleného spôsobu objavovania vzťahov, môžeme našu úlohu formulovať ako objavovanie všetkých možných vzťahov medzi konceptmi alebo objavovanie zúžiť len na definovaný typ vzťahu. Špecifikovaním typu vzťahu sa môžeme sústrediť len na vzťahy dôležité pre doménu vyučovania, pre ktorú budeme generovať doménový model.

Pri objavovaní vzťahov v doméne vyučovania môžeme využiť:

- *spracovanie obsahu* - získavanie vzťahov spracovaním textu dokumentov pomocou metód pre spracovanie prirodzeného jazyka,
- *spracovanie štruktúry obsahu* - získavanie vzťahov pomocou prepojení konceptov na dokumenty, prepojení medzi dokumentmi, prepojení na iné stránky, pomocou priradených anotácií a pod.

Proces objavovania vzťahov sa dá opísať všeobecným procesom objavovania znalostí z textu (Obr. 3), ktorý sa skladá zo štyroch hlavných častí, a to (a) predspracovanie, (b) objavovanie znalostí, (c) prezentácia výsledkov, (d) optimalizácia výsledkov [13].



Obr. 3. Proces objavovania znalostí [13].

Úlohou predspracovania (angl. *preprocessing*) je transformovať vstupné dokumenty do strojom spracovateľného formátu, nad ktorým sa dajú vykonávať ďalšie operácie. Pri predspracovaní textu sú využívane najmä metódy predspracovania prirodzeného jazyka, ktoré sú bližšie opísané v prílohe (Príloha A). Súčasťou predspracovania môžu byť aj ďalšie úlohy, napr. extrakcia konceptov.

Objavovanie znalostí je kľúčová časť celého procesu. V tejto fáze sa vykonávajú nad predspracovanými dokumentmi operácie na odvodenie znalostí. Možné prístupy k objavovaniu vzťahov sú opísané v kapitole 3.1. Výstupom sú extrahované znalosti, konkrétne vzťahy medzi konceptmi.

Prezentácia výsledkov zahŕňa zobrazovanie a manipuláciu s výsledkami prostredníctvom grafického používateľského rozhrania alebo formou dopytov nad výsledkami.

V rámci optimalizácie výsledkov sa môžu ešte následne vykonávať rôzne operácie (angl. *postprocessing*), napr. zhlukovanie, zoraďovanie, zovšeobecňovanie, odstránenie zbytočných dát a pod.

3.1 Spracovanie obsahu

Spracovanie obsahu predstavuje spracovanie textu množiny dokumentov pomocou metód pre spracovanie prirodzeného jazyka (Príloha A) a následné aplikovanie metódy pre objavovanie vzťahov na predspracovaný text. Metóda môže byť postavená na štatistických, lingvistických alebo logických prístupoch. Najčastejšie sa však v praxi využívajú hybridné prístupy [40].

3.1.1 Štatistické metódy

Štatistické metódy sú väčšinou odvodené z metód pre dolovanie dát, získavanie informácií a strojového učenia. Štatistické metódy neuvažujú v dostatočnej miere sémantiku a vzťahy medzi dokumentmi [40].

Podobnosť a vektorový model

Tento prístup vychádza z tzv. distribučnej hypotézy (angl. *distributional hypothesis*), ktorá pochádza z lingvistiky a hovorí, že slová, ktoré sa nachádzajú v rovnakom kontexte, zvyknú mať podobný význam [18]. Každé slovo je reprezentované vektorom – jeho kontextom, čo môže byť celý dokument, určité okolie slova alebo konštrukcie, v ktorých slovo vystupuje, napr. prísudok-predmet, prísudok-podmet, prívlastky, prístavky. Porovnávaním vektorov sa dá následne určiť podobnosť alebo príbuzenstvo dvoch slov. Pre porovnávanie sa zvyčajne používajú metriky podobnosti a vzdialenosti vektorov. Najčastejšie používanými sú [3]:

- metriky binárnej podobnosti - *Diceov* a *Jaccardov* koeficient,
- metriky geometrickej podobnosti - kosínusová vzdialenosť, *Minkovského* metrika, *Manhattanská* metrika, Euklidovská vzdialenosť,
- metriky založené na pravdepodobnostnom rozdelení - príbuzenská entropia, vzájomná informácia, bodová vzájomná informácia, *Jensen-Shannon* a *Skew* odlišnosti.

Latentná sémantická analýza. LSA (angl. *Latent Semantic Analysis*) sa dá aplikovať na vektorový model. Je to zmenšenie dimenzie vektorov, ktoré predstavujú kontexty slov; pomocou metódy singulárneho rozkladu (*singular vector decomposition - SVD*). Takto upravené vektory sa dajú následne ľahšie porovnávať pomocou spomínaných metrík podobnosti. V prípade LSA sa používa kosínusová vzdialenosť, pričom hodnoty blížiace sa k 1 predstavujú veľmi podobné slová a hodnoty blížiace sa k 0 predstavujú veľmi odlišné slová [3].

V práci [36] bol využitý vektorový model pre výpočet podobnosti konceptov. Na určenie podobnosti bola použitá metrika kosínusová vzdialenosť. Na základe podobnosti boli vybraté najvhodnejšie kandidáty, s ktorými koncept môže byť vo vzťahu. Metóda bola overená aplikovaním na testovacie dáta a vyhodnotená pomocou metrík úplnosť, presnosť a F-metrika. Pri ich výpočte sa porovnávali vygenerované vzťahy so vzťahmi z doménového modelu vytvoreného odborníkom. Táto metóda mala presnosť 50,9 % a úplnosť 59,4 %.

Metóda pre objavovanie vzťahov v práci [6] hľadá vzťahy vyplývajúce z kontextu slov, ktoré patria do triedy konceptu. Trieda konceptu je vopred známa množina slov, ktoré spolu tvoria jeden koncept. Pre každé slovo je nájdený v korpuse kontext, v ktorom sa vyskytuje. Kontext je definovaný rozsahom podľa vopred definovaného symetrického vzoru, tzv. S-vzoru. Vzor však nie je definovaný pomocou lexikálnych alebo syntaktických jednotiek, ale musí obsahovať slová vyskytujúce sa v korpuse s frekvenciou, ktorá je ohraničená definovanými prahmi (napr. HSHX, kde H je reťazec slov vyskytujúcich sa častejšie ako 100 krát na milión slov, S je slovo z triedy konceptu a X je slovo, ktoré sa vyskytuje menej ako 1000 krát na milión slov a má dostatočne vysokú súvislosť so slovom S podľa metriky bodová vzájomná informácia). Následne sú pre každé slovo z triedy konceptu zoskupené podobné kontexty, ktoré predstavujú ten istý typ vzťahu. Nakoniec sú vytvorené zhluky podobných typov vzťahov rôznych slov z triedy konceptu a sú generované dvojice slov, medzi ktorými sa nachádza daný vzťah. Názvy vzťahov boli odvodené z kontextov slov (napr. *capital-of(Luanda, Angola)*, *lake-found-in(Marion, Catfish))*, *star-in(Antares, Scorpius)*). Metóda bola aplikovaná na známe domény (krajiny, súhvezdia, druhy rýb), overenie teda prebiehalo porovnaním výsledkov so zlatým štandardom získaným z encyklopedických vedomostí a vyhodnotením pomocou metriky úplnosť a presnosť. Metóda mala presnosť 90 % a úplnosť 86 % pre doménu krajín, presnosť 68 % a úplnosť 93 % pre doménu hviezd a presnosť 81 % a úplnosť 71 % pre doménu rýb.

Zhlukovanie

Zhlukovanie (angl. *clustering*) znamená zoskupovanie podobných pojmov na základe ich príbuznosti alebo podobnosti. Zoskupené slová môžu vytvoriť jeden koncept alebo môžu byť vytvorené medzi nimi vzťahy. Zhlukovanie možno využiť aj pri tvorbe hierarchie konceptov.

Latentná Dirichletova alokácia. LDA (angl. *Latent Dirichlet Allocation*) delí slová do okruhov podľa témy, ku ktorej sa vzťahujú. Využíva na to podmienenú pravdepodobnosť výskytu slova v dokumente a v danom okruhu. Výsledkom je množina okruhov, v ktorých sa nachádzajú podobné slová, medzi ktorými môže existovať vzťah. Príslušnosť slova k nejakému okruhu má Dirichletove pravdepodobnostné rozdelenie. Je podobná metóde LSA a takisto sa dá využiť pri objavovaní vzťahov podobnosti a príbuznosti.

Formálna analýza konceptov. FCA (angl. *Formal Concept Analysis*) je metóda pre odvodzovanie hierarchie konceptov z množiny objektov a ich vlastností. Každý koncept je tvorený množinou objektov, ktoré majú rovnaké vlastnosti. Koncepty, ktoré sa nachádzajú nižšie v hierarchii, obsahujú podmnožinu objektov nadradených konceptov. Objekty majú viac spoločných vlastností, teda nižšie koncepty sú špecifickejšie. Objekty môžu predstavovať kľúčové slová z textu a ich vlastnosti môžu byť odvodené napr. zo slovesných rámcov, v ktorých pojmy vystupujú [3].

Kolokácie

Metódy skúmajúce kolokácie slov v texte využívajú pri objavovaní vzťahov pravdepodobnosť spoločného výskytu dvoch slov v dokumente, odseku alebo korpuse [21, 22, 26, 44]. Tá môže byť počítaná napríklad pomocou frekvencie výskytu slov v dokumente, korpuse a pod. Táto technika sa nazýva testovanie hypotéz. Testovanie hypotéz skúma sa, či sa dve slová vyskytujú spolu v texte častejšie ako by to predpovedala náhoda. Nulová hypotéza tvrdí, že výskyt dvoch slov v texte je čisto náhodný. Alternatívna hypotéza tvrdí, že výskyt slov je zapríčinený ich súvislosťou. Na základe štatistického testu je nulová hypotéza prijatá alebo zamietnutá. Štatistický test vráti pravdepodobnosť, s akou sme sa pomýlili pri zamietnutí nulovej hypotézy. Pravdepodobnosť je porovnávaná

s definovaným prahom, po ktorý sú výsledky štatistický významné. Ako štatistický test sa najčastejšie používa Studentov test alebo χ^2 test [3], dajú sa však využiť aj metriky ako vzájomná informácia, kosínusová podobnosť, prípadne iné [40]. Štatistický test zároveň vyjadruje váhu objaveného vzťahu. Nevýhodou tohto prístupu je, že identifikuje vzťahy neznámych typov, pretože kolokácie môžu existovať v mnohých formách, od fráz ako „trójsky kôň“, „dobrý deň“ po asociácie ako „les - zver“, „Steve - Apple“ [3].

Začleňovanie pojmov

Pri začleňovaní pojmov (angl. *term subsumption*) je využitá podmienená pravdepodobnosť výskytu pojmov v dokumentoch, aby sa medzi pojmami vytvorili vzťahy nadradenosti/podradenosti. Váha takéhoto vzťahu vyjadruje, o koľko je prvý pojem všeobecnejší ako druhý pojem. Podmienkou pre existenciu vzťahu medzi dvomi pojmami je, že pojmy sa musia spolu nachádzať v korpuse s pravdepodobnosťou aspoň 80 %. Potom pojem z dvojice, ktorý sa nachádza v korpuse častejšie je označený za všeobecnejší. Táto metóda je preto účinnejšia nad korpusom, ktorý používa užšiu škálu slov a frekventovane sa v ňom objavujú kľúčové pojmy, napríklad vedecké texty s definovanou slovnou zásobou. Tento nedostatok sa dá ale obísť zoskupením synoným do konceptov a hľadaním vzťahov medzi týmito konceptmi [31, 33].

Objavovanie na základe združovacích pravidiel

Vstupom pre objavovanie na základe združovacích pravidiel (angl. *association rule mining*) je množina párov konceptov. Úlohou objavovania vzťahov podľa združovacích pravidiel je vytváranie vzťahov medzi konceptmi na príslušnej úrovni abstrakcie, napr. zovšeobecňovaním konceptov, aby pokrývali viacero dodaných párov. Úroveň abstrakcie je určená na základe vopred definovaných prahových hodnôt [40].

Práca [25] opisuje metódu pre objavovanie vzťahov pomocou združovacích pravidiel. Pre množinu konceptov sú z textu vygenerované združovacie pravidlá. Aplikovaním pravidiel na koncepty je zostavená tzv. sémantická matica, ktorá opisuje, medzi ktorými konceptmi existuje nejaký vzťah. Porovnaním množín konceptov, s ktorými majú dva koncepty vzťah, je získaná hodnota, podľa ktorej je učený typ vzťahu (žiadny vzťah, keď je prienik množín prázdny až vzťah rovnosti, keď sa množiny rovnajú). Pri overovaní metódy na testovacích dátach bola vyhodnotená presnosť vygenerovaných vzťahov, ktorá sa pohybovala okolo 85 %.

3.1.2 Lingvistické metódy

Tieto metódy sú závislé na spracovaní prirodzeného jazyka. Najčastejšie používanými lingvistickými metódami pre objavovanie vzťahov sú podľa [3, 40] metódy, ktoré využívajú syntaktické závislosti, lexikálno-syntaktické vzory alebo sémantické šablóny a sémantické slovníky.

Syntaktické závislosti

Tento prístup opisuje slová (termy) kontextovými vlastnosťami. Kontextové vlastnosti slova predstavujú vzťahy, v ktorých sa slovo nachádza. Sú pomenované podľa slovies, ktoré boli použité na ich odvodenie. Je generovaný zoznam syntaktických alebo pseudo-syntaktických závislostí podľa toho, či sú získavané tokenovo-úrovňovými (angl. *deep parsing*) alebo segmentovo-úrovňovými metódami (angl. *shallow parsing*). Tokenovo-úrovňové metódy analyzujú závislosti medzi prísudkom, podmetom, predmetom a pod. Segmentovo-úrovňové metódy získavajú závislosti pomocou

regulárnych výrazov nad množinou slov označených slovným druhom (angl. *POS (Part-of-speech) tag*). Po vygenerovaní vzťahov sa môžu slová vystupujúce vo vzťahu zatriediť do správnej úrovne abstrakcie stromu konceptov. Podobné slovesá môžu byť zoskupené do toho istého typu vzťahu [3, 40].

V [3] je opísaná metóda pre objavovanie vzťahov z korpusu odvodených zo slovies. Metóda využíva *slovesné rámce* (angl. *verb frames*). Z predspracovaných dát sa vyparsujú n-tice, ktoré vyhovujú výrazu NP-V-NP alebo NP-V-P-NP (NP – podstatnomenná fráza, V - sloveso, P - predložka). Vzťahy sú označené slovesom alebo slovesom s predložkou. Následne sú slová vo vzťahu zovšeobecnené podľa dostupnej ontológie na základe metriky tak, aby konečné koncepty zastrešovali všetky inštancie vzťahu nájdené v korpuse a zároveň boli čo najkonkrétnejšie, napr. z vety „*The virus leads to severe acute disease in macaques*." boli odvodené vzťahy *lead(subj:virus, to:disease, in: macaque)*, *lead_to(virus,other)*, *lead_in(virus,organism)*. Metóda bola overená porovnaním so zlatým štandardom navrhnutým odborníkom a ohodnotením pomocou metrík (počet presne určených vzťahov, priemerná vzdialenosť zle odhadnutých konceptov od správnych konceptov v ontológií). Najlepší dosiahnutý výsledok pomocou tejto metódy bol 33,53 % presne určených vzťahov.

V prácach [32, 7] sú využité syntaktické závislosti získané tokenovo-úrovňovými metódami. Každá veta je reprezentovaná grafom závislostí medzi vetnými členmi a vzťahy sú odvodzované na základe definovaných pravidiel pre jednotlivé typy vzťahov. Metóda bola overená aplikovaním na testovacie dáta a vyhodnotená metrikami úplnosť, presnosť a F-metrika. V práci [32] bola presnosť 65-100 % a úplnosť 57-100 %.

Reprezentáciu viet grafom závislostí medzi vetnými členmi, syntaktickým stromom, využíva aj metóda opísaná v práci [4]. Práca sa zameriava na jeden typ vzťahu – príčinný vzťah (angl. *causal relationship*), ktorý je opísaný vzorom a pravidlami, ktoré musia spĺňať identifikované jednotky vo vzore (trojica podmet – prísudok – predmet, kde prísudok naznačuje príčinný vzťah, napr. trojica *<heavy rain><caused><the flooding>*). Ak sa vzor nachádzal v syntaktickom strome vety, bol odvodený vzťah. Metóda bola overená aplikovaním na testovacie dáta a vyhodnotená metrikami úplnosť, presnosť a F-metrika. Presnosť tejto metódy bola 94,44 % a úplnosť 61,82 %.

Lexikálno-syntaktické vzory

Pri hľadaní vzťahov sa parsuje podľa ručne zostavených vzorov. Používajú sa pri hľadaní vzťahov nadradenosti (`is-a`) a vzťahov príslušnosti (`part-of`). Príkladom vzoru pre hľadanie vzťahov nadradenosti je *„NP such as NP"* alebo *„NP,..., and NP"* (NP – podstatnomenná fráza) [19]. Keďže manuálne písanie vzorov je náročné, zvyknú sa používať sémantické šablóny, ktoré poskytujú lepšie možnosti definície vzťahov a umožňujú hľadať aj komplexnejšie vzťahy. Napríklad šablóna z práce [34] pre hľadanie osôb, kde sa pred menom musí nachádzať titul a v strede mena môže byť iniciála (NNP – vlastné podstatné meno):

```
[syn=NNP, sem=PERSON] =>
[sem=title]{1,2}
\ [orth=capitalized],
[orth=upperinitial]?,
[orth=capitalized] / ;
```

Lexikálno-syntaktické vzory boli využité i v práci [35], kde boli objavované hierarchické vzťahy hľadaním vysvetľujúcich a určujúcich fráz vo výučbovom texte. Frázy boli definované súborom

pravidiel. Pravidlá sa zameriavali na výskyt konkrétnych slovies, ktoré naznačujú hierarchický vzťah (napr. byť, predstavovať, chápať). Najlepšia dosiahnutá hodnota F-metriky v tejto práci pri overení voči zlatému štandardu bola 55 %. Presnosť sa vtedy pohybovala okolo 75 % a úplnosť okolo 45 %.

Sémantické slovníky

Tento prístup využíva pri spracovaní sémantický slovník, akým je napríklad WordNet pre anglický jazyk, ktorý obsahuje rôzne vzťahy pre podstatné mená, prídavné mená, slovesá, príslovky, napr. synonymá, antonymá, podradenosť/nadradenosť, podobnosť [3]. WordNet je všeobecný, ale existujú aj doménovo zamerané slovníky, napr. UMLS (The Unified Medical Language System) [24].

Pre slovenský jazyk je dostupný Slovenský národný korpus z JÚĽŠ SAV[1], ktorý obsahuje gramatický opis slov (napr. slovný druh, číslo, pád, rod), ale neobsahuje žiadne sémantické vzťahy.

3.1.3 Logické metódy

Sú najmenej zvyčajné. Súvisia s metódami pre reprezentáciu informácií, dedukciu a strojové učenie. Vzťahy môžu byť odvodené robením logických záverov z už existujúcich vzťahov na základe princípov tranzitívnosti a dedičnosti [40].

3.1.4 Hybridné metódy

Hybridné metódy kombinujú predchádzajúce prístupy. Spájanie rôznych prístupov umožňuje využívať výhody jednotlivých metód a dosahovať tak lepšie výsledky.

V práci [30] je predstavená kombinácia štatistických a lingvistických metód. Autori uvádzajú ako výhody tejto kombinácie nezávislosť od jazyka a typu vzťahu, ktorú poskytujú štatistické metódy a schopnosť správne rozpoznať význam v prípade viacvýznamových slov, čo je možné pomocou lingvistických metód. Z textu sú najprv extrahované vety s podobným významom, tzv. parafrázy. Následne sú „zarovnané“ časti viet s rovnakým významom a zvyšok viet sa využíva na rozlíšenie významu slov v prípade viacerých možných významov slova. V ďalšom kroku sú vety lematizované a označené slovnými druhmi. V takto upravených vetách sa hľadajú slová, ktoré môžu byť medzi sebou zameniteľné a nezmení sa tak význam vety. Slová sú hľadané porovnávaním kontextu slov. Kontext je reprezentovaný vektorovým modelom. Podobnosť kontextov je určená pomocou metrík kosínusová vzdialenosť, bodová vzájomná informácia a metrikami založenými na podmienenej pravdepodobnosti. Metóda je overená aplikovaním na testovacie dáta a vyhodnotením presnosti pre rôzne použité metriky na určenie podobnosti kontextov. Najlepší výsledok pri hľadaní synoným mal presnosť 75 %.

3.1.5 Spôsoby overenia

Metódy pre objavovanie vzťahov sú overované porovnaním s tzv. „zlatým“ štandardom, ktorý predstavuje zvyčajne zdroj dát manuálne vytvorený doménovými odborníkmi (ontológia, slovník a pod.). Podobnosť so „zlatým“ štandardom je vyhodnotená rôznymi metrikami.

[1] http://korpus.juls.savba.sk/

Najčastejšie využívanými sú [3]:

- úplnosť (angl. *recall*), čo je pomer správne objavených znalostí (konceptov, vzťahov, a pod.) ku všetkým existujúcim znalostiam nachádzajúcich sa v „zlatom“ štandarde,
- presnosť (angl. *precision*), čo je podiel správne objavených znalostí zo všetkých znalostí objavených metódou,
- F-metrika, ktorá predstavuje harmonický priemer predchádzajúcich dvoch.

Úplnosť a presnosť sú väčšinou nepriamo úmerné, t.j. metódy, ktoré objavia veľa vzťahov sú menej presné a metódy, ktoré vykazujú vysokú presnosť pokrývajú iba malú množinu vzťahov, lebo sa napr. špecializujú iba na jeden prípad výskytu vzťahu medzi konceptmi.

V niektorých prácach sa vyskytujú aj variácie týchto metrík, napr. v práci [36] bola úplnosť modifikovaná tak, aby porovnávala objavené vzťahy len s podmnožinou vzťahov zo „zlatého“ štandardu, s tzv. povinnými vzťahmi. V práci [6] sa s absenciou „zlatého“ štandardu pre nájdené typy vzťahov vyrovnali zavedením metriky pokrytie (angl. *coverage*) namiesto metriky úplnosť. Táto metrika vyjadrovala podiel konceptov, pre ktoré bol nájdený aspoň jeden vzťah s relevantným – príbuzným konceptom. Prípadne sú vytvorené metriky špeciálne pre vyhodnotenie konkrétnych aspektov metódy pre objavovanie vzťahov, napr. v práci [3] to boli počet presne určených vzťahov v ontológii a priemerná vzdialenosť zle odhadnutých konceptov od správnych konceptov v ontológií meraná počtom hrán alebo uzlov medzi zle odhadnutým a správnym konceptom (angl. *learning accuracy*).

3.1.6 Zhrnutie

Hlavná výhoda štatistických metód je, že sú nezávislé od jazyka a pri ich aplikácií nie sú potrebné ani hlbšie lingvistické znalosti. Nevýhodou je, že ak chceme, aby metódy boli dostatočne efektívne, je potrebné veľké množstvo dokumentov. Napriek tomu sa štatistické metódy ukázali vo všeobecnosti efektívnejšie ako iné prístupy.

Nevýhodou lingvistických metód je, že sú závislé na jazyku, v ktorom sú napísané dokumenty. S tým súvisí aj nutnosť mať aspoň základné lingvistické znalosti o jazyku a jeho syntaktických konštrukciách. Takisto vytvorenie správnej množiny vzorov alebo pravidiel pre objavenie vzťahov je časovo náročné. Jeden typ vzťahu môže byť reprezentovaný v texte viacerými spôsobmi a je ťažké navrhnúť vzory a pravidlá pre jeho objavovanie tak, aby pokryli všetky prípady. Výhodou môže byť pomerne jednoduchá implementácia v podobe konečného automatu a rýchle spracovanie textu. Podľa [3] sú to najčastejšie používané metódy pre objavovanie vzťahov. Hlavne metódy využívajúce syntaktické závislosti, konkrétne slovesné rámce a metódy, v ktorých sa odvádzajú vzťahy podľa združovacích pravidiel (angl. *association rule mining*).

Logické metódy sú najmenej používané a pre odvádzanie vzťahov je potrebné, aby už predtým existovali vzťahy alebo informácie, na základe ktorých je možné robiť logické závery. Logické metódy sú aplikovateľné na „ťažké“ ontológie, ktoré obsahujú aj axiómy.

Zvyčajne používané metriky, ktorými sa hodnotí úspešnosť metódy pre objavovanie vzťahov sú presnosť (angl. *precision*), úplnosť (angl. *recall*) a F-metrika, ktorá predstavuje harmonický priemer predchádzajúcich dvoch. Z hľadiska týchto metrík v spomínaných prácach sa javia ako úspešnejšie prístupy využívajúce lingvistické metódy. V priemere sú pri lingvistických metódach v analyzovaných

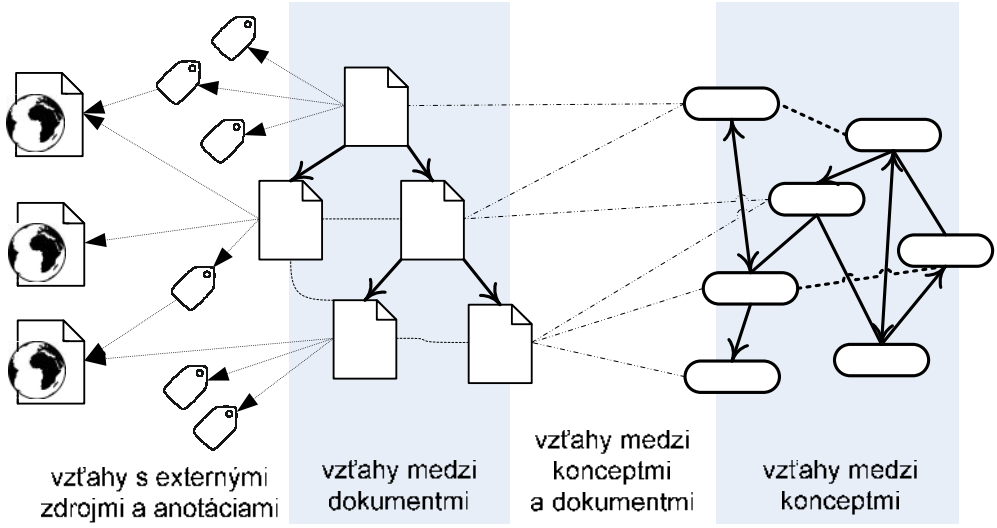
prácach uvádzané o 5-10 % lepšie hodnoty metrík presnosť a úplnosť. Môže to byť spôsobené tým, že lingvistické metódy sa zameriavajú na objavovanie konkrétnych vzorov a pravidiel, a preto vykazujú lepšiu presnosť. Musíme brať do úvahy aj to, že metodika overenia a dáta, na ktorých prebiehalo overenie sú v každej práci iné. Preto ich číselné porovnávanie nie je významné.

Rôznymi metódami sa dajú objavovať rôzne typy vzťahov. Pre objavovanie vzťahov podobnosti a príbuznosti pojmov sú najvhodnejšie metódy založené na porovnávaní kontextu slov, napr. LSA, metódy využívajúce zhlukovanie, napr. LDA, združovacie pravidlá (napr. v práci [25]), syntaktické závislosti, lexikálno-syntaktické vzory a sémantické slovníky (napr. WordNet obsahuje množiny synoným, tzv. *synset*). Tieto metódy totiž priamo vyhodnocujú podobnosť konceptov rôznymi metrikami alebo v prípade lingvistických metód hľadajú takéto vzťahy podľa špeciálne navrhnutých vzorov.

Pre objavovanie hierarchických vzťahov je možné využiť formálnu analýzu konceptov, začleňovanie pojmov, lexikálno-syntaktické vzory a syntaktické závislosti.

3.2 Spracovanie štruktúry obsahu

Výučbový obsah predstavujú dokumenty a doménový model. Samotné dokumenty sú hierarchicky zoradené na kurzy, kapitoly, sekcie a pod. Medzi jednotlivými dokumentmi môžu existovať aj iné prepojenia, napr. podľa podobnosti ich obsahu. Každý dokument je prepojený na doménový model (obsahuje alebo má priradené relevantné doménové termy). Navyše môžu byť časti jednotlivých dokumentov anotované alebo prepojené na externé zdroje. Takúto štruktúru (Obr. 4) možno prezentovať ako graf, a preto je možné pre objavovanie vzťahov využiť aj grafové algoritmy. Externé zdroje (vľavo) môžu byť ešte cyklicky prepojené s konceptmi doménového modelu, lebo ich môžu opisovať.



Obr. 4. Grafová štruktúra výučbového obsahu.

Príkladmi grafových algoritmov použiteľných pri objavovaní vzťahov sú *PageRank* algoritmus použitý v práci [36], algoritmus *šírenia aktivácie* použitý v prácach [28, 36] a *Semantic GrowBag* algoritmus predstavený v práci [8].

3.2.1 PageRank algoritmus

Tento algoritmus používa väčšina internetových vyhľadávačov pri zoraďovaní stránok, ktoré sú odpoveďou na dopyt používateľa. Spočíva v ohodnotení stránok koeficientom, na základe ktorého sú potom ponúkané pri vyhľadávaní. Na tento koeficient vplýva hlavne počet stránok odkazujúcich na hodnotenú stránku. PageRank bol predstavený v práci [1].

PageRank hodnotenie predstavuje pravdepodobnostné rozdelenie, čiže suma hodnotení všetkých stránok je rovná 1. Vyjadruje pravdepodobnosť, že používateľ sa náhodným klikaním na linky prekliká až na hodnotenú stránku (tzv. model náhodného surfera). Pravdepodobnosť je vyjadrená nasledujúcim vzťahom:

$$PageRank(p_i) = \frac{1-d}{N} + d \sum_{p_j \in I(p_i)} \frac{PageRank(p_j)}{O(p_j)} \tag{1}$$

kde hodnotenie (alebo „PageRank“) stránky p_i sa rovná sume podielov hodnotení ($PageRank(p_j)$) a počtov odkazov na iné stránky ($O(p_j)$) všetkých stránok, ktoré odkazujú na stránku ($I(p_i)$). Vzorec zahŕňa aj d, tzv. brzdiaci faktor, ktorý predstavuje pravdepodobnosť, že náhodný surfer prestane prezerať ďalšie stránky a začne novou náhodnou stránkou. N predstavuje celkový počet stránok. Prvý sčítanec vo vzorci sa nazýva aj *zdroj hodnotenia*.

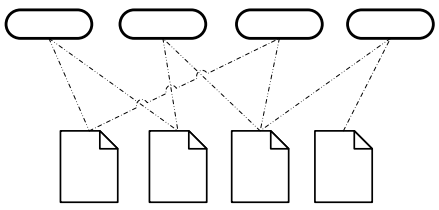
Z takto definovaného vzorca vyplýva, že vysoké hodnotenie stránky nezávisí len od počtu stránok, ktoré odkazujú na hodnotenú stránku ale aj od hodnotení odkazujúcich stránok. T.j. vysoké hodnotenie sa dá dosiahnuť aj s málo odkazmi od vysoko hodnotených stránok.

Aplikovaním na graf môžeme hodnotiť dôležitosť uzlov. Rozšírením tohto algoritmu je *PageRank algoritmus s prioritami* [43], kedy sa počíta koeficient iba vzhľadom na definovanú množinu koreňových uzlov a nie celý graf. Hodnotenie je v tomto pozmenenom algoritme vyjadrené vzťahom:

$$PageRank'(p_i) = (1-\beta)\left(\sum_{p_j \in I(p_i)} PageRank'(p_j) PB(p_i \mid p_j) \right) + \beta PB(p_i) \tag{2}$$

kde β je pravdepodobnosť, ako často sa náhodný surfer vráti na koreňové stránky a $PB(p_i)$ vyjadruje relatívnu dôležitosť stránky (angl. *prior bias*), je to apriórna pravdepodobnosť (angl. *prior probability*). Suma týchto pravdepodobností pre všetky stránky je rovná 1.

V prípade objavovania vzťahov uzly grafu predstavujú koncepty a algoritmom možno zistiť relevantné koncepty. Rozšírenie algoritmu PageRank bolo úspešne aplikované v práci [36] pri objavovaní vzťahov medzi konceptmi prepojenými s dokumentmi (Obr. 5). Ako koreňový uzol bol označený jediný koncept, ktorého „prior bias“ bola 1 a výsledkom boli relatívne dôležitosti konceptov voči tomuto konceptu. Presnosť tejto metódy bola 56,9 % a jej úplnosť bola 50 %.



Obr. 5. Graf dokumentov s priradenými konceptmi [36].

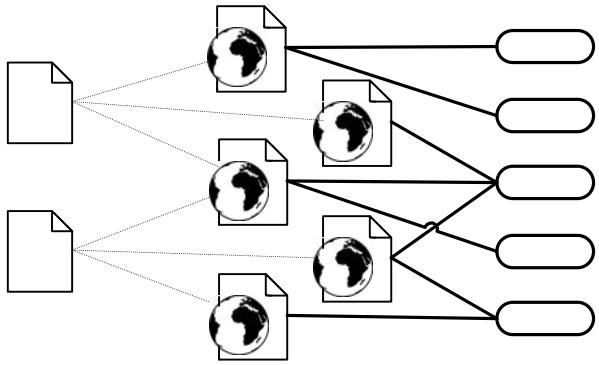
3.2.2 Šírenie aktivácie

Šírenie aktivácie sa využíva na vyhľadávanie v sieti konceptov. Bolo prezentované v práci [2] a použité na vyhľadávanie napr. v práci [39]. Je ho možné aplikovať aj na hľadanie podobných pojmov. Počiatočným uzlom v sieti je priradená aktivačná energia, ktorá sa šíri sieťou podľa nasledujúceho algoritmu:

```
procedure energize(energy E, node nk) {
      energy(nk) := energy(nk) + E
      E' := E / degree of nk
      if(E' > T) {
            for each node nj in Nk {
                  E" := E' * ejk
                  energize(E", nj)
            }
      }
}
```

kde N_k je množina susedov uzla n_k, e_k je váha hrany spájajúca uzly n_k a n_j, T je konštantná prahová hodnota a *energy*(n_k) je dátová štruktúra, ktorá uchováva hodnoty energií uzlov.

Pri aplikácii algoritmu šírenia aktivácie uzly siete predstavujú koncepty a dokumenty, ku ktorým sa koncepty viažu (Obr. 5). Na túto sieť je nasledovne aplikovaný algoritmus rekurzívneho šírenia energie z počiatočného uzla do celej siete. Výsledkom sú objavené podobné koncepty. Podobnosť je vyjadrená ako pomer energie v koncových uzloch ku energii počiatočného uzla. Tento algoritmus bol úspešne aplikovaný pri objavovaní vzťahov medzi konceptmi v práci [36]. Dosiahnutá presnosť bola 44,3 % a úplnosť 54,4 %. V práci [28] boli pomocou algoritmu objavované aj vzťahy medzi dokumentmi pomocou prepojení na externé zdroje (Obr. 6).



Obr. 6. Grafová reprezentácia dokumentov prepojených na externé zdroje a koncepty [28].

3.2.3 Semantic GrowBag algoritmus

Semantic GrowBag algoritmus [8] je schopný vytvoriť z množiny dokumentov s priradenými relevantnými doménovými termami kategorizačný systém, t.j. hierarchiu pojmov. Využíva sa v ňom rozšírený PageRank algoritmus s prioritami a teória začleňovania pojmov. Výhodou tohto algoritmu oproti klasickému začleňovaniu pojmov je, že nájde aj tzv. „skryté“ vzťahy medzi pojmami. Sú to vzťahy medzi pojmami, ktoré nie sú spolu priradené k jednému dokumentu, ale majú veľa spoločných pojmov, s ktorými áno. Semantic Growbag algoritmus teda dokáže nájsť omnoho viac vzťahov a dokáže klasifikovať silu vzťahu (slabý, silný).

Algoritmus sa skladá z nasledujúcich krokov:

1. Vypočítať spoločný výskyt pojmov berúc do úvahy aj „skryté“ vzťahy.

   a) Zostroj maticu M (*n x n*) pre *n* pojmov, ktorá obsahuje ohodnotené vzťahy medzi pojmami na základe ich spoločného výskytu v dokumentoch.

      Prvky matice sú definované nasledovne:

$$m(j,i) = \frac{cooc(i,j) * ICF(i)}{\sum_j cooc(i,j) * ICF(i)} \tag{3}$$

      kde *cooc(i,j)* vyjadruje spoločný výskyt pojmov *i* a *j* (počet dokumentov, v ktorých sa obidva nachádzajú), *ICF(i)* je inverzná frekvencia spoločného výskytu definovaná ako:

$$ICF(i) = \log\left(\frac{n}{numOfCoocTerms(i)}\right) \tag{4}$$

      kde *n* je celkový počet pojmov a *numOfCoocTerms(i)* je počet pojmov, ktoré sa vyskytujú v dokumentoch spolu s pojmom *i*.

   b) Pre každý pojem urči najčastejšie sa spolu vyskytujúce pojmy (*priamych susedov*).

      Zoraď prvky vektora z matice M reprezentujúce váhy so spoločne sa vyskytujúcimi pojmami pre pojem *i* a vyber len prvých *P* %, kde *P* je zvyčajne z intervalu 10-30 %.

   c) Pre každý pojem vypočítaj PageRank (zoznam pojmov s hodnotami PageRank predstavujúci odpoveď, ak by sme sa dopytovali na pojem) pomocou matice M použitím rozšíreného PageRank algoritmu s prioritami vzhľadom na jeho *priamych susedov*.

2. Objaviť vzťahy medzi pojmami na základe spoločného výskytu pojmov.

   a) Zoraď a osekaj PageRank pojmu tak, aby v ňom ostali len pojmy s najväčším hodnotením.

   b) Ak sa v oboch PageRank-och pre dvojicu pojmov *i* a *j* nachádza pojem *i* alebo pojem *j* vždy s vyšším hodnotením, potom označ tento pojem ako kandidát na začleňovanie pojmu s nižším hodnotením (je všeobecnejší).

   c) Na základe hodnotenia pojmov klasifikuj silu vzťahu. Ak sa pojem s nižším hodnotením nachádza v *priamych susedoch* pojmu s vyšším hodnotením, tak je vzťah slabý. Ak sa aj pojem s vyšším hodnotením nachádza v *priamych susedoch* pojmu s nižším hodnotením, vzťah je silný. V opačnom prípade je kandidát zahodený. Výsledok je zoznam trojíc *(pojem i, pojem j, sila vzťahu)*. Trojica označuje, že *pojem i* začleňuje *pojem j* s určitou silou (slabý/silný vzťah).

3. Zostrojiť pre každý pojem *GrowBag* graf zobrazujúci vzťahy s priamymi susedmi pojmu.

   a) Vytvor uzly z pojmov, ktoré začleňujú pojmy, s ktorými má pojem „skrytý“ vzťah.

   b) Rekurzívne vytvor uzly z pojmov, ktoré sú začlenené už pridaným pojmami.

   c) Vytvor hrany medzi uzlami, graficky rozlíš silu vzťahu.

3.2.4 Zhrnutie

Štruktúra výučbového obsahu môže byť reprezentovaná ako graf, a preto je možné použiť na objavovanie vzťahov medzi konceptmi aj grafové algoritmy. V analyzovaných prácach boli najčastejšie využívanými PageRank algoritmus a algoritmus šírenia aktivácie, ktoré sú štandardne používané na vyhľadávanie v sieťach. Ich zakomponovanie do procesu objavovania vzťahov môže spočívať v jednoduchom aplikovaní algoritmu na štruktúru výučbového obsahu ako to bolo v prácach [28, 36] alebo môžu byť súčasťou komplexnejšieho algoritmu ako je to v prípade Semantic Growbag algoritmus [8].

Práca [36] použila obidva tieto algoritmy. Väčšia presnosť pri objavovaní vzťahov medzi konceptmi bola dosiahnutá analýzou pomocou PageRank algoritmu s prioritami, ale väčšia úplnosť pomocou algoritmu šírenia aktivácie. V tejto práci bolo ukázané, že grafový algoritmus dokázal vylepšiť aj presnosť metódy pre objavovanie vzťahov založenej na štatistickom spracovaní.

Základom Semantic GrowBag algoritmu je takisto metóda pre objavovanie vzťahov založená na spracovaní obsahu dokumentov, konkrétne kolokácie pojmov v dokumentoch. Následne je na tieto prvotné objavené vzťahy aplikovaný PageRank algoritmus, ktorým sú zostrojené hierarchické vzťahy.

Kombináciou grafových algoritmov aplikovaných na štruktúru konceptov naviazaných na výučbové dokumenty s metódou pre objavovanie vzťahov z obsahu výučbových dokumentov môžu byť teda dosiahnuté lepšie výsledky.

4 Ciele práce

Cieľom tejto práce je navrhnúť metódu pre objavovanie vzťahov vo výučbovom obsahu adaptívneho systému. Výučbový obsah v našom prípade budú predstavovať textové dokumenty hierarchicky zoradené do kapitol. Našou úlohou bude zostaviť „ľahkú“ ontológiu z tohto obsahu, ktorá bude tvoriť doménový model. Ten sa bude skladať z relevantných doménových termov a vzťahov medzi nimi. Vzniknutý doménový model bude využívaný pri odporúčaní obsahu používateľom adaptívneho systému v procese učenia. Doménový model slúži aj na modelovanie používateľa. Budeme teda hľadať vzťahy medzi pojmami – relevantnými doménovými termami, ktoré budú buď extrahované z obsahu alebo priradené dokumentom učiteľom. Jadro doménového modelu je tvorené hierarchickými vzťahmi. Preto ako vhodný typ vzťahov, ktoré by mala naša metóda objaviť, sa javia hierarchické vzťahy. Vzhľadom na využitie doménového modelu v procese odporúčania sa ako druhý vhodný typ vzťahov na objavovanie javia vzťahy podobnosti a súvislosti.

Za cieľ si kladieme podporiť tvorbu adaptívnych kurzov a pomôcť automatizovať tento proces. Metóda pre objavovanie vzťahov bude integrovaná do systému pre správu výučbového obsahu, ktorý už podporuje manuálnu tvorbu doménového modelu. Manuálne vytvorenie správneho a úplného doménového modelu vyžaduje od učiteľa vynaloženie veľkého úsilia. Preto je preňho výhodnejšie mať možnosť automaticky vygenerovať aspoň čiastočný doménový model a následne ho upraviť do finálnej podoby.

Objavovať vzťahy pomocou niektorej z lingvistických metód je zložité. Návrh vhodných vzorov a pravidiel, podľa ktorých by malo prebiehať objavovanie, by vyžadoval množstvo úsilia a zabral veľké množstvo času a výsledok by nemusel pokrývať všetky prípady. Pri navrhovaní vzorov a pravidiel by bolo potrebné dopredu naštudovať prehľadávaný obsah a prispôsobiť mu vzory a pravidlá, čo nebude možné, keďže naša metóda má byť implementovaná v systéme pre správu obsahu a mala by byť aplikovateľná všeobecne na akýkoľvek spravovaný výučbový obsah. Navyše podobné práce využívajúce lingvistickú stránku textu už existujú [35].

Preto by sme chceli navrhnúť a overiť metódu založenú na štatistických metódach, akými sú napr. LSA, LDA alebo v prípade hierarchických vzťahov FCA a začleňovanie pojmov. Pri objavovaní vzťahov podobnosti a súvislosti považujeme za vhodnú vektorovú reprezentáciu pojmov, resp. ich kontextu, pretože je ľahko spracovateľná. Vylepšenia výsledkov metódy môžu byť dosiahnuté použitím jedného z grafových algoritmov na štruktúru výučbového obsahu podobne ako v predchádzajúcich prácach [28, 36]. Pri overovaní nás okrem vyhodnotenia úspešnosti metódy zaujíma aj, či sa štatistický prístup dokáže vyrovnať „na mieru šitému“ lingvistickému prístupu a či sa štatistickým prístupom dajú získať nové vzťahy neobjaviteľné lingvistickými metódami. V rámci overenia integrujeme metódu do systému pre správu výučbového obsahu.

5 Metóda pre objavovanie vzťahov

Navrhovaná metóda pre objavovanie vzťahov z výučbové obsahu vychádza zo štatistických prístupov k objavovaniu vzťahov. Zameriava sa na objavovanie *vzťahov medzi konceptmi*. Vstupom metódy bude *množina dokumentov*, pre ktorú sa bude vytvárať doménový model. Výstupom bude *„ľahká" ontológia*, ktorá bude tvorená množinou pojmov (termov) poprepájaných vzťahmi. Termy budú priradené dokumentom.

Metóda je zameraná na objavovanie *hierarchických vzťahov*. Na vrchu hierarchie sa budú vyskytovať všeobecnejšie termy, ktoré budú pokrývať širšie oblasti (napr. term „ontológia"), na nižších úrovniach sa budú nachádzať špecifickejšie termy, ktoré spadajú pod všeobecnejší term (napr. term „OWL"). Vytváranie takejto hierarchie je založené na začleňovaní pojmov (pozri 3.1.1).

Navyše budú objavené *vzťahy príbuznosti* medzi pojmami. Budú to tie, ktoré nebudú klasifikované ako hierarchické, ale budú dostatočne relevantné na ich uchovanie v ontológii, pretože takéto vzťahy môžu byť takisto užitočné pri odporúčaní výučbového obsahu.

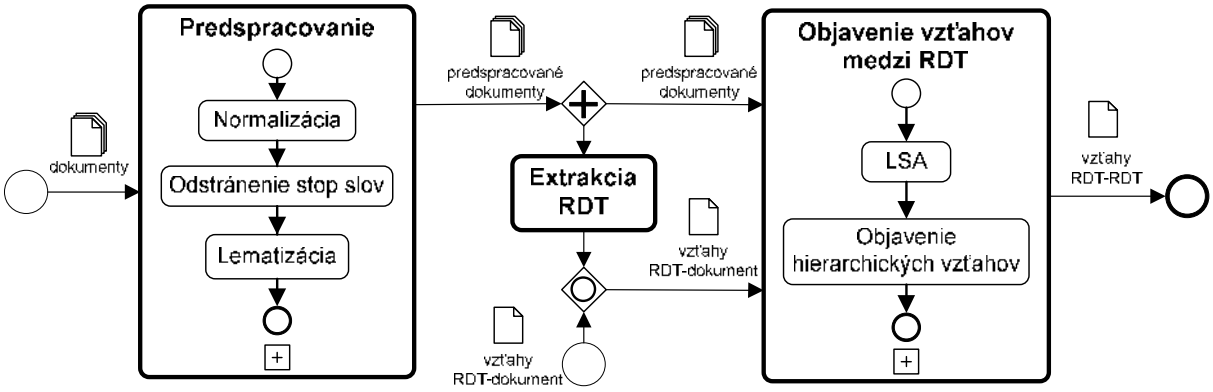
Pri návrhu metódy sme brali do úvahy fakt, že metóda bude slúžiť na objavovanie vzťahov vo výučbovom obsahu. To znamená, obsahu s jednoznačnou slovnou zásobou, čiže vyšším výskytom kľúčových pojmov v jednej forme na rozdiel od napríklad beletrie, ktorá používa širšiu škálu slov s mnohými synonymami. Je to predpoklad pre úspech štatistického prístupu.

Proces objavovania vzťahov je zobrazený na obrázku (Obr. 7). Skladá sa z fáz (a) predspracovanie dokumentov, (b) extrakcia relevantných doménových termov a ich vzťahov s dokumentmi, (c) odvodenie vzťahov medzi termami.

Vstupom bude *množina dokumentov*, pre ktorý bude používateľ chcieť vygenerovať doménový model. Výstupom bude vygenerovaný *doménový model*, ktorý tvorí:

1) *súbor relevantných doménových termov* (RDT), predstavujúci kľúčové pojmy z dokumentov,
2) *súbor vzťahov medzi termami*, definovaných váhou sémantickej súvislosti termov a typom.

Doménový model je *naviazaný na dokumenty* prostredníctvom *vzťahov medzi termami a dokumentami*. Každý vzťah definovaný váhou podľa dôležitosti termu v dokumente.



Obr. 7. Proces objavovania vzťahov vo výučbovom obsahu.

5.1 Predspracovanie dokumentov

Predspracovanie je jediná jazykovo závislá časť našej metódy. Vo fáze predspracovania budú dokumenty upravené do podoby vhodnej na ďalšie spracovanie. V prípade slovenského jazyka text prejde najprv normalizáciou (odstránenie interpunkcie a špeciálnych znakov). Ďalším krokom predspracovania bude odstránenie tzv. stop slov z textu (slová, ktoré nevplývajú na význam, napr. predložky, spojky). Na zvyšné slová bude nakoniec aplikovaná lematizácia (uvedenie slov do základného tvaru).

5.2 Extrakcia RDT a vzťahov medzi termami a dokumentmi

Ako je naznačené v procese (Obr. 7), tento krok je voliteľný. Pri tvorbe výučbového obsahu môžu byť termy priraďované k dokumentom autorom obsahu. Teda aj v našej metóde je možné využiť už existujúce dáta – súbor relevantných doménových termov a ich vzťahov s dokumentmi.

Tento krok je však možné aj automatizovať, pretože termy sa dajú extrahovať z textov dokumentov. V tomto prípade môžeme pri ich odvodzovaní využiť metriku *tf-idf* (*term frequency – inversed term frequency*), ktorá sa štandardne využíva na tento účel. Termy budú reprezentované jednoslovnými podstatnými menami z textu. Táto metrika vyjadruje, aký dôležitý je term pre dokument z korpusu – súboru dokumentov. Všetky podstatné mená z dokumentov sú ňou ohodnotené a tie, ktorých frekvencia je väčšia ako definovaná prahová hodnota, sú označené ako relevantné doménové termy. Hodnota *tf-idf* zároveň predstavuje váhu vzťahu medzi termom a dokumentom, z ktorého bol term odvodený. Tento prístup bol už úspešne aplikovaný v práci [36].

5.3 Odvodenie vzťahov medzi termami

Odvodenie vzťahov bude prebiehať v dvoch etapách:

- zostavenie siete termov,
- objavenie hierarchických vzťahov.

5.3.1 Zostavenie siete termov

V tejto etape vznikne sieť z definovaných relevantných doménových termov. Vzťahy medzi termami budú mať určitú váhu, ale nebude známy ich typ. Sieť bude zostavená pomocou latentnej sémantickej analýzy (LSA). LSA umožní nájsť vzťahy aj medzi termami, ktoré sa spolu priamo nenachádzajú v dokumentoch, ale vyskytujú sa v korpuse v spoločnom kontexte, t.j. nejako spolu súvisia.

Pre každý term bude vybraný jeho kontext – okolie termu v dokumente definované počtom slov. Následne bude zostavená matica, ktorej stĺpce budú predstavovať kontexty a riadky slová, ktoré sa nachádzajú v kontextoch. Prvky matice budú predstavovať počet výskytov slova v kontexte. Táto matica bude transformovaná na hodnoty *tf-idf*. Potom bude na maticu bude aplikovaný singulárny rozklad matice (angl. *SVD – Singular Vector Deomposition*). Stĺpce vzniknutej matice budú navzájom porovnané a ich podobnosť bude ohodnotená metrikou kosínusová vzdialenosť. Na základe porovnania budú vygenerované vzťahy medzi termami s váhou hodnoty kosínusovej vzdialenosti, ktorá sa bude pohybovať v intervale <0,1>. Vybrané budú len vzťahy s váhou, ktorá je väčšia ako definovaná prahová hodnota.

5.3.2 Objavenie hierarchických vzťahov

Pomocou LSA sme získali vzťahy ohodnotené váhami, ale nie typy vzťahov. Na základe podobnosti kontextov termov vieme len odvodiť, že termy spolu nejako súvisia. Nie však, či je jeden term nadradený druhému alebo sú len príbuzné. Preto sme sa rozhodli, že pri určovaní typu vzťahu využijeme tzv. začleňovanie pojmov (angl. *term subsumption*).

Navrhujeme dva varianty ako objaviť hierarchie pojmov zo siete:

- *množinový variant*, ktorý využíva teóriu začleňovania pojmov,
- *PageRank variant*, v ktorom aplikujeme PageRank algoritmus na štruktúru výučbového obsahu (termy naviazané na sieť dokumentov).

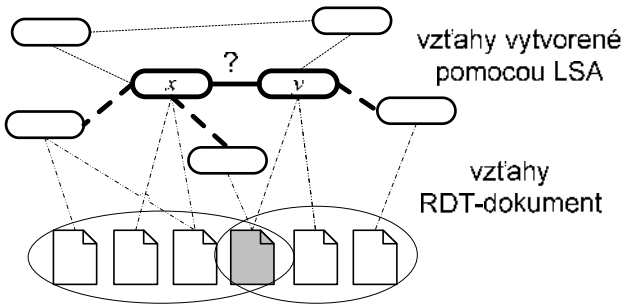
Množinový variant objavovania hierarchických vzťahov

Vzťahy získané pomocou LSA, ktorých váha sa blíži k 1, existujú medzi veľmi súvisiacimi termami, ktoré môžu byť medzi sebou v texte nahraditeľné, resp. jeden môže byť špecializáciou druhého. Na takéto dvojice termov aplikujeme podmienku, ktorá definuje, či je jeden term všeobecnejší ako druhý. Najprv pre každý term z dvojice identifikujeme množinu dokumentov, ku ktorým je priradený. Do množiny dokumentov pridáme aj dokumenty, ku ktorým sú priradený veľmi súvisiaci susedia termu zo siete termov. Podmienka je potom definovaná takto:

Term *x* je všeobecnejší ako term *y*, ak:

- prienik množín dokumentov je nenulový,
- množina dokumentov termu *x* má väčšiu kardinalitu ako množina dokumentov termu *y* (je ich viac, teda viac frekventovanejší term bude označený za všeobecnejší – term *x*).

Veľmi súvisiacich susedov definujeme ako termy, s ktorými má term tiež vzťah s váhou blížiacou sa k 1 (Obr. 8).



Obr. 8. Príklad určovania hierarchického vzťahu množinovým variantom metódy.

Formálne môže byť táto podmienka vyjadrená ako pravdepodobnosť, že dokument z prieniku množín je priradený k termu *x* je väčšia ako pravdepodobnosť, že je priradený k termu *y*:

$$P(O_x \mid O_y) > P(O_y \mid O_x) \tag{5}$$

kde O_x predstavuje množinu dokumentov, v ktorých sa nachádza term *x* alebo jeho susedia zo siete termov (okolie termu *x*), O_y predstavuje obdobnú množinu pre term *y* (okolie termu *y*). Podmienená pravdepodobnosť $P(O_x|O_y)$ je definovaná ako podiel kardinality prieniku množín (počtu spoločných dokumentov) a kardinality množiny O_y (počtu dokumentov, v ktorých sa nachádza term *y* alebo jeho susedia zo siete termov).

Formálne zapísaná ako:

$$P(O_x \mid O_y) = \frac{|O_x \cap O_y|}{|O_y|} \tag{6}$$

V originálnej práci o začleňovaní pojmov [33] existovala podmienka, že termy sa museli vyskytovať spolu v dokumente s pravdepodobnosťou aspoň 80 %. Túto časť sme sa ale rozhodli vynechať, pretože vzťah objavený pomocou LSA môže existovať aj medzi termami, ktoré sa nevyskytujú spolu v jednom dokumente. Radšej sme ju nahradili prvou časťou podmienky o prieniku množín dokumentov priradených k termu a jeho veľmi súvisiacim susedom. V prípade, že dvojica termov vyhovie tejto podmienke, vytvorí sa medzi nimi vzťah nadradenosti/podradenosti. V opačnom prípade vzťah príbuznosti.

Postup určovania hierarchických vzťahov môže byť zapísaný pseudoalgoritmom (Algoritmus 1). Vstupom algoritmu sú vzťahy medzi termami vygenerované pomocou LSA (*RDT-RDTRels*), vzťahy medzi termami a dokumentmi (*RDT-DOCRels*), súbor relevantných doménových termov (*RDTs*) a váha *w*, ktorou definujeme minimálnu váhu, ktorú musí mať vzťah podobnosti, aby bol kandidát na hierarchický vzťah. Je to aj minimálna váha vzťahu s veľmi súvisiacimi susedmi termu. V algoritme sme doplnili podmienku o prípad, keď množiny dokumentov majú neprázdny prienik, ale rovnakú kardinalitu. V tomto prípade je nadradený term ten, ktorý sa častejšie vyskytuje v texte. Výstupom algoritmu je množina hierarchických vzťahov (*hierarchicalRels*) identifikovaných zo vstupných vzťahov vygenerovaných pomocou LSA.

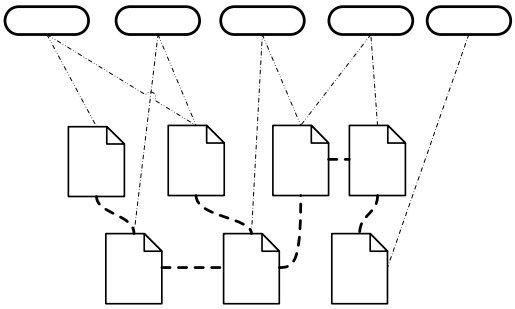
Algoritmus 1. Pseudoalgoritmus množinového variantu objavovania hierarchických vzťahov.

```
1:  procedure FindHierarchical(RDT-RDTRels, RDT-DOCRels, RDTs, w)
2:      hierarchicalRels ← [ ]
3:      RDTNeighbours ← GetNeighbors(RDT-RDTRels)

4:      for all relationship ∈ RDT-RDTRels do
5:          if relationship[weight] > w then
6:              fromRDT ← relationship[from]
7:              fromSet ← GetDOCSet(fromRDT, RDTNeighbors[fromRDT], RDT-DOCRels)

8:              toRDT ← relationship[to]
9:              toSet ← GetDOCSet(toRDT, RDTNeighbors[toRDT], RDT-DOCRels)

10:             if fromSet ∩ toSet ≠ ∅ then
11:                 if |fromSet| > |toSet| then
12:                     hierarchicalRels ← [toRDT, fromRDT, relationship[weight]]

13:                 else if |toSet| > |fromSet| then
14:                     hierarchicalRels ← [fromRDT, toRDT, relationship[weight]]

15:                 else if |fromSet| == |toSet| then
16:                     if TextOccurrences(fromRDT) > TextOccurrences(toRDT) then
17:                         hierarchicalRels ← [toRDT, fromRDT, relationship[weight]]
18:                     else if TextOccurrences(toRDT) > TextOccurrences(fromRDT) then
19:                         hierarchicalRels ← [fromRDT, toRDT, relationship[weight]]
20:                     end if
21:                 end if
22:             end if
23:         end if
24:     end for
25:     return hierarchicalRels
26: end procedure
```

Naša metóda je založená na začleňovaní pojmov predstavenom v práci [33]. Existujúci prístup sme však modifikovali. V pôvodnej práci autori odvodili relevantné doménové termy z textu a hierarchicky ich začleňovali na základe podmienenej pravdepodobnosti ich spoločného výskytu v texte. Termy sa museli spolu vyskytovať v dokumentoch s pravdepodobnosťou väčšou ako 80 %, potom častejšie sa vyskytujúci term bol označený za všeobecnejší. Naša metóda nevyžaduje spoločný výskyt dokumentov v texte. Vzťahy medzi termami sú nájdené latentnou sémantickou analýzou. Potom neporovnávame počet výskytov termov v dokumentoch, ale porovnávame množiny dokumentov, ku ktorým bol term priradený na základe existujúcich vzťahov medzi termami a dokumentmi. Tieto vzťahy môžu byť manuálne vytvorené autorom výučbového obsahu, preto nezaručujú, že term sa vyskytuje aj v texte dokumentu. Autor obsahu môže vytvoriť aj vzťahy medzi termami a dokumentmi, ktoré by nebolo možné automaticky objaviť.

PageRank variant objavovania hierarchických vzťahov

V tomto spôsobe navrhujeme použiť rozšírený PageRank algoritmus s prioritami, ktorý aplikujeme na grafovú štruktúru zloženú z termov napojených na sieť dokumentov (Obr. 9). Predpokladáme, že vzťahy medzi dokumentmi už existujú. V prípade, že nie, je ich ľahké odvodiť napr. aplikáciou algoritmu šírenia aktivácie na graf dokumentov s priradenými termami (Obr. 5).



Obr. 9. Grafová štruktúra - termy napojené na sieť dokumentov.

Pre každý term aplikujeme na graf PageRank algoritmus s prioritami, pričom koreňové uzly budú všetky dokumenty, ku ktorým je term priradený. Pre každý term bude takto vytvorený zoznam termov zoradený podľa PageRank hodnotenia. V zozname bude ponechaných len prvých *k%* termov, ktoré majú relevantne vysoké hodnotenie.

V ďalšom kroku budeme pre dvojicu termov zo vzťahu, o ktorom zisťujeme, či je hierarchický, porovnávať ich PageRank zoznamy. Ak sa oba termy nachádzajú v oboch zoznamoch a jeden term z dvojice sa nachádza v oboch zoznamoch vždy na vyššom mieste ako druhý, tak je tento term nadradený a medzi termami existuje hierarchický vzťah nadradenosti/podradenosti.

Tento postup je inšpirovaný *Semantic GrowBag* algoritmom [8]. Náš postup sa líši od Semantic Growbag algoritmu tým, že základ tvoria vzťahy vygenerované pomocou LSA a nie vzťahy založené na kolokáciach termov v dokumentoch. Druhým rozdielom je, že neaplikujeme PageRank algoritmus na vzťahy medzi termami, ale využívame štruktúru výučbového obsahu – termy prepojené na sieť dokumentov. Začiatočné uzly sú dokumenty, ku ktorým je priradený term. PageRank potom vráti zoznam termov, ktoré takisto súvisia s týmito dokumentmi. Na prvom mieste zoznamu by sa mal väčšinou nachádzať samotný term, pre ktorý sa generuje zoznam.

Postup určovania hierarchických vzťahov sme zapísali pseudoalgoritmom (Algoritmus 2). Vstupom algoritmu sú vzťahy vygenerované pomocou LSA (*RDT-RDTRels*), vzťahy medzi dokumentami a termami (*RDT-DOCRels*) a vzťahy medzi dokumentmi navzájom (*DOC-DOCRels*). Zo všetkých týchto vzťahov je zostrojený graf dokumentov prepojených na termy. Preto je vstupom aj súbor relevantných doménových termov (*RDTs*). Posledným parametrom je váha *w*, ktorá definuje, akú minimálnu váhu musí mať vzťah vygenerovaný pomocou LSA, aby bol kandidátom na hierarchický vzťah. Výstupom je množina objavených hierarchických vzťahov (*hierarchicalRels*).

Algoritmus 2. Pseudoalgoritmus PageRank variantu objavovania hierarchických vzťahov.

```
1: procedure FindHierarchical(RDT-RDTRels, RDT-DOCRels, DOC-DOCRels, RDTs, w)
2:     hierarchicalRels ← [ ]
3:     graph ← ConstructGraph(RDT-RDTRels, RDT-DOCRels, DOC-DOCRels)

4:     for all relationship ∈ RDT-RDTRels do
5:         if relationship[weight] > w then
6:             fromRDT ← relationship[from]
7:             fromStartNodes ← GetStartNodes(fromRDT, RDT-DOCRels)
8:             fromRDTPRList ← PageRankWithPriors(graph, fromStartNodes)

9:             toRDT ← relationship[to]
10:            toStartNodes ← GetStartNodes(toRDT, RDT-DOCRels)
11:            toRDTPRList ← PageRankWithPriors(graph, toStartNodes)

12:            if fromRDT ∈ fromRDTPRList ∧ fromRDT ∈ toRDTPRList then
13:                if toRDT ∈ fromRDTPRList ∧ toRDT ∈ toRDTPRList then

14:                    if Index(fromRDT, fromRDTPRList) < Index(toRDT, fromRDTPRList) then
15:                        if Index(fromRDT, toRDTPRList) < Index(toRDT, toRDTPRList) then
16:                            hierarchicalRels ← [toRDT, fromRDT, relationship[weight]]
17:                        end if

18:                    else if Index(fromRDT, fromRDTPRList) > Index(toRDT, fromRDTPRList) then
19:                        if Index(fromRDT, toRDTPRList) > Index(toRDT, toRDTPRList) then
20:                            hierarchicalRels ← [fromRDT, toRDT, relationship[weight]]
21:                        end if
22:                    end if
23:                end if
24:            end if
25:        end if
26:    end for
27:    return hierarchicalRels
28: end procedure
```

6 Overenie

Cieľom overenia bolo zistiť presnosť vygenerovaných vzťahov a overiť, či naša metóda generuje aj vzťahy, ktoré nedokáže objaviť metóda založená na ligvistickom prístupe.

Metódu sme overovali voči zlatému štandardu – existujúcim „ľahkým" ontológiám kurzov pre adaptívny výučbový systém ALEF opísaný v kapitole 2.2.3. Ontológie manuálne vytvorila skupina doménových expertov. Kurzy sú podrobne opísané v nasledujúcej podkapitole.

Druhou časťou overenia bolo porovnanie našich výsledkov s výsledkami existujúceho lingvistického prístupu pre objavovanie hierarchických vzťahov [35]. Porovnanie je opísané v podkapitole 6.4.

V rámci overenia sme metódu integrovali aj do systému pre správu výučbového obsahu, aby mohla byť využívaná učiteľmi pri tvorbe doménových modelov adaptívnych kurzov (kapitola 6.5).

6.1 Opis a predspracovanie dát

Zvolené dáta pre overenie našej metódy boli kurzy Logického a Funkcionálneho programovania. Kurzy sa skladali z dokumentov a relevantných doménových termov priradených k dokumentom. Dáta obsahujú málo dokumentov, čo môže negatívne vplývať na výsledky metódy. Na druhej strane sú dobre opísané priradenými relvantnými doménovými termami. Je škoda, že nie všetky relevantné doménové termy sú priradené k dokumentom, pretože vzťahy medzi dokumentmi a termami sú dôležitým vstupom našej metódy. Ale fakt, že term nie je priradený k žiadnemu dokumentu môže byť aj dôsledkom toho, že term sa viazal k nejakej otázke alebo cvičeniu z adaptívneho kurzu a my sme brali do úvahy len vysvetľujúce texty. V tomto prípade by mali autori kurzov zvážiť priradenie termov aj k vysvetľujúcim textom. Základné charakteristiky kurzov sú uvedené v tabuľke (Tab. 1).

Tab. 1. Charakteristiky kurzov Funkcionálneho a Logického programovania.

	Funkcionálne programovanie	Logické programovanie
počet dokumentov	79	42
počet slov	28 455	23 383
priemerná dĺžka dokumentu	360,19	556,74
počet vzťahov medzi dokumentmi	73	39
počet relevantných doménových termov	162	137
priemerná dĺžka relevantného doménového termu	1,70	1,41
počet priradených relevantných doménových termov korpusu	156	118
počet vzťahov medzi termami a dokumentmi	335	289
priemerný počet relevantných doménových termov na dokument	1,94	2,10

Korpus tvorili dokumenty, ktoré boli hierarchicky zoradené do kapitol podľa knižnej predlohy. Dokumenty boli v XML formáte a v slovenskom jazyku. Pred aplikáciou latentnej sémantickej analýzy ich bolo potrebné náležite predspracovať. Predspracovanie sa skladalo z týchto krokov:

- *odstránenie príkladov so zdrojovým kódom* – z dokumentov sme odstránili všetok text, ktorý predstavoval zdrojový kód, pretože tento text obsahoval veľa „slov", ktoré boli v skutočnosti len premenné alebo riadiace konštrukcie a nevplývali na význam vysvetľujúceho textu,
- *odstránanie tagov* – odstránili sme všetky XML značky a ponechali len čistý text,

- *normalizácia* – odstránili sme všetky špeciálne znaky a intepunkciu s výnimkou špeciálnych znakov, ktoré predstavovali relevantné doménové termy (napr. =.., <, *s-výraz*, *#'),*
- *odstránenie stop slov*,
- *lematizácia* – na lematizáciu sme použili databázu slovenských slov, databázu sme doplnili o chýbajúce slová tvoriace relevantné doménové termy, aby sme všetky ich výskyty dokázali identifikovať v texte,
- *vyfiltrovanie dokumentov podľa tf-idf* – dokumenty sú skrátené o slová, ktoré nepredstavujú relevantné doménové termy alebo sa nenachádzajú v najdôležitejších slovách podľa metriky *tf-ifdf* (takto je pri LSA kontext slova tvorený len relevantnými slovami).

Podrobnejší opis implementácie predspracovania sa nachádza v technickej dokumentácii (Príloha B).

6.2 Použité metriky

Pri vyhodnotení sme použili metriky presnosť, úplnosť a F-metriku. Pre porovnanie nami vytvoreného doménového modelu so zlatým štandardom sme použili modifikované metriky, pretože hierarchický vzťah *is-a* je tranzitívna relácia. Modifikácie boli prevzaté z práce [35].

Obidve modifikované metriky používajú pre porovnanie hierarchických vzťahov termu v dvoch doménových modeloch množinu všetkých jemu nadradených a všetkých jemu podradených termov v danom modeli (angl. *semantic cotopy*):

$$SC(rdt, DM) = \{rdt_j \in RDT_{DM} : isa_{DM}(rdt, rdt_j) \vee isa_{DM}(rdt_j, rdt)\} \tag{7}$$

SC je množina všetkých nadradených a podradených termov pre relevantný doménový term *rdt* v doménovom modeli *DM*. RDT_{DM} označuje množinu všetkých relevantných doménových termov z doménového modelu *DM*. Notácia isa_{DM} vyjadruje existenciu hierarchického vzťahu medzi danými termami v doménovom modeli *DM*.

Potom modifikované metriky sú definované nasledovne:

$$P_T(DM_{retr}, DM_{rel}) = \frac{\sum_{rdt \in DM_{retr} \cup DM_{rel}} |SC(rdt, DM_{retr}) \cap SC(rdt, DM_{rel})|}{\sum_{rdt \in DM_{retr} \cup DM_{rel}} |SC(rdt, DM_{retr})|}$$

$$R_T(DM_{retr}, DM_{rel}) = \frac{\sum_{rdt \in DM_{retr} \cup DM_{rel}} |SC(rdt, DM_{retr}) \cap SC(rdt, DM_{rel})|}{\sum_{rdt \in DM_{retr} \cup DM_{rel}} |SC(rdt, DM_{rel})|} \tag{8}$$

P_T a R_T je taxonomická presnosť a úplnosť nami získaného (angl. *retrieved*) doménového modelu DM_{retr} vzhľadom na príslušný (angl. *relevant*) doménový model – zlatý štandard DM_{rel}. Notácia $rdt \in DM$ vyjadruje, že *rdt* patrí do množiny RDT_{DM}. Takto definovaná presnosť a úplnosť zohľadňuje tranzitívnosť hierarchického vzťahu *is-a*. V prípade chybne vytvoreného hierarchického vzťahu odzrkadlí tento fakt aj na všetkých nadradených a podradených termoch.

Podmienkou pre objavenie vzťahu medzi termami je ich výskyt v texte. Pri spracovaní výučbového obsahu sa nám nepodarilo nájsť v texte všetky termy. Bolo to v prípade, ak sa nevyskytovali v texte dokumentov alebo sa vyskytovali v inom tvare ako ich autori kurzu definovali v doménovom modeli. V kurze Funkcionálneho programovania sme nenašli v texte 16 relevantných doménových termov, v kurze Logického programovania 7 relevantných doménových termov (Tab. 2).

Tab. 2. Relevantné doménové termy nenájdené v texte.

kurz	*neobjavené termy*
Funkcionálne programovanie	„oddp“, „totálna funkcia“, „cdar“, „cdadr“, „bodka-dvojica“, „apply“, „evenp“, „procedurálna paradigma programovania“, „deklaratívna paradigma programovania“, „aplikatívna paradigma programovania“, „logická paradigma programovania“, „predikát na test typu údaju“, „predikát na test poradia rovnosti“, „vstupno-výstupná priraďovacia funkcia“, „priama rekurzia“, „#“‘
Logické programovanie	„údajový typ“, „jeden krok rekurzie“, „redukcia rekurzie“, „postfixový operátor“, „zoznam (deep)“, „zoznam (top)“, „triedenie“

Keďže sa nám nepodarilo v texte objaviť všetky relevantné doménové termy definované v zlatom štandarde, nemohli sme objaviť ani všetky vzťahy zo zlatého štandardu. Preto sme použili na vyhodnotenie aj taxonomickú presnosť a úplnosť upravenú tak, aby nezohľadňovala nenájdené termy:

$$P'_T(DM_{retr}, DM_{rel}) = \frac{\sum_{rdt \in DM_{retr} \cup DM'_{rel}} |SC(rdt, DM_{retr}) \cap SC(rdt, DM_{rel})|}{\sum_{rdt \in DM_{retr} \cup DM'_{rel}} |SC(rdt, DM_{retr})|}$$

$$R'_T(DM_{retr}, DM_{rel}) = \frac{\sum_{rdt \in DM_{retr} \cup DM'_{rel}} |SC(rdt, DM_{retr}) \cap SC(rdt, DM_{rel})|}{\sum_{rdt \in DM_{retr} \cup DM'_{rel}} |SC(rdt, DM_{rel})|} \quad (9)$$

kde DM'_{rel} je zlatý štandard bez relevantných doménových termov, ktoré sme neidentifikovali v texte.

Druhá modifikácia taxonomickej presnosti a úplnosti zohľadňuje iba termy, medzi ktorými bol našou metódou identifikovaný hierarchický vzťah *is-a*. Je definovaná nasledovne:

$$P''_T(DM_{retr}, DM_{rel}) = \frac{\sum_{rdt \in DM_{retr}} |SC(rdt, DM_{retr}) \cap SC(rdt, DM_{rel})|}{\sum_{rdt \in DM_{retr}} |SC(rdt, DM_{retr})|}$$

$$R''_T(DM_{retr}, DM_{rel}) = \frac{\sum_{rdt \in DM_{retr}} |SC(rdt, DM_{retr}) \cap SC(rdt, DM_{rel})|}{\sum_{rdt \in DM_{retr}} |SC(rdt, DM_{rel})|} \quad (10)$$

Pre všetky tri varianty taxonomickej presnosti a úplnosti sme vyhodnotili aj F-metriku ako harmonický priemer predchádzajúcich dvoch metrík:

$$F_T = \frac{2 * P_T * R_T}{P_T + R_T} \tag{11}$$

Pre množiny relevantných doménových termov z doménových modelov platí:

$$RDT_{DM_{rel}} \supseteq RDT_{DM'_{rel}} \supseteq RDT_{DM_{retr}} \tag{12}$$

Potom suma

$$\sum_{rdt \in DM} | SC(rdt, DM_{retr}) \cap SC(rdt, DM_{rel}) |$$

pre $DM \in \{DM_{retr} \cup DM_{rel}, DM_{retr} \cup DM'_{rel}, DM_{retr}\}$ je vždy rovnaká a pre všetky tri varianty taxonomickej presnosti funkcia nadobúda rovnaké hodnoty. Platí teda:

$$P_T = P'_T = P''_T \tag{13}$$

6.3 Overenie voči zlatému štandardu

Naším zlatým štandardom boli doménové modely (ontológie) kurzov Logick0ho a Funkcion8lneho programovania vytvorené skupinou doménových expertov a autorom kurzov. Doménové modely sa skladali z relevantných doménových termov a vzťahov troch typov medzi termami. Typ *related-to* označuje sémantickú súvislosť termov, typ *is-a* vyjadruje hierarchický vzťah medzi termami (vzťah všeobecnosti/špecifikosti) a typ *prerequisity-to* vyjadruje vhodné poradie učenia sa termov pre ich pochopenie z pohľadu učiteľa. Charakteristiky zlatého štandardu sú uvedené v tabuľke (Tab. 3).

Tab. 3. Charakteristiky zlatého štandardu.

	Funkcionálne programovanie	Logické programovanie
počet relevantných doménových termov	162	137
počet vzťahov medzi termami	256	274
typ *related-to*	76	85
typ *is-a*	128	87
typ *prerequisity-to*	52	102

Keďže sme predpokladali, že naša metóda na objavovanie hierarchických vzťahov má na vstupe vzťahy podobnosti vygenerované latentnou sémantickou analýzou, zvolili sme štandardnú hodnotu parametra $w = 0{,}9$ pre obidva varianty metódy, t.j. metóda vykoná test na identifikáciu hierarchických vzťahov len nad vzťahmi, ktorých váha je väčšia ako 0,9.

Pri overovaní voči zlatému štandardu sme aplikovali našu metódu na tri súbory vzťahov *related-to*:

- vzťahy zo zlatého štandardu,
- automaticky vygenerované vzťahy podobnosti [36],
- vzťahy vygenerované latentnou sémantickou analýzou (kapitola 5.3).

6.3.1 Vzťahy zo zlatého štandardu

V tejto časti overenia sme aplikovali našu metódu pre objavovanie hierarchických vzťahov na súbor vzťahov zo zlatého štandardu a skúmali, či dokáže správne identifikovať hierarchické vzťahy.

Pri experimentoch sme vždy vybrali časť vzťahov podľa ich typu v zlatom štandarde, zmenili sme ich typ na *related-to* a aplikovali na ne našu metódu. Následne sme sledovali, koľkým z nich bol správne určený typ *is-a*. Tento spôsob overenia vychádza zo spôsobu, akým bol vytvorený zlatý štandard. Predpokladáme teda, že ak by zlatý štandard neobsahoval vzťahy typu *is-a*, nachádzali by sa medzi termami z *is-a* vzťahoch vzťahy typu *related-to*. Overujeme teda schopnosť metódy vytvoriť zlatý štandard.

Metódu sme aplikovali najprv na vzťahy všetkých typov a sledovali sme, koľko identifikuje správne ako hierarchické. V ďalšom pokuse sme aplikovali metódu len na vzťahy zo zlatého štandardu typu *related-to*. V tomto prípade sme očakávali, že metóda neobjaví veľa hierarchických vzťahov, pretože vzťahy typu *related-to* sa nenachádzajú v zlatom štandarde medzi termami, ktoré sú už prepojené vzťahom *is-a*. Potom sme aplikovali metódu na vzťahy zlatého štandardu typu *related-to* aj *is-a*. Vtedy by sa počet objavených hierarchických vzťahov mal zvýšiť. V poslednom pokuse sme aplikovali metódu len na vzťahy typu *prerequisity-to*. Sledovali sme, či sa dajú hierarchické vzťahy objaviť aj na základe týchto vzťahov.

Výsledky pokusov sú zaznamenané v tabuľke (Tab. 4) pre obidva varianty metódy na objavovanie hierarchických vzťahov. Pre každý pokus s daným kurzom je zaznamenaný počet všetkých objavených vzťahov, počet správnych vzťahov, t.j. vzťahov, ktoré sa nachádzajú v zlatom štandarde buď priamo alebo tranzitívne, a vypočítaná taxonomická presnosť a úplnosť. Keďže sa jedná o vzťahy zo zlatého štandardu, nevyskytli sa žiadne neobjavené relevantné doménové termy. Metrika R'_T má preto rovnakú hodnotu ako R_T a neuvádzame ju vo výsledkoch.

Uvedené výsledky boli získané pri aplikácii metódy na vzťahy s váhou w väčšou ako 0,9, čo je štandardný parameter oboch variantov metódy. Najlepšie výsledky boli ale dosiahnuté pri aplikácii na vzťahy s váhou väčšou ako 0,75 v kurze Logického programovania a 0,94 v kurze Funkcionálneho programovania. Nad týmito hranicami sa už nevyskytovali vzťahy typu *prerequisity*-to, ktoré znižovali presnosť. Podrobnejšie výsledky sú uvedené v prílohe (pozri Príloha C).

Tab. 4. Výsledky overenia na vzťahoch zlatého štandardu (w = 0,9) (FP – Funkcionálne programovanie, LP – Logické programovanie, obj. – počet objavených vzťahov, spr. – počet správnych vzťahov).

	vzťahy	*množinový variant*							*PageRank variant*						
		obj.	*spr.*	P_T	R_T	F_T	R''_T	F''_T	*obj.*	*spr.*	P_T	R_T	F_T	R''_T	F''_T
FP	*všetky*	85	76	0,80	0,36	0,49	0,48	0,60	78	43	0,87	0,28	0,28	0,41	0,56
	related-to	6	0	0	0	0	0	0	6	0	0	0	0	0	0
	rel.-to+is-a	83	77	**0,91**	0,36	0,51	**0,47**	**0,62**	75	43	**0,89**	0,28	0,43	**0,42**	**0,57**
	prerequisity	3	1	0,33	0	0,01	0,5	0,4	3	0	0	0	0	0	0
LP	*všetky*	92	50	0,52	0,32	0,40	0,37	0,43	22	12	0,50	0,06	0,11	0,50	0,50
	related-to	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	rel.-to+is-a	59	53	**0,83**	0,35	0,49	**0,65**	**0,73**	14	12	**0,93**	0,06	0,12	**0,54**	**0,68**
	prerequisity	43	0	0	0	0	0	0	8	0	0	0	0	0	0

6.3.2 Automaticky vygenerované vzťahy podobnosti

Pri overovaní sme mali k dispozícii okrem zlatého štandardu aj iné už existujúce doménové modely pre kurzy Logického a Funkcionálneho programovania, ktoré obsahovali rovnaké relevantné doménové termy ako zlatý štandard. Vzťahy medzi termami boli automaticky vygenerované vzťahy podobnosti, t.j. všetky boli len typu *related-to*. Tieto doménové modely boli vytvorené v práci [36].

V tejto časti overenia sme aplikovali našu metódu pre objavovanie hierarchických vzťahov na súbor automaticky vygenerovaných vzťahov z tohto doménového modelu a skúmali, či sa podarí objaviť hierarchické vzťahy, ktoré sú v zlatom štandarde.

Doménový model kurzu Funkcionálneho programovania mal 492 vzťahov, Logického programovania 603 vzťahov. Keďže skoro žiadny z týchto vzťahov nemal váhu väčšiu ako 0,9, hľadali sme hodnotu parametra w, pri ktorej metóda dosiahla najlepšie výsledky na týchto vzťahoch. Podrobnejší opis vzťahov podobnosti sa nachádza v prílohe (Príloha C).

Výsledky sú uvedené v tabuľke nižšie (Tab. 5). Keďže v týchto vzťahoch sa vyskytovali všetky relevantné doménové termy, neuvádzame vo výsledkoch R'_T, pretože je rovnaké ako R_T.

Tab. 5. Výsledky overenia na automaticky vygenerovaných vzťahoch podobnosti (FP – Funkcionálne programovanie, LP – Logické programovanie, obj. – počet objavených vzťahov, spr. –počet správnych vzťahov).

	w	*množinový variant*							*PageRank variant*						
		obj.	*spr.*	P_T	R_T	F_T	R''_T	F''_T	*obj.*	*spr.*	P_T	R_T	F_T	R''_T	F''_T
FP	0,06	319	35	**0,20**	**0,26**	**0,23**	0,28	0,23	302	43	0,23	0,25	0,24	0,28	0,25
	0,07	307	35	0,20	0,25	0,22	0,26	0,22	298	42	**0,24**	**0,24**	**0,24**	0,28	0,26
	0,50	24	11	0,52	0,05	0,09	0,35	0,42	22	11	**0,57**	0,05	0,09	**0,39**	**0,46**
	0,66	11	4	**0,45**	0,02	0,04	**0,42**	**0,43**	11	4	0,45	0,02	0,04	0,42	0,43
LP	0	495	31	**0,09**	**0,27**	**0,14**	0,56	0,16	416	22	0,09	0,18	0,12	0,40	0,15
	0,07	400	23	0,07	0,14	0,09	0,31	0,11	373	17	**0,10**	**0,18**	**0,13**	0,40	0,15
	0,19	52	10	**0,19**	0,05	0,08	**0,67**	**0,29**	53	9	**0,20**	0,05	0,08	**0,71**	**0,31**

Na vzťahoch z kurzu Funkcionálneho programovania boli dosiahnuté najlepšie hodnoty metriky F_T pri aplikácii na vzťahy s váhou väčšou ako $w = 0,06$ pri množinovom variante metódy a $w = 0,07$ pri PageRank variante metódy, t.j. metóda bola aplikovaná skoro na všetky vzťahy modelu (95,5 % a 93,7 % vzťahov). Najlepšie hodnoty F''_T boli dosiahnuté pri $w = 0,66$ (2,4 % vzťahov) pri množinovom variante metódy a $w = 0,50$ (5 % vzťahov) pri PageRank variante metódy.

Experimentmi na vzťahoch z kurzu Logického programovania sme dosiahli najlepšiu hodnotu F_T pri aplikácii metódy na všetky vzťahy pri množinovom variante metódy a pri aplikácii metódy na vzťahy s váhou väčšou ako $w = 0,07$ (88,89 % vzťahov) v prípade PageRank variantu metódy. Najlepšia hodnota F''_T bola dosiahnutá pri $w = 0,19$ (13,43 % vzťahov) pre obidva varianty metódy.

6.3.3 Vzťahy vygenerované latentnou sémantickou analýzou (LSA)

Nakoniec sme aplikovali našu metódu pre objavovanie hierarchických vzťahov na vzťahy vygenerované latentnou sémantickou analýzou, tak ako je to v návrhu našej metódy. Zistili sme, že počet vygenerovaných vzťahov LSA závisí od veľkosti okolia relevantných doménových termov v texte, ktoré porovnávame. Čím väčšie okolie, tým viac vygenerovaných vzťahov. Pre najlepšie výsledky metódy pre objavovanie hierarchických vzťahov sme použili veľkosť okolia 10 slov pre relevantné doménové termy z kurzu Funkcionálneho programovania a veľkosť 8 slov pre termy z kurzu Logického programovania. Tieto veľkosti boli získané experimentálne.

Váha vzťahov vygenerovaných LSA je vypočítaná ako kosínusová podobnosť okolí termov. Pohybuje sa v intervale od 0 po 1. Preto sme zvolili štandardnú hodnotu parametra metódy pre objavovanie vzťahov $w = 0{,}9$, t.j. všetky vzťahy medzi termami, ktoré majú váhu väčšiu ako 0,9 budú otestované, či sú hierarchické. Experimenty sme najprv vykonávali s touto zvolenou štandardnou hodnotou. Neskôr sme hľadali hodnotu w, pri ktorej metóda dosahovala najlepšie výsledky.

Pre kurz Funkcionálneho programovania bolo vygenerovaných pomocou LSA 2631 vzťahov. Z nich 318 vzťahov malo váhu väčšiu ako $w = 0{,}9$. Kurz Logického programovania mal 2573 vzťahov vygenerovaných LSA, z toho 248 malo váhu väčšiu ako 0,9. Výsledky pre tieto vzťahy sú uvedené v nasledujúcej tabuľke (Tab. 6). Pri týchto vzťahoch sme ďalej sledovali ako sa mení presnosť a úplnosť s počtom vygenerovaných vzťahov a pri koľkých vzťahoch boli dosiahnuté najlepšie výsledky (Obr. 10, Obr. 11).

Tab. 6. Výsledky overenia na vzťahoch podobnosti vygenerovaných pomocou LSA (FP – Funkcionálne programovanie, LP – Logické programovanie, obj. – počet objavených vzťahov, spr. –počet správnych vzťahov).

	w	*množinový variant*									*PageRank variant*								
		obj.	*spr.*	P_T	R_T	F_T	R'_T	F'_T	R''_T	F''_T	*obj.*	*spr.*	P_T	R_T	F_T	R'_T	F'_T	R''_T	F''_T
FP	0,83	469	86	0,13	0,45	0,20	0,57	0,21	0,58	0,21	147	36	0,27	0,19	**0,22**	0,24	**0,25**	0,31	0,29
	0,92	119	46	0,42	0,26	0,32	0,33	**0,37**	0,51	0,46	296	44	0,15	0,22	0,18	0,28	0,20	0,31	0,20
	0,90	184	63	0,31	0,36	**0,33**	0,46	0,37	0,53	0,39	86	27	0,33	0,12	0,18	0,16	0,21	0,25	0,28
	0,99	16	9	0,56	0,04	0,07	0,05	0,08	0,69	**0,62**	12	7	**0,58**	0,03	0,05	0,04	0,07	0,78	**0,67**
LP	0,83	461	57	0,09	0,56	**0,16**	0,62	**0,16**	0,67	0,16	147	9	0,09	0,12	0,10	0,13	0,11	0,32	0,14
	0,85	383	51	0,08	0,38	0,13	0,42	0,13	0,48	0,14	131	9	0,09	0,12	**0,10**	0,13	**0,11**	0,32	0,15
	0,90	162	24	0,10	0,19	0,13	0,21	0,13	0,46	0,16	64	5	0,15	0,06	0,08	0,06	0,09	0,27	0,2
	0,99	14	5	0,36	0,02	0,05	0,03	0,05	0,33	**0,34**	9	2	0,22	0,01	0,02	0,01	0,02	0,33	**0,27**

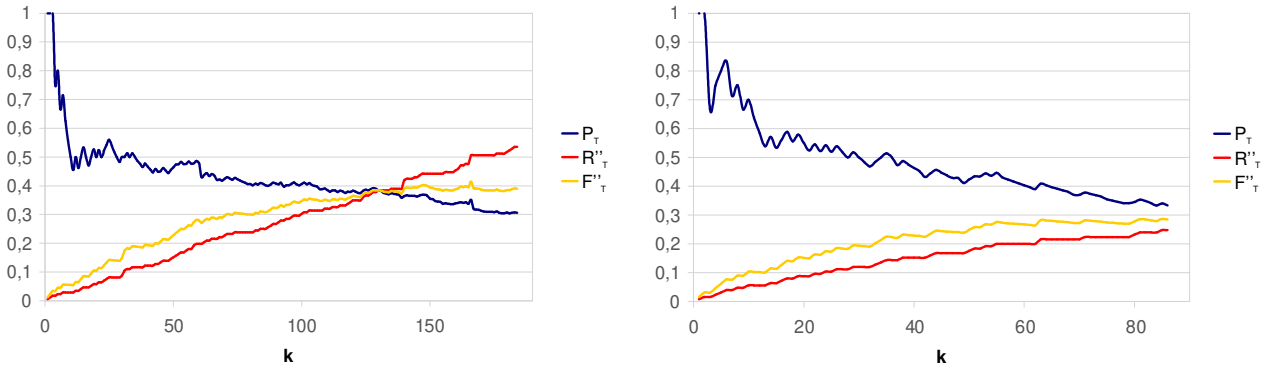
Pri hľadaní hodnoty w, pri ktorej metóda dosahovala najlepšie výsledky, sme prišli na nasledujúce:

Najlepšia hodnota F_T a F'_T pre kurz Logického programovania bola dosiahnutá na vzťahoch s váhou väčšou ako $w = 0{,}83$ (26,98 % všetkých vzťahov) pre množinový variant metódy a $w = 0{,}85$ (22,53 % vzťahov) pre PageRank variant. Najlepšia hodnota F''_T bola pri hodnote $w = 0{,}99$ (1,14 % vzťahov).

V prípade kurzu Funkcionálneho programovania bola najlepšia F_T vypočítaná pre $w = 0{,}9$ (12,47 % vzťahov), najlepšia F'_T pre $w = 0{,}92$ (8,63 % vzťahov) pre množinový variant metódy. Pre PageRank variant bola najlepšia hodnota F_T a F'_T vypočítaná pre $w = 0{,}83$ (28,81 % vzťahov). Najlepšia hodnota F''_T bola pre $w = 0{,}99$ (0,87 % vzťahov) v oboch variantoch metódy pre objavovanie hierarchických vzťahov.

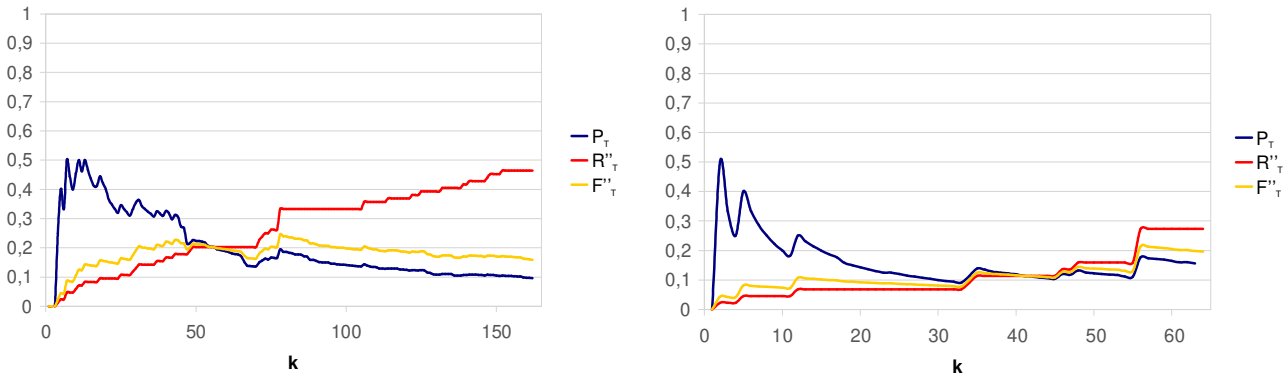
Pre štandardnú hodnotu parametra $w = 0{,}90$ sme sledovali ako sa mení s počtom vygenerovaných vzťahov úplnosť a presnosť. Na grafoch sú zobrazené presnosť, úplnosť a F-metrika oboch variantov metódy pre kurz Funkcionálneho (Obr. 10) aj Logického programovania (Obr. 11). Vzťahy sú zoradené podľa ich váhy.

Najlepšie hodnoty pre kurz Funkcionálneho programovania boli nasledovné. V množinovom variante metódy bola najlepšia hodnota F-metriky $F''_T = 0{,}41$ pri $k = 166$ vygenerovaných vzťahoch. Presnosť v tomto bode bola $P_T = 0{,}36$ a úplnosť $R''_T = 0{,}51$. Najlepšia dosiahnutá presnosť bola $P_T = 1$ ($k = 3$) a úplnosť $R''_T = 0{,}53$ ($k = 183$). V PageRank variante metódy bola najlepšia hodnota F-metriky $F''_T = 0{,}29$ pri $k = 85$ vygenerovaných vzťahoch. Presnosť v tomto bode bola $P_T = 0{,}34$ a úplnosť $R''_T = 0{,}25$. Najlepšia dosiahnutá presnosť bola $P_T = 1$ ($k = 2$) a úplnosť $R''_T = 0{,}25$ ($k = 85$).



Obr. 10. Presnosť, úplnosť a F-metrika prvých k vygenerovaných vzťahov pre kurz Funkcionálneho programovania. Vľavo je množinový variant metódy, vpravo PageRank variant.

Experiment s kurzom Logického programovania dosiahol nasledovné najlepšie hodnoty. V množinovom variante metódy bola najvyššia hodnota F-metriky $F''_T = 0{,}25$ pri $k = 78$ vygenerovaných vzťahoch. Presnosť v tomto bode bola $P_T = 0{,}19$ a úplnosť $R''_T = 0{,}33$. Najlepšia dosiahnutá presnosť bola $P_T = 0{,}50$ ($k = 13$) a úplnosť $R''_T = 0{,}46$ ($k = 152$). V PageRank variante metódy bola najlepšia hodnota F-metriky $F''_T = 0{,}21$ pri $k = 56$ vygenerovaných vzťahoch. Presnosť v tomto bode bola $P_T = 0{,}18$ a úplnosť $R''_T = 0{,}27$. Najlepšia dosiahnutá presnosť bola $P_T = 0{,}50$ ($k = 2$) a úplnosť $R''_T = 0{,}27$ ($k = 56$). Výsledky s metrikami R_T, F_T *a* R'_T, F'_T sú uvedené v prílohe (Príloha C).



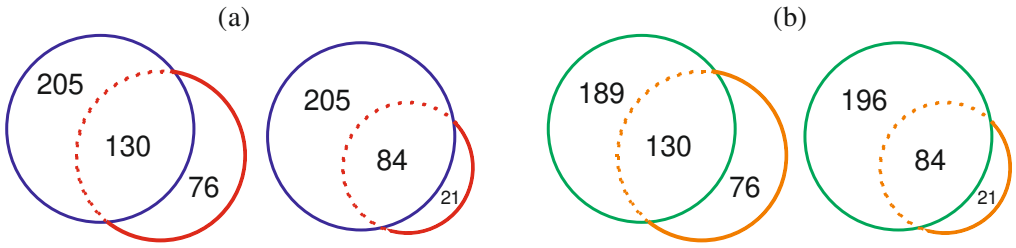
Obr. 11. Presnosť, úplnosť a F-metrika prvých k vygenerovaných vzťahov pre kurz Logického programovania. Vľavo je množinový variant metódy, vpravo PageRank variant.

6.3.4 Obohacovanie vzťahmi medzi termami a dokumentmi

V snahe zlepšiť výsledky našej metódy sme navrhli vylepšiť jej vstup – vzťahy medzi relevantnými doménovými termami a dokumetmi. Keďže tieto vzťahy sú dôležitým vstupom pre oba varianty metódy na objavovanie hierarchických vzťahov, rozhodli sme sa existujúce vzťahy doplniť. Dokumentom sme priradili ďalšie relevantné doménové termy na základe *tf-idf* metriky. Ak sa relevantný doménový term nachádzal medzi prvými desiatimi najdôležitejšími slovami dokumentu, bol doplnený vzťah medzi dokumentom a týmto termom. Kurz Funkcionálneho programovania sa nám takto podarilo doplniť o 76 vzťahov, kurz Logického programovania o 21 vzťahov (Obr. 12).

Všetky predchádzajúce experimenty sme vykonali aj s doplnenými vzťahmi. Vzťahy zlepšili úplnosť pri všetkých experimentoch priemerne o 1,75 %, ale presnosť nezlepšili. Ak sme metódu aplikovali

len na vygenerované vzťahy medzi termami a dokumentmi (bez existujúcich), presnosť metódy sa zlepšila v priemere o 2 %, ale úplnosť klesla priemerne o 9 %, pretože metóda našla menej vzťahov.



Obr. 12. Podiel existujúcich a doplnených vzťahov medzi relevantnými doménovými termami a dokumentmi. (a) pre všetky relevantné doménové termy, (b) len pre relevantné doménové termy objavené v texte, Funkcionálne programovanie vždy vľavo, Logické programovanie vždy vpravo.

6.3.5 Zhrnutie

Pokusmi na vzťahoch zo zlatého štandardu sme dokázali, že metóda má potenciál odhaliť hierarchické vzťahy. Vidno to na poklese nájdených vzťahov pri aplikovaní metódy len na vzťahy typu *related-to*. Ďalej sme zistili, že vzťahy typu *prerequisity-to* nie sú vhodným vstupom pre našu metódu na objavovanie hierarchických vzťahov, pretože presnosť metódy klesla, ak boli súčasťou vstupu (aplikovanie na všetky vzťahy zo zlatého štandardu) a pri aplikovaní metódy len na tieto vzťahy neboli objavené hierarchické vzťahy zo zlatého štandardu.

Experimenty s automaticky vygenerovanými vzťahmi podobnosti ukázali, že metóda dokázala nájsť hierarchické vzťahy s rozumnou presnosťou a úplnosťou, ale až po úprave hodnoty vstupného parametra w tak, aby metóda bola aplikovaná len na malé percento vzťahov. Tento fakt bol spôsobený tým, že v týchto doménových modeloch korešpondovalo len malé percento vzťahov so vzťahmi zo zlatého štandardu a naša metóda očakávala na vstupe vzťahy vygenerované pomocou LSA.

Pri aplikácii metódy na vzťahy vygenerované pomocou LSA sme dosiahli pri štandardnej hodnote parametra $w = 0{,}9$ celkom dobrú presnosť a úplnosť. Pri Funkcionálnom programovaní presnosť 0,36 a úplnosť 0,51 pri 166 vygenerovaných vzťahoch, pričom bolo objavených zhruba tretina vzťahov zo zlatého štandardu. V kurze Logického programovania sme dosiahli presnosť 0,19 a úplnosť 0,33 pri 78 vygenerovaných vzťahoch. Tak ako pri automaticky vygenerovaných vzťahoch podobnosti aj v prípade vzťahov vygenerovaných pomocou LSA dávala metóda najlepšie výsledky pri aplikácii len na malé percento vzťahov, konkrétne pri parametri $w = 0{,}99$. To dokazuje náš predpoklad, že je veľká pravdepodobnosť, že medzi veľmi podobnými pojmami – synonymami existuje vzťah *is-a*.

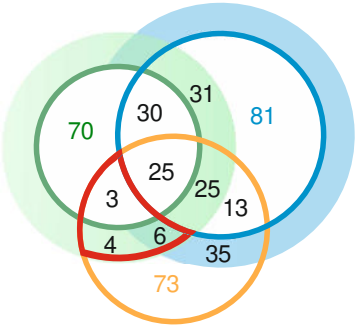
Uvedená presnosť a úplnosť platia pre množinový variant metódy na objavovanie hierarchických vzťahov, ktorý vo všeobecnosti vždy objavil viac vzťahov ako PageRank variant. PageRank variant metódy neobjavil skoro žiadne správne vzťahy navyše ako množinový variant. V kurze Funkcionálneho programovania našiel 18 vzťahov nenájdených množinovým variantom, ale len 2 zo zlatého štandardu, v kurze Logického programovania 17 vzťahov, ale žiadny správny. Preto uprednostňujeme množinový variant metódy pred PageRank variantom.

Pri obohacovaní vzťahov medzi relevantnými doménovými termami a dokumentmi sa zvýšila úplnosť metódy, ale zároveň tieto vzťahy negatívne vplývali na presnosť a metóda našla menej vzťahov. Tento jav je nežiaduci pri zostavovaní doménového modelu, preto uprednostňujeme vzťahy definované

autorom kurzu. V prípade, že tieto vzťahy chýbajú, vystačí si naša metóda aj s dodefinovanými vzťahmi, t.j. nezhorší sa jej presnosť, ale vygeneruje približne o štvrtinu menej vzťahov.

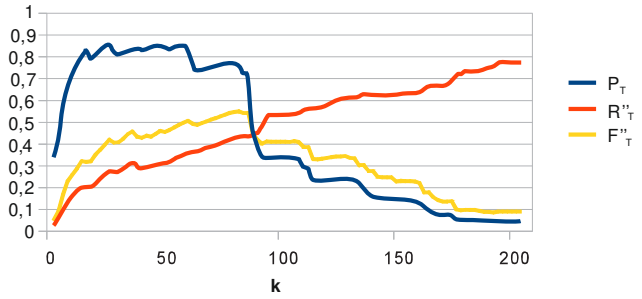
6.4 Porovnanie s existujúcim prístupom

Okrem porovnania so zlatým štandardom sme porovnávali našu metódu aj s existujúcou metódou na objavovanie hierarchických vzťahov založenou na lingvistickom spracovaní [35]. Vzťahy sú objavované hľadaním definovaných fráz, ktoré naznačujú hierarchický vzťah. Táto metóda bola overovaná na rovnakých dátach. K dispozícii sme mali výstup pre kurz Funkcionálneho programovania. Porovnali sme hierarchické vzťahy nájdené týmto prístupom s výstupom našej metódy. Konkrétne sme porovnali výstup množinového variantu našej metódy na vzťahoch nájdených pomocou LSA s parametrom $w = 0{,}9$ (184 vzťahov) s 205 vzťahmi vygenerovanými lingvistickým prístupom (Obr. 13). Zlatý štandard tvorilo 128 hierarchických vzťahov. Vidíme, že naša metóda dokázala objaviť vzťahy zo zlatého štandardu, ktoré neobjavil lingvistický prístup. Preto by mohla byť použiteľná ako doplnok lingvistického prístupu.



Obr. 13. Objavené vzťahy (žlté) porovnané so vzťahmi vytvorenými lingvistickým prístupom (modré) a vzťahmi zo zlatého štandardu (zelené). V bledých častiach sa nachádzajú vzťahy, ktoré sa nachádzajú v daných doménových modeloch, ale len tranzitívne. Červenou farbou je vyznačená množina vzťahov zo zlatého štandardu, ktorú nedokázal objaviť lingvistický prístup.

Ak porovnáme presnosť a úplnosť našej metódy (Obr. 10 vľavo) s lingvistickým prístupom (Obr. 14), vidíme, že naša metóda má celkovú menšiu úplnosť, ale presnosť s počtom vygenerovaných vzťahov neklesá tak prudko a zastaví sa na hodnote $P_T = 0{,}31$. Lingvistický prístup dosahuje lepšiu hodnotu F-metriky $F''_T = 0{,}55$ ($k = 83$) v porovnaní s našou najlepšou hodnotou $F''_T = 0{,}41$ ($k = 166$).



Obr. 14. Presnosť, úplnosť a F-metrika prvých k vzťahov vygenerovaných lingvistickým prístupom [35].

6.5 Vygenerované vzťahy

V tabuľke (Tab. 7) uvádzame prvých 20 vzťahov podľa ich váhy vygenerovaných množinovým variantom našej metódy. Vidno, že vzťahy pochádzajú z latentnej sémantickej analýzy. Váha vzťahu vyjadruje kosínusovú podobnosť slov nachádzajúcich sa okolo termu v texte. Pozorujeme, že väčšinou sa nadradený všeobecnejší term vyskytuje v názve špecifickejšieho, t.j. oba termy boli lokalizované na tom istom mieste v texte, a preto majú veľmi podobné okolité slová a vznikol vzťah s vysokou váhou. Táto skutočnosť avšak ojedinele vyprodukovala zlé vzťahy, napr. medzi termami *vyhodnotenie výrazu* a *výraz* alebo *vykonanie kroku rekurzia* a *rekurzia*.

Tab. 7. Prvých 20 vzťahov podľa váhy vygenerovaných množinovým variantom metódy na objavovanie hierarchických vzťahov. Posledný stĺpec zaznamenáva, či sa vzťah nachádzal v zlatom štandarde.

podradený term	*nadradený term*	*váha*	*správny?*
podvýraz	výraz	1	nie
čistý výraz	výraz	1	áno
FIRST/REST rekurzia	rekurzia	1	áno
lokálna premenná	premenná	1	áno
vyhodnotenie výrazu	výraz	1	nie
rekurzívna funkcia	funkcia	1	áno
funkcia	čiastočná funkcia	0,997191	áno
rest	first	0,997002	nie
atomický typ údaju | atomický údajový typ | atóm	zoznam	0,996054	nie
lambda výraz	výraz	0,994253	nie
forma	funkcia	0,994153	nie
paradigma programovania ohraničeniami	paradigma programovania	0,994111	áno
kompozícia	funkcia	0,993601	nie
monotónna rekurzia	rekurzia	0,991675	áno
s-výraz | symbolický výraz	výraz	0,991658	áno
cdr	car	0,991593	nie
vykonanie kroku rekurzia	rekurzia	0,991179	nie
priraďovacia funkcia	funkcia	0,990443	áno
cdr	CxR	0,989399	áno

V nasledujúcej tabuľke (Tab. 8) sú vzťahy zo zlatého štandardu, ktoré neobjavil lingvistický prístup ale naša metóda áno.

Tab. 8. Vzťahy zo zlatého štandardu objavené našou metódou, ale neobjavené lingvistickým prístupom.

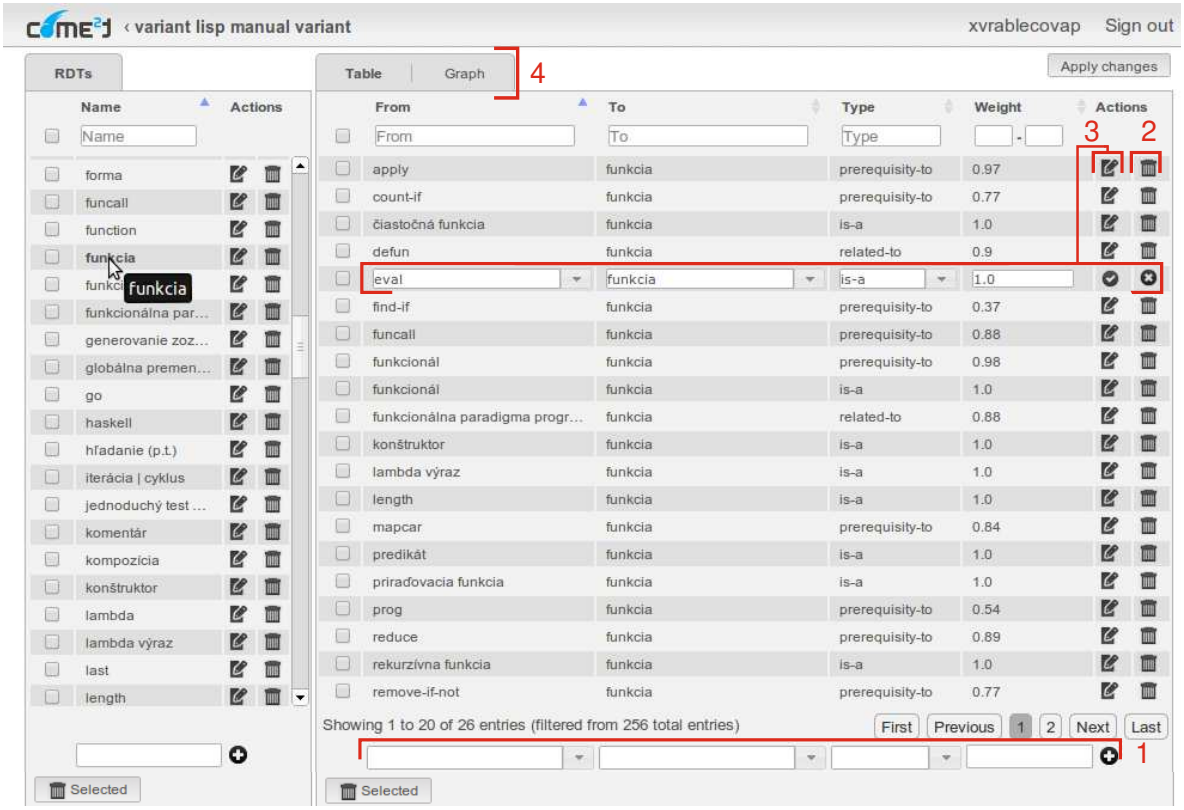
podradený term	*nadradený term*	*váha*
cdr	CxR	0,989399
(apostrof)	funkcia	0,986072
car	CxR	0,968813
rozlišovač	funkcia	0,951928
logický predikát	funkcia	0,946679
not	predikát	0,946593
mapcar	funkcia	0,939026
funcall	funkcia	0,931528
prin1	funkcia	0,931289
<	funkcia	0,925593
reduce	funkcia	0,913049
CxR	funkcia	0,905272
or	funkcia	0,902196

6.6 Integrácia do systému COME^{2}T

Naša metóda na objavovanie hierarchických vzťahov bola integrovaná do systému COME^{2}T [14], ktorý slúži na správu výučbového obsahu portálu ALEF opísaného v kapitole 2.2.3. Tento systém bol vytvorený v rámci predmetu Tímový projekt v ak. r. 2011/2012 pod vedením M. Šimka.

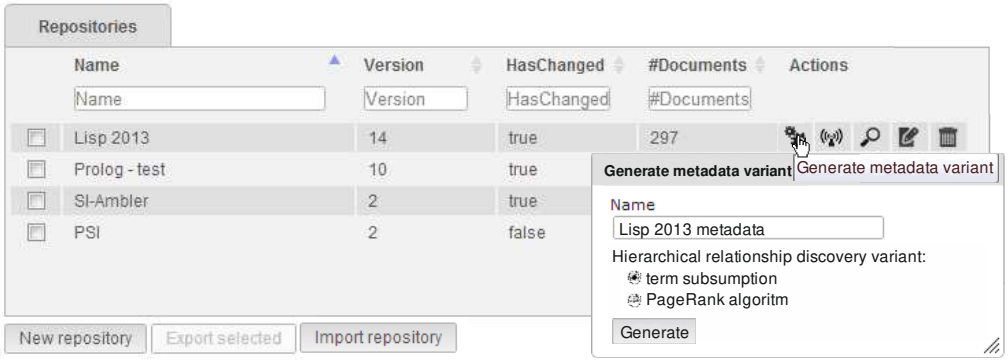
Predpokladaný scenár použitia metódy v systéme je, že učiteľ použije metódu na vygenerovanie hierarchických vzťahov. Následne pomocou rozhrania na manuálne vytváranie doménového modelu v systéme COME^{2}T odstráni tie vygenerované vzťahy, ktoré sú podľa neho nesprávne a doménový model doplní o vzťahy, ktoré podľa neho chýbajú. Takýto postup je pre učiteľa z hľadiska vynaloženého úsilia určite lepší ako keby mal učiteľ vytvárať celý doménový model úplne sám.

Na obrázku (Obr. 15) je znázornené rozhranie systému COME^{2}T pre manuálnu tvorbu doménového modelu. Tabuľka vľavo je určená na prácu s relevantnými doménovými termami (pridávanie, odoberanie, upravovanie). Tabuľka vpravo je určená na prácu so vzťahmi medzi termami. Vzťahy je možné pridávať (1). Vzťah je definovaný dvoma termami (na obrázku *From* a *To*), typom vzťahu a váhou z intervalu 0 až 1. Ďalej rozhranie umožňuje vymazávanie vzťahov (2) a aktualizovanie (3). Aktualizovať sa dajú všetky parametre uvedené pri vytváraní vzťahu. Obe tabuľky poskytujú filtrovanie a zoraďovanie. Po výbere termu (kliknutí naň) v tabuľke relevantných doménových termov je tabuľka vzťahov vyfiltrovaná len na vzťahy, v ktorých sa nachádza vybratý term (zobrazené na obrázku pre term *funkcia*). Tieto prvky uľahčujú navigáciu v doménovom modeli. Pre ešte lepšiu prácu s doménovým modelom ho systém umožňuje aj vizualizovať vo forme grafu (4). Grafové rozhranie však momentálne plne podporuje len prácu s relevantnými doménovými termami a umožňuje len pridávanie a prezeranie vzťahov.



Obr. 15. Rozhranie systému COME^{2}T na manuálnu tvorbu doménového modelu.

Veríme, že naša metóda uľahčí manuálne zostavovanie doménového modelu napriek tomu, že jej presnosť a úplnosť nie sú stopercentné. Návrh rozhrania pre automatické generovanie doménového modelu je na obrázku (Obr. 16). Detaily o implementácii sa nachádzajú v technickej dokumentácii (Príloha B).



Obr. 16. Návrh rozhrania pre automatické generovanie doménového modelu v systéme COME^{2}T. Repozitár v systéme predstavuje kurz – obsahuje všetky dokumenty kurzu.

6.7 Zhrnutie

Overením voči zlatému štandardu sme dokázali, že metóda má potenciál objaviť hierarchické vzťahy z výučbového obsahu, pretože dokázala správne identifikovať väčšinu hierarchických vzťahov zo zlatého štandardu. Potvrdili sme predpoklad, že vzťahy vygenerované latentnou sémantickou analýzou s vysokou váhou sú s veľkou pravdepodobnosťou hierarchické. Zistili sme, že množinový variant metódy dokázal objaviť viac vzťahov ako PageRank variant. Preto je vhodnejší na integráciu do systému na správu výučbového obsahu. Ďalej sme zistili, že naša metóda je schopná generovať vzťahy aj v prípade, že korpus dokumentov nemá priradené relevantné doménové termy a vstupom metódy je len korpus a súbor relevantných doménových pojmov.

V porovnaní s existujúcim lingvistickým prístupom má naša metóda menšiu úplnosť a presnosť, ale s počtom vygenerovaných vzťahov je presnosť stabilnejšia a pomalšie sa blíži k nule. Takisto sa ukázalo, že naša metóda dokáže objaviť hierarchické vzťahy, ktoré neobjavil lingvistický prístup. Ďalšou výhodou našej metódy je, že v systéme na správu výučbového obsahu môže objavovať vzťahy v akomkoľvek výučbovom obsahu a nie je obmedzená na výskyt definovaných fráz. Jazyková závislosť spočíva len v predspracovaní výučbového obsahu.

Na výsledky overenia mohol negatívne vplývať fakt, že náš korpus bol v slovenskom jazyku a lematizácia bola limitovaná len na slová, ktoré sa nachádzali v databáze. Prípadne mohli byť relevantné doménové termy spomínané v texte len zámenami, takže ich nebolo možné lokalizovať v texte. Preto by bolo vhodné začleniť do predspracovania vyriešenie anafor a odkazujúcich výrazov.

Ďalej na výsledky vplývala veľkosť korpusu. V prípade štatistických metód je potrebné väčšie množstvo dát. Tento fakt je možné pozorovať aj porovnaním výsledkov metódy na rôznych kurzoch. Experimenty na kurze Funkcionálneho programovania so 79 dokumentmi poskytli lepšie výsledky ako experimenty na kurze Logického programovania so 42 dokumentmi.

Presnosť, úplnosť a F-metrika boli ovplyvnené aj kvalitou zlatého štandardu, ktorý bol zostavený len malou skupinou doménových expertov, a neobsahuje všetky hierarchické vzťahy. Preto aj správne vzťahy mohli byť označené za nesprávne v prípade, že sa nevyskytovali v zlatom štandarde.

7 Záver

V práci sme sa zamerali na obsah adaptívnych výučbových systémov. Adaptívne systémy potrebujú okrem obsahu, ktorý zobrazujú používateľom poznať aj jeho význam, aby mohli používateľom odporúčať vhodný obsah a sledovať ich progres vo výučbovom procese. Preto je potrebné opísať výučbový obsah dátami v strojom spracovateľnom formáte. Na vyjadrenie sémantiky obsahu z určitej doménovej oblasti slúžia ontológie. My sme sa zamerali na tzv. „ľahké“ ontológie, ktoré sa skladajú z relevantných doménových termov a vzťahov medzi nimi. Ontológia tvorí časť adaptívneho systému nazývanú doménový model. Najsignifikantnejšie vzťahy doménového modelu sú hierarchické vzťahy, ktoré dokážu zoradiť relevantné doménové termy od najvšeobecnejších po najšpecifickejšie. Je tu uplatnený princíp dedičnosti, t.j. špecifickejšie termy zdieľajú všetky vlastnosti všetkých nadradených všeobecnejších termov. Druhým dôležitým typom vzťahu v doménovom modeli sú vzťahy podobnosti, ktoré adaptívny systém takisto využíva pri odporúčaní obsahu.

Adaptívny výučbový obsah je celý vytváraný pedagógmi. Okrem textovej časti ktorú vidia študenti, musia zostrojiť aj relevantné doménové termy a vhodne ich poprepájať. Manuálne vytvorenie doménového modelu je pre učiteľov náročné. Preto vznikajú metódy na automatizáciu tohto procesu. My sme sa zamerali na objavovanie vzťahov medzi relevantnými doménovými termami.

Analyzovali sme súčasné možnosti objavovania vzťahov z výučbového obsahu. Zistili sme, že výučbový obsah je vhodný na objavovanie vzťahov jednak pre to, že výučbové texty sú jednoznačne napísané a jednak aj pre jeho štruktúru, ktorá je takisto využiteľná v procese objavovania vzťahov. Oboznámili sme sa s dvomi hlavnými prístupmi pre objavovanie vzťahov z textu – štatistickým a lingvistickým. Porovnali sme ich výhody a nevýhody a rozhodli sme sa preskúmať štatistický prístup, ktorý je menej používaný v doméne vzdelávania.

Navrhli sme metódu pre objavovanie vzťahov z výučbového obsahu, ktorú sme úspešne integrovali do systému pre správu výučbového obsahu. Pri objavovaní vzťahov podobnosti sme využili latentnú sémantickú analýzu. V prípade hierarchických vzťahov sme navrhli dva varianty objavovania založené na existujúcich metódach [8, 33].

Dôvodom voľby štatistického prístupu bola jeho nezávislosť od jazyka textu. Na rozdiel od lingvistického prístupu nevyžaduje štatistický prístup hlbšie skúmanie textu a navrhovanie vzorov a pravidiel pre objavenie hierarchických vzťahov. Preto môže byť integrovaný do systému pre správu výučbového obsahu a aplikovaný na akýkoľvek výučbový obsah.

Zrealizovali sme dôkladné overenie metódy na adaptívnych kurzoch Funkcionálneho a Logického programovania z výučbového systému ALEF. Výsledky našej metódy sme porovnali so zlatým štandardom aj s existujúcim lingvistickým prístupom [35], čím sme opísali vlastnosti metódy v danej doméne. Najlepšie výsledky mali presnosť 31 % a úplnosť 53 %. Metóda poskytla lepšie výsledky pri väčšom korpuse dokumentov. V porovnaní s lingvistickým prístupom mala naša metóda menšiu presnosť a úplnosť, ale presnosť s počtom vygenerovaných vzťahov bola stabilnejšia. Naša metóda dokázala nájsť také vzťahy, ktoré neboli objavené lingvistickým prístupom. Dôležitým vstupom metódy pre objavovanie hierarchických vzťahov sú vzťahy medzi termami a dokumentmi. Metóda poskytla lepšie výsledky, keď mala k dispozícii viac týchto vzťahov a boli definované učiteľom ako keď boli automaticky vygenerované.

Výsledky boli ovplyvnené fázou predspracovania. Lematizácia bola limitovaná len na slová, ktoré mala dostupné v databáze. Tie pokrývali ale len hovorový jazyk a nie doménovo špecifické pojmy. Preto sme ju museli dopĺňať. Predspracovanie by sa dalo ešte vylepšiť vyriešením anafor a odkazujúcich výrazov, aby boli lokalizované všetky výskyty termov v korpuse.

Lepšie výsledky by mohli byť dosiahnuté vylepšením spôsobu získavania vzťahov podobnosti medzi termami. Latentná sémantická analýza neobjavila vzťahy medzi všetkými termami, medzi ktorými bol hierarchický vzťah v zlatom štandarde, a preto nemohol byť identifikovaný ani metódou na objavenie hierarchických vzťahov.

Ďalším návrhom na vylepšenie je vyfiltrovanie nesprávne určených hierarchických vzťahov na základe váhy hierarchických vzťahov. Tá je momentálne počítaná iba počas latentnej sémantickej analýzy ako kosínusová podobnosť termov. Zavedením nového spôsobu počítania váhy hierarchických vzťahov by bolo možné obmedziť nesprávne určené hierarchické vzťahy.

Niektoré termy neboli objavené v texte, pretože ich učiteľ definoval v doménovom modeli v inom tvare ako sa vyskytujú v texte. Ak by namiesto latentnej sémantickej analýzy bol použitý spôsob získavania vzťahov medzi termami, ktorý by nevyžadoval výskyt relevantných doménových termov v texte, mohol by ich učiteľ definovať ľubovoľne. Prípadne by mohli byť všetky termy definované nie len názvom, ale aj skupinou tvarov, v ktorých sa nachádzajú v texte.

V neposlednom rade boli výsledky ovplyvnené aj kvalitou samotného zlatého štandardu vytvoreného len malou skupinou doménových expertov. A tak sa mohlo stať, že naša metóda objavila správny vzťah, ale nenachádzal sa v zlatom štandarde a tým sa znížila presnosť a úplnosť metódy.

Hlavnými prínosmi našej práce sú:

- zmapovali sme možnosti objavovania vzťahov z výučbového obsahu,
- prispeli sme do oblasti menej využívaných štatistických metód pre objavovanie vzťahov,
- naša metóda je vhodná ako doplnok existujúceho lingvistického prístupu [35], pretože dokázala objaviť odlišné vzťahy ako lingvistický prístup,
- vďaka výhodám štatistického prístupu oproti lingvistickému mohla byť naša metóda integrovaná do systému pre správu výučbového obsahu, kde môže pomáhať učiteľom pri tvorbe doménového modelu,
- naša práca je prínosom pre rozrastajúcu sa oblasť automatizovaného získavania doménového modelu.

Použitá literatúra

[1] BRIN, S., PAGE, S.: *The anatomy of a large-scale hypertextual Web search engine*. In: WWW7, 1998, s. 107-117.

[2] CEGLOWSKI, M., COBURN, A. CUADRADO, J.: *Semantic search of unstructured data using contextual network graphs*. In: 2003.

[3] CIMIANO, P.: *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*. Springer, 2006, 347 s.

[4] COLE, S., VALTORTA, M., ROYAL, M. et al.: *A Lightweight Tool for Automatically Extracting Causal Relationships from Text*. In: SoutheastCon, 2006, s. 125-129.

[5] CVITAS, A.: *Relation extraction from text documents*. In: MIPRO, 2011, s. 1565-1570.

[6] DAVIDOV, D., RAPPOPORT, A., KOPPEL, M.: *Fully Unsupervised Discovery of Concept-Specific Relationships by Web Mining*. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, 2007, s. 232-239.

[7] DEY, L., ABULAISH, M., SHARMA, J. et al.: *Text Mining through Entity-Relationship Based Information Extraction*. In: Proceedings of the 2007 IEEE/WIC/ACM International Conference on Web Intelligence and Inteligent Agent Technology, 2007, s. 177-180.

[8] DIEDERICH, J., BALKE, W.: *The Semantic GrowBag Algorithm: Automatically Deriving Categorization Systems*. In: ECDL, 2007, s. 1-13.

[9] DOLOG, P., HENZE, N., NEJDL, W. et al.: *The Personal Reader: Personalizing and Enriching Learning Resources using Semantic Web Technologies*. In: Proc. of the AH 2004, 2004, pp. 85-94.

[10] DRYMONAS, E., ZERVANOU, K., PETRAKIS, E. G. M.: *Unsupervised ontology acquisition from plain texts: the OntoGain system*. In: NLDB, 2010, s. 277-287.

[11] EL BACHARI, E., ABDELWAHED, E., EL ADNANI, M.: *Design of an Adaptive E-Learning Model Based on Learner's Personality*. In: UbiCC Journal, vol. 5, 2010, no. 3, pp. 1-8.

[12] EL BACHARI, E., ABDELWAHED, E., EL ADNANI, M.: *E-Learning Personalization Based on Dynamic Learners' Preference*. In: International Journal of Computer Science & Information Technology, vol. 3, 2011, no. 3, pp. 200-216.

[13] FELDMAN, R., SANGER, J.: *The Text Mining Handbook*. Cambridge University Press, 2006, 410 s.

[14] FRANTA, M., GAJDOŠ, M., HABDÁK, M. et al.: *Management of Lighweight Semantic Content for an Adaptive Web-Based (Learning) Portal*. In: IIT.SRC, 2012, s. 495-496.

[15] FÜRST, F., TRICHET, F.: *Heavyweight Ontology Engineering*. In: On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops, 2006, s. 38-39.

[16] GIUNCHIGLIA, F., ZAIHRAYEU, I.: *Lightweight Ontologies*. In: Encyclopedia of Database Systems, 2009, s. 1613-1619.

[17] GRUBER, T. R.: *A translation approach to portable ontology specifications*. In: Knowledge Acquisition, vol. 5, 1993, no. 2, pp. 199-220.

[18] HARRIS, Z. S.: *Mathematical Structures of Language*. 1. vyd. New York: Wiley, 1968. 230 s. ISBN 0470353163.

[19] HEARST, M. A.: *Automated discovery of WordNet relations*. In: WordNet: An Electronic Lexical Database, 1998, s. 131-153.

[20] HENZE, N., DOLOG, P., NEJDL, W.: *Reasoning and ontologies for personalized e-learning in the semantic web*. In: Educational Technology & Society, vol. 7, 2004, pp. 82-97.

[21] HEYER, G., LÄUTER, M., QUASTHOFF, U. et al.: *Learning Relations using Collocations*. In: IJCAI Workshop on Ontology Learning, 2001.

[22] CHURCH, K. W., HANKS, P.: *Word association norms, mutual information, and lexicography*. In: Computational Linguistics, vol. 16, 1990, no. 1, pp. 22-29.

[23] JIANG, X., TAN, A.: *CRCTOL: A semantic-based domain ontology learning system*. In: Journal of the American Society for Information Science and Technology, vol. 61, 2010, no. 1, pp. 150-168.

[24] LINDBERG, D. A., HUMPHREYS, B. L., MCCRAY, A. T.: *The Unified Medical Language System*. In: Methods of Information in Medicine, vol. 32, 1993, no. 4, pp. 281-291.

[25] LUO, X., YAN, K., CHEN, X.: *Automatic Discovery of Semantic Relations Based on Association Rule*. In: Journal of Software, vol. 3, 2008, no. 8, pp. 11-18.

[26] MÄDCHE, A., STAAB, S.: *Discovering Conceptual Relations from Text*. In: European Conference on Artificial Intelligence, 2000, s. 321-325.

[27] MÄDCHE, A., VOLZ, R.: *The ontology extraction & maintenance framework: Text-to-onto*. In: ICDM, 2001.

[28] MIHÁL, V.: *Objavovanie vzťahov vo výučbovom texte na základe poznámok vytvorených používateľmi*. Bratislava: FIIT STU, 2011. Diplomová práca.

[29] MIZOGUCHI, R.: *Part 1: introduction to ontological engineering*. In: New Generation Computing - Quantum computing, vol. 21, 2003, no. 4, pp. 365-384.

[30] MORALIYSKI, R., DOUCET, A., AHONEN-MYKA, H.: *Automatic Discovery of Word Semantic Relations using Paraphrase Alignment and Distributional Lexical Semantics Analysis*. In: Natural Language Engineering, vol. 16, 2010, no. 4, pp. 347-358.

[31] NJIKE-FOTZO H., GALLINARI, P.: *Learning «Generalization/Specialization» Relations between Concepts – Application for Automatically Building Thematic Document Hierarchies*. In: RIAO, 2004, s. 143-155.

[32] PANDIT, S., HONAVAR, V.: *Ontology-guided Extraction of Complex Nested Relationships*. In: ICTAI, 2010, s. 173-178.

[33] SANDERSON, M., CROFT, B.: *Deriving concept hierarchies from text*. In: SIGIR, 1999, s. 206-213.

[34] SPILIOPOULOU, M., MÜLLER, R. M., BRUNZEL, M. et al.: *Coupling Information Extraction and Data Mining for Ontology Learning in PARMENIDES*. In: RIAO, 2004, s. 156-169.

[35] ŠIMKO, M.: *Automated Acquisition of Domain Model for Adaptive Collaborative Web-based Learning*. Bratislava: FIIT STU, 2012. 172 s. Dizertačná práca.

[36] ŠIMKO, M., BIELIKOVÁ, M.: *Automatic Concept Relationships Discovery for an Adaptive E-Course*. In: EDM, 2009, s. 171–179.

[37] ŠIMKO, M., BARLA, M., BIELIKOVÁ, M.: *ALEF: A Framework for Adaptive Web-based Learning 2.0*. In: Reynolds, N., Turcsányi-Szabó, M. (Eds.): KCKS 2010, IFIP Advances in Information and Communication Technology, vol. 324, 2010, pp. 367-378.

[38] USCHOLD, M., GRUNINGER, M.: *Ontologies and semantics for seamless connectivity*. In: SIGMOD Record, vol. 33, 2004, no. 4, pp. 58-64.

[39] VOJTEK, P., BIELIKOVÁ, M.: *Vhodnosť lokálneho ohodnocovania grafu v sociálnej sieti obchodného registra SR*. In: WIKT, 2009, s. 65-68.

[40] WONG, W., LIU, W., BENNAMOUN, M.: *Ontology Learning from Text : A Look back and into the Future*. In: ACM Computing Surveys, 2011, 36 s.

[41] WEBER, G.: *Episodic learner modeling*. In: Cognitive Science, vol. 20, 1996, no. 2, pp. 195-236.

[42] WEBER, G., BRUSILOVSKY, P.: *ELM-ART: An Adaptive Versatile System for Web-based Instruction*. In: International Journal of Artificial Intelligence in Education, vol. 12, 2001, pp. 351-384.

[43] WHITE, S., SMYTH, P.: *Algorithms for estimating relative importance in networks*. In: KDD, 2003, s. 266-275.

[44] YAMAGUCHI, T.: *Acquiring Conceptual Relationships from Domain-Specific Texts*. In: IJCAI Workshop on Ontology Learning, 2001, pp. 0-2.

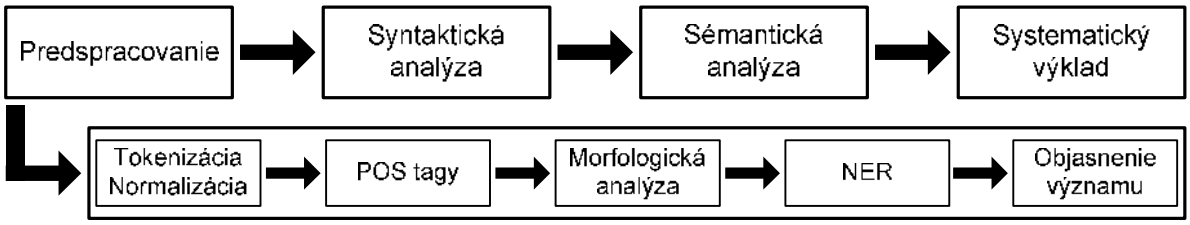
Prílohy

Príloha A: Spracovanie prirodzeného jazyka

Spracovanie prirodzeného jazyka (angl. *Natural Language Processing*) predstavuje súbor techník, ktoré slúžia na prevod textu v prirodzenom jazyku, ktorý každodenne používame, do strojom spracovateľného formátu, na ktorý sa dajú následne aplikovať metódy objavovania znalostí. Proces spracovania prirodzeného jazyka (Obr. 17) je založený na princípe prekladača.

Skladá sa zo štyroch fáz:

- lexikálna analýza,
- syntaktická analýza,
- spracovanie sémantiky,
- generovanie cieľového výstupu.



Obr. 17. Štandardný tok pri spracovaní prirodzeného jazyka [3].

Lexikálna analýza

V tejto fáze prebieha predspracovanie textu. Súčasťou predspracovania môžu byť činnosti ako [3]:

- *tokenizácia* – rozdelenie textu na tzv. tokeny, nájdenie hraníc slov a viet, problémy môžu predstavovať napr. skratky ukončené bodkou alebo viacslovné názvy,
- *normalizácia* – prevod časov a dátumov do štandardného tvaru, odstránenie skratiek, špeciálnych znakov, interpunkcie,
- *odstránenie tzv. stop slov* – vyhodenie slov, ktoré nemajú skoro žiadny význam, resp. neovplyvňujú význam textu ako celku,
- *označenie slovných druhov* (angl. *POS – Part-of-speech tagging*) – označenie tokenov slovným druhom, napr. sloveso, podstatné meno a pod.,
- *lematizácia* – prevod slov na základný tvar,
- *stemming* – transformácia na slov na ich kmeň (odstránenie prefixov a sufixov slov),
- rozpoznávanie názvov (angl. *NER - Named Entity Recognition*),
- zjednotenie slov označujúcich tú istú entitu, napr. slová v skrátenom tvare, synonymá, zámená (angl. *Anaphora resolution*).

Predspracovanie slovenského jazyka sa líši od predspracovania anglického jazyka, preto lexikálna analýza môže obsahovať činnosti typické pre slovenský jazyk, napr. odstránenie diakritiky, lematizácia.

Syntaktická analýza

Vo fáze syntaktickej analýzy sa identifikujú syntaktické jednotky. Patria sem metódy:

- *segmentovo-úrovňové* (angl. *chunking* alebo *shallow parsing*) - rozdelenie vstupu na menšie „kúsky“ na základe zoskupovania slov podľa syntaktických pravidiel (implementované ako regulárne výrazy alebo konečný automat),
- *tokenovo-úrovňové* (angl. *parsing* alebo *deep parsing*) - rozbor vstupu na členy podľa (väčšinou bezkontextovej) gramatiky, výpočtovo náročnejšie ako predchádzajúca metóda [3].

Kombinácia POS značenia a rozboru vety (angl. *deep parsing*) poskytuje syntaktické štruktúry a informácie o závislostiach potrebné pre ďalšiu lingvistickú analýzu pri objavovaní konceptov a vzťahov. Využíva sa v mnohých nástrojoch pre spracovanie prirodzeného jazyka [10, 23, 27].

Spracovanie sémantiky a generovanie cieľového výstupu

V tejto fáze sú na syntaktickú štruktúru aplikované metódy pre objavovanie znalostí. Výstupom spracovania sémantiky je logická forma dát, napríklad odvodené syntaktické závislosti, vektorový model, súbor dát vyhodnotenými metrikami ako frekvencia alebo pravdepodobnosť výskytu. [3].

Konečnú fázu predstavuje generovanie znalostí, zo sémantickej reprezentácie vstupných dokumentov a prezentácia výsledkov.

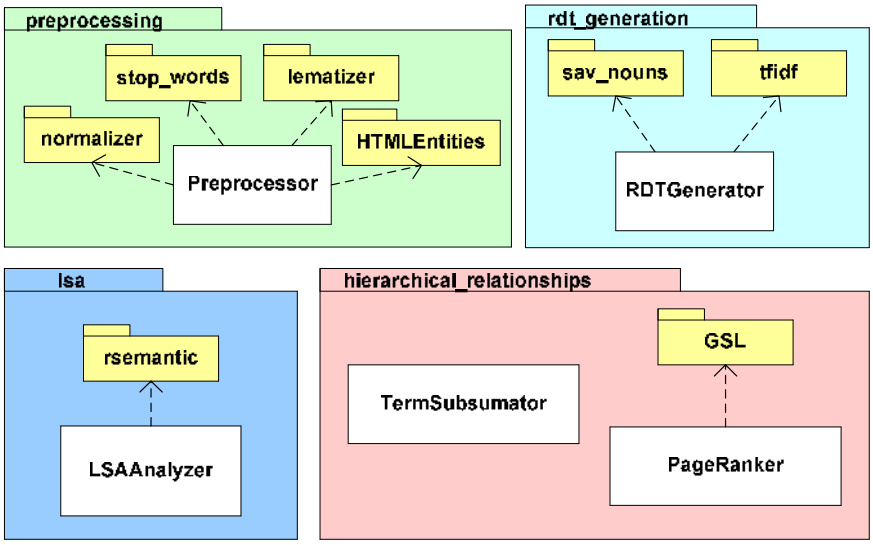
Príloha B: Technická dokumentácia

V technickej dokumentácii uvádzame opis implementácie, spôsob predspracovania dokumentov a opis integrácie implementovanej metódy do systému COME^{2}T.

B.1 Implementácia

Metóda pre objavovanie vzťahov je implementovaná vo forme súboru služieb. Naše riešenie je ľahko integrovateľné. Služby sú rozdelené podľa krokov našej metódy a je ich možné využiť aj oddelene, napr. len službu pre predspracovanie slovenského textu alebo službu pre objavenie vzťahov. Služby sú zobrazené na diagrame balíkov (Obr. 18). Implementovali sme aj službu pre objavenie relevantných doménových termov, hoci pri overovaní sme využili už existujúce relevantné doménové termy. Súbor služieb je ľahko doplniteľný o ďalšie, ktoré napr. implementujú iné metódy pre objavovanie vzťahov, termov alebo iné spôsoby predspracovania textu. Keďže naša metóda je postavená na štatistických prístupoch, pri implementácii iného spôsobu predspracovania textu dokumentov a extrakcie termov by ju bolo možné použiť aj na iné jazyky ako slovenčina.

Služby sme sa rozhodli implementovať v jazyku Ruby, pretože systém na správu výučbového obsahu je implementovaný pomocou frameworku Ruby on Rails.



Obr. 18. Architektúra riešenia. Žltou farbou sú znázornené služby, ktoré sme nevytvorili, ale len použili v našej implementácii.

B.1.1 Predspracovanie textu

Pri predspracovaní sme využili služby pre predspracovanie slovenského jazyka vytvorené na predmete Tímový projekt v ak. roku 2011/2012 pod vedením D. Chudej. Služby sú napísané v jazyku Ruby.

- *normalizer* – normalizácia, teda odstránenie interpunkcie a špeciálnych znakov z textu,
- *stop_words* – odstránenie stop slov, využíva zoznam slovenských stop slov,
- *lematizer* – lematizácia, teda uvedenie slovenských slov do základného tvaru,
- *sav_nouns* – identifikovanie podstatných mien, využíva korpus z JÚĽŠ SAV,
- *tfidf* – počítanie metriky tf-idf pre korpus dokumentov.

Predspracovanie sme implementovali v triede *Preprocessor* a skladalo sa z nasledujúcich krokov:

- *odstránenie príkladov so zdrojovým kódom* – z XML obsahu dokumentov sme pomocou regulárneho výrazu odstránili všetky časti označené tagom *programlisting,*
- *odstránanie tagov* – regulárnym výrazom sme odstránili všetky XML tagy,
- *dekódovanie HTML entít* – použili sme gem *HTMLEntities*,
- *normalizácia* – využili sme existujúcu službu *normalizer*, ktorú sme upravili tak, aby sme neodstránili špeciálne znaky predstavujúce relevantné doménové termy (napr. =.., <, #'),
- *odstránenie stop slov* – využili sme službu *stop_words*,
- *lematizácia* – využili sme službu *lematizer,* ktorá lematizuje podľa databázy slovenských slov; my sme doplnili databázu o slová, ktoré obsahovali relevantné doménové termy a nenachádzali sa ešte v databáze,
- *vyfiltrovanie dokumentov podľa tf-idf* – použili sme službu *tfidf* na identifikovanie najvýznamnejších slov pre každý dokument z korpusu, následné sme vymazali z dokumentov všetky slová okrem slov z relevantných doménových termov a prvých 10 najdôležitejších slov dokumentu podľa tf-idf metriky.

Rozhranie triedy *Preprocessor* je:

```
preprocess(text)
```

kde vstupný parameter `text` je slovenský text a výstupom je predspracovaný text.

B.1.2 Extrakcia termov

Implementovali sme aj služby na extrakciu relevantných doménových termov a vytvorenie vzťahov medzi termami a dokumentmi. Túto službu sme však nakoniec nevyužili, pretože sme overovali metódu na existujúcich relevantných doménových termoch priradených ku korpusu. Služba extrahovala termy pre dokumenty iba na základe tf-idf metriky nakoľko extrakcia termov nebola predmetom tejto práce. Na identifikáciu podstatných mien sme použili službu *sav_nouns*, ktorá používa slovenský národný korpus z JÚĽŠ SAV[2]. Overenie ukázalo, že manuálne vytvorené vzťahy medzi relevantnými doménovými termami a dokumentmi sú vhodnejším vstupom pre našu metódu.

Služba bola implemetovaná v triede *RDTGenerator*, jej rozhranie je:

```
generate_rdts(documents, frequency_treshold)
```

kde parameter `documents` je pole predspracovaných slovenských textov a parameter `frequency_treshold` je minimálna hodnota tf-idf metriky, pri ktorej je nájdené slovo považované za relevantný doménový term. Štandardnú hodnotu sme experimentálne stanovili na 0,6. Výstupom súbor relevantných doménových termov a súbor vzťahov medzi termami a dokumentmi.

B.1.3 Objavovanie vzťahov medzi termami

Pri objavovaní vzťahov medzi termami sme najprv potrebovali vygenerovať vzťahy podobnosti pomocou latentnej sémantickej analýzy. Službu na získanie vzťahov podobnosti sme implementovali

[2] http://korpus.juls.savba.sk/

v triede *LSAAnalyzer*. Na latentnú sémantickú analýzu sme využili gem *rsemantic*[3]. Vstupom služby je korpus predspracovaných dokumentov a relevantné doménové termy. Služba vyhľadá výskyty relevantných doménových termov v texte, zostrojí ich okolia (vektory slov) a na tieto okolia je aplikovaná latentná sémantická analýza. Následne na základe výsledku LSA (kosínusové podobnosti vektorov) sú vytvorené vzťahy medzi termami reprezentovanými okoliami s hodnotou podobnosti ako váhou vzťahu. Služba vráti všetky vzťahy s váhou väčšou ako 0,5.

Rozhranie triedy *LSAAnalyzer* je:

```
generate_relationships(documents, rdts, context_radius)
```

kde parameter `documents` je pole predspracovaných textov, parameter `rdts` je pole relevantných doménových termov a parameter `context_radius` je počet slov naľavo a napravo od relevantného doménového termu v texte, ktoré majú byť považované za jeho okolie. My sme pri overení používali napr. hodnotu 5, t.j. okolie termu tvorilo desať slov. Výstupom je dvojdimenzionálne pole predstavujúce maticu vzťahov medzi termami. Rozmery matice korešpondujú s veľkosťou vstupného poľa `rdts`. Prvkami matice sú buď váha vzťahu, ak sa medzi termami nachádza vzťah alebo 0.

V ďalšom kroku objavovania vzťahov sú zo vzťahov pdobnosti identifikované hierarchické vzťahy. Dva varianty metódy na objavovanie hierarchických vzťahov opísané v kapitole 5.3 sú implementované v triedach *TermSubsumator* (množinový variant) a *PageRanker* (PageRank variant). PageRank algoritmus s prioritami bol implementovaný pomocou matíc. Využili sme knižnicu GSL[4].

Rozhranie triedy *TermSubsumator* je:

```
find_hierarchical(rdts_matrix, rdt_document_hash, rdts, documents, w)
```

kde parameter `rdts_matrix` predstavuje maticu vzťahov pdobnosti medzi termami. Jej veľkosť korešponduje s veľkosťou poľa `rdts`, čo je pole relevantných doménových termov. Parameter `documents` je pole predspracovaných textov. Parameter `rdt_document_hash` je hash objekt obsahujúci vzťahy medzi termami a dokumentmi v tvare :

`rdt_document_hash['term'][index] =` váha vzťahu

kde `index` je index dokumentu v poli `documents`. Parameter `w` je minimálna váha vzťahov z matice `rdts_matrix`, pre ktoré sa bude zisťovať, či sú hierarchické. Štandardnú hodnotu sme stanovili 0,9.

Rozhranie triedy *PageRanker* je:

```
find_hierarchical(rdts_matrix, rdt_document_hash, document_hash, rdts, document_count, w)
```

kde parametre `rdts_matrix`, `rdt_document_hash`, `rdts` a `w` sú rovnaké ako v predchádzajúcom rozhraní. Parameter `document_count` predstavuje počet dokumentov v korpuse a parameter `document_hash` je hash objekt uchovávajúci vzťahy medzi dokumentmi v tvare :

`document_hash[index][index]` = váha vzťahu

kde `index` je index dokumentu v poli predspracovaných textov, z ktorých bol vytvorený hash objekt.

[3] https://github.com/josephwilk/rsemantic
[4] Ruby interface to GNU Scientific Library: http://rubyforge.org/projects/rb-gsl/

B.2 Integrácia do systému COME^{2}T

Pred implementáciou sme zaznamenali nasledujúci prípad použitia. Následne sme navrhli používateľské rozhranie zobrazené na obrázku (Obr. 20).

UC01: Automatické vygenerovanie vzťahov medzi relevantnými doménovými termami

Kontext použitia: Učiteľ si vygeneruje vzťahy medzi termami, aby mohol manuálne doupravovať vzniknutý doménový model.

Vstupná podmienka: Existuje vytvorený doménový model s relevantnými doménovými termami priradenými ku korpusu dokumentov.

Výstupná úspešná podmienka: Existujúci doménový model je doplnený o automaticky vygenerované vzťahy medzi termami alebo je vytvorený úplne nový doménový model s vygenerovanými vzťahmi.

Výstupná neúspešná podmienka: Používateľom požadovaná operácia sa nevykonala. Existujúce dáta ostali v stave pred začatím operácie.

Hlavný úspešný tok:

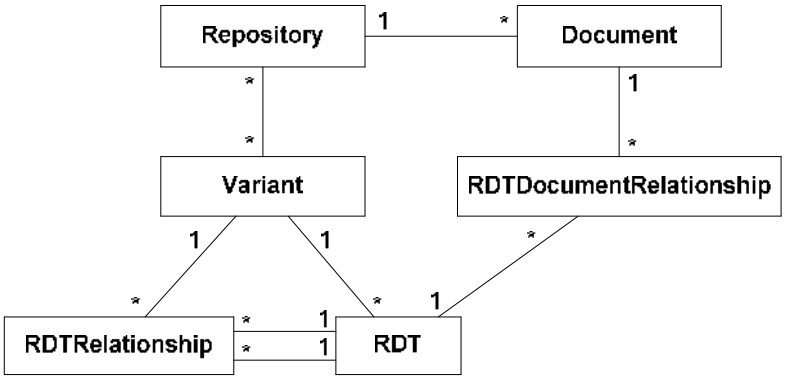
1. Používateľ označí repozitár a zvolí, že chce automaticky vygenerovať vzťahy medzi termami.
2. Aplikácia zobrazí formulár (Obr. 20).
3. Používateľ zvolí, z ktorého z variantov metadát priradených repozitáru majú pochádzať relevantné doménové termy, medzi ktorými sa budú vytvárať vzťahy.
4. Používateľ zvolí, že chce vzťahy uložiť do vybraného variantu metadát
5. Používateľ zvolí, parametre metódy na objavovanie vzťahov: veľkosť okolia termov v texte, parameter *w*; alebo ponechá štandardné hodnoty.
6. Používateľ zvolí, že chce vygenerovať vzťahy medzi termami so zadanými parametrami.
7. Aplikácia začne proces generovania vzťahov a informuje používateľa o stave procesu.
8. Prípad použitia končí.

Rozšírenia:

4a. Používateľ chce vzťahy uložiť ako nový variant metadát:
    4a1. Používateľ zvolí, že chce vzťahy uložiť ako nový variant metadát.
    4a2. Používateľ zadá meno nového variantu metadát.
    4a3. Prípad použitia pokračuje bodom 5 hlavného toku.

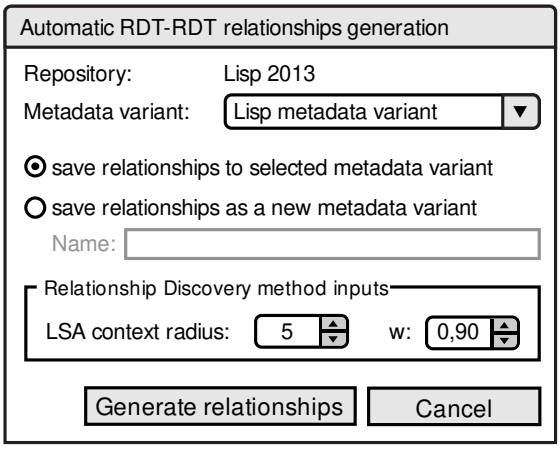
7a. Používateľ zadal zlé hodnoty parametrov:
    7a1. Aplikácia zobrazí chybové hlásenia s informáciou o chybách.
    7a2. Prípad použitia pokračuje bodom 3 hlavného toku.

Systém COME^{2}T [14] je postavený pomocou frameworku Ruby on Rails. Relevantná časť logického modelu systému je zobrazená na obrázku (Obr. 19). Entita *Repository* reprezentuje adaptívny výučbový kurz, t.j. obsahuje dokumenty. Ku kurzu je priradený jeho doménový model, tzv. variant metadát skladajúci sa z relevantných doménových termov a vzťahov medzi nimi (na obrázku entita *Variant*). Entity v dátovom modeli zodpovedajú triedam v aplikácii (modelom podľa vzoru MVC, na ktorom je založený framework Ruby on Rails).



Obr. 19. Časť logického dátového modelu systému COME^{2}T.

V rámci integrácie nami vytvorených služieb sme v aplikácii doplnili triedu *Repository* o metódu na automatické generovanie doménového modelu. Úlohou metódy je transformovať dáta z databázy (dokumenty z repozitára, relevantné doménové termy z priradeného variantu metadát a vzťahy medzi termami a dokumentami) do formátu, ktorý na vstupe potrebujú nami vytvorené služby na objavovanie vzťahov. Ďalej v metóde nasleduje volanie samotných služieb.



Obr. 20. Návrh používateľského rozhrania pre generovanie vzťahov medzi relevantnými doménovými termami.

Príloha C: Overenie – doplňujúce výsledky

V tejto prílohe sú uvedené doplňujúce výsledky experimentov získané pri overovaní metódy na objavovanie hierarchických vzťahov.

C.1 Vzťahy zlatého štandardu

V nasledujúcej tabuľke (Tab. 9) sú uvedené najlepšie dosiahnuté výsledky pri aplikácii metódy na vzťahy zo zlatého štandardu. Na kurze Funkcionálneho programovania dosahoval množinový variant metódy najlepšie hodnoty F_T pri aplikácii na vzťahy s váhou väčšou ako $w = 0{,}94$ a najlepšie hodnoty F''_T pre $w = 0{,}97$, PageRank dosahoval najlepšie hodnoty F-metriky pri $w = 0{,}97$. Na kurze Logického programovania dosahovala metóda najlepšie hodnoty F_T aj F''_T pre vzťahy s váhou väčšou ako $w = 0{,}75$.

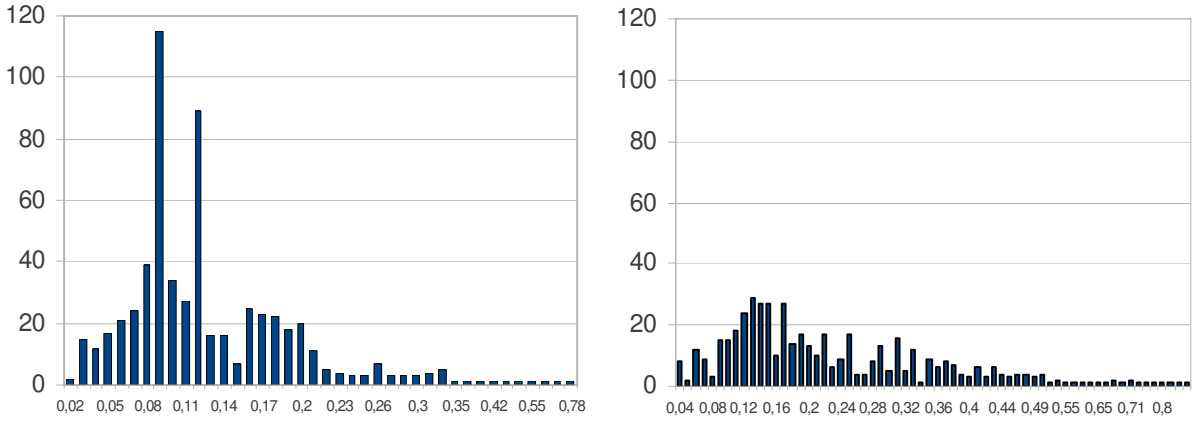
Tab. 9. Najlepšie výsledky overenia na vzťahoch zlatého štandardu (FP – Funkcionálne programovanie, LP – Logické programovanie, obj. – počet objavených vzťahov, spr. – počet správnych vzťahov).

	w	*vzťahy*	*množinový variant*							*PageRank variant*						
			obj.	*spr.*	P_T	R_T	F_T	R''_T	F''_T	*obj.*	*spr.*	P_T	R_T	F_T	R''_T	F''_T
FP	0,94	*všetky*	84	80	**0,94**	**0,37**	**0,53**	0,48	0,63	72	43	0,94	0,28	0,44	0,42	0,58
		related-to	3	0	0	0	0	0	0	3	0	0	0	0	0	0
		rel.-to+is-a	83	79	0,94	0,36	0,52	0,47	0,63	72	43	0,94	0,28	0,44	0,42	0,58
		prerequisity	1	1	1	0	0,01	1	1	0	0	0	0	0	0	0
	0,97	*všetky*	82	79	**0,95**	0,36	0,53	**0,48**	**0,64**	71	43	**0,95**	**0,28**	**0,44**	**0,42**	**0,58**
		related-to	2	0	0	0	0	0	0	2	0	0	0	0	0	0
		rel.-to+is-a	82	79	0,95	0,36	0,53	0,48	0,64	71	43	0,95	0,28	0,44	0,42	0,58
		prerequisity	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LP	0,75	*všetky*	92	50	**0,52**	**0,32**	**0,40**	**0,37**	**0,43**	22	12	**0,50**	**0,06**	**0,11**	**0,50**	**0,50**
		related-to	0	0	0	0	0	0	0	1	0	0	0	0	0	0
		rel.-to+is-a	59	53	0,83	0,35	0,49	0,65	0,73	14	12	0,93	0,06	0,12	0,54	0,68
		prerequisity	43	0	0	0	0	0	0	8	0	0	0	0	0	0

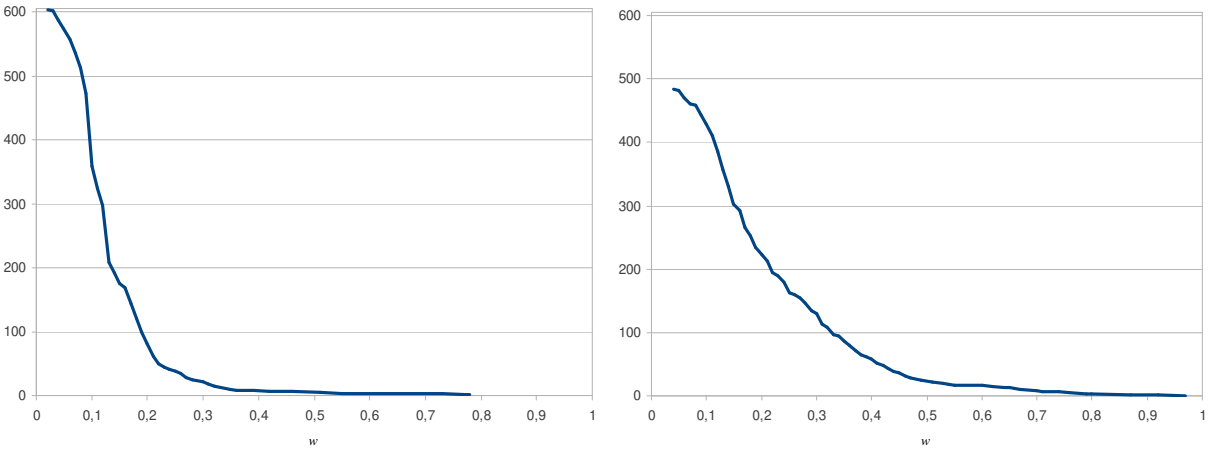
C.2 Automaticky vygenerované vzťahy podobnosti

Keďže vzťahy z automaticky vygenerovaných modelov neobsahovali skoro žiadne vzťahy s váhou väčšou ako $w = 0{,}9$, preskúmali sme súbor vzťahov podrobnejšie. Vytvorili sme histogram, aby sme zistili rozloženie prvkov v súbore podľa váhy (Obr. 21) a načrtli sme závislosť počtu vzťahov, na ktoré by bola aplikovaná metóda od parametra w (Obr. 22).

Najlepšie výsledky boli dosiahnuté pri hodnote parametra $w = 0{,}66$ (12 vzťahov) pre Funkcionálne programovanie a $w = 0{,}19$ (81 vzťahov) pre Logické programovanie. Výsledky sú uvedené v kapitole Overenie (kapitola 6.3.2).



Obr. 21. Početnosti vzťahov podľa váhy v automaticky vygenerovaných vzťahoch podobnosti. (Logické programovanie vľavo, Funkcionálne programovanie vpravo)

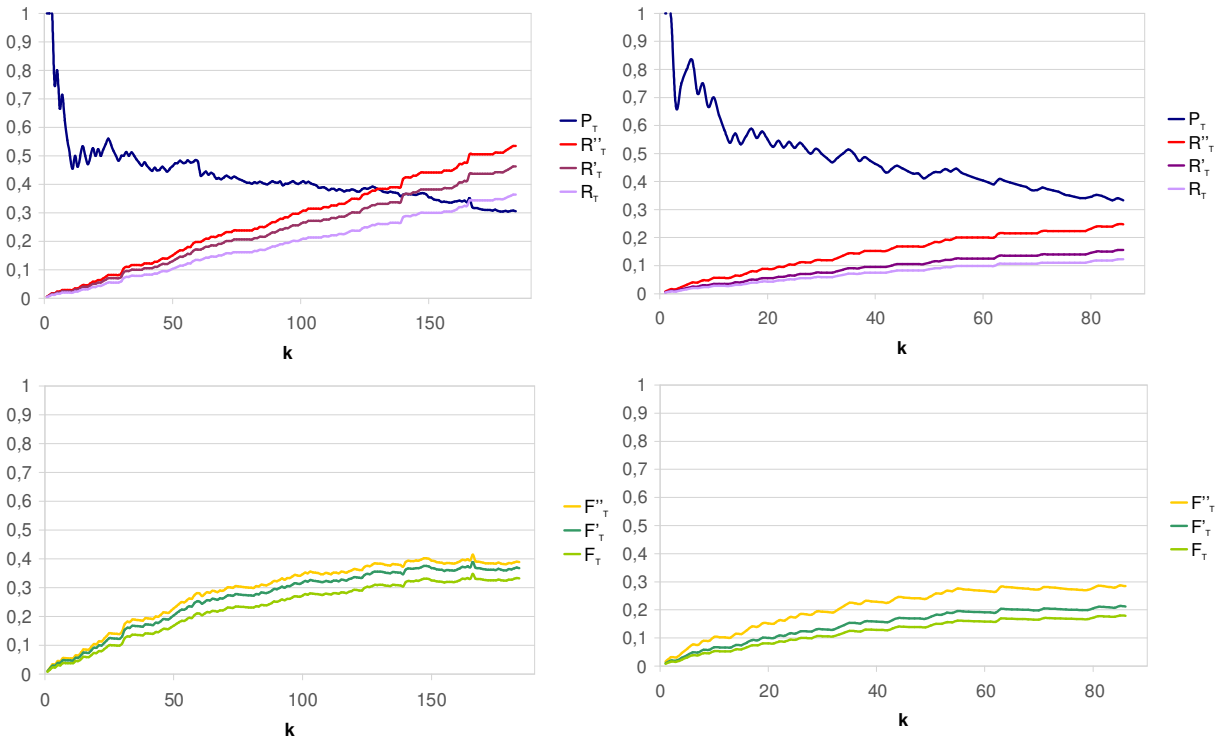


Obr. 22. Závislosť počtu vzťahov, na ktoré by bola aplikovaná metóda na objavovanie hierarchických vzťahov od parametra *w*. (Logické programovanie vľavo, Funkcionálne programovanie vpravo)

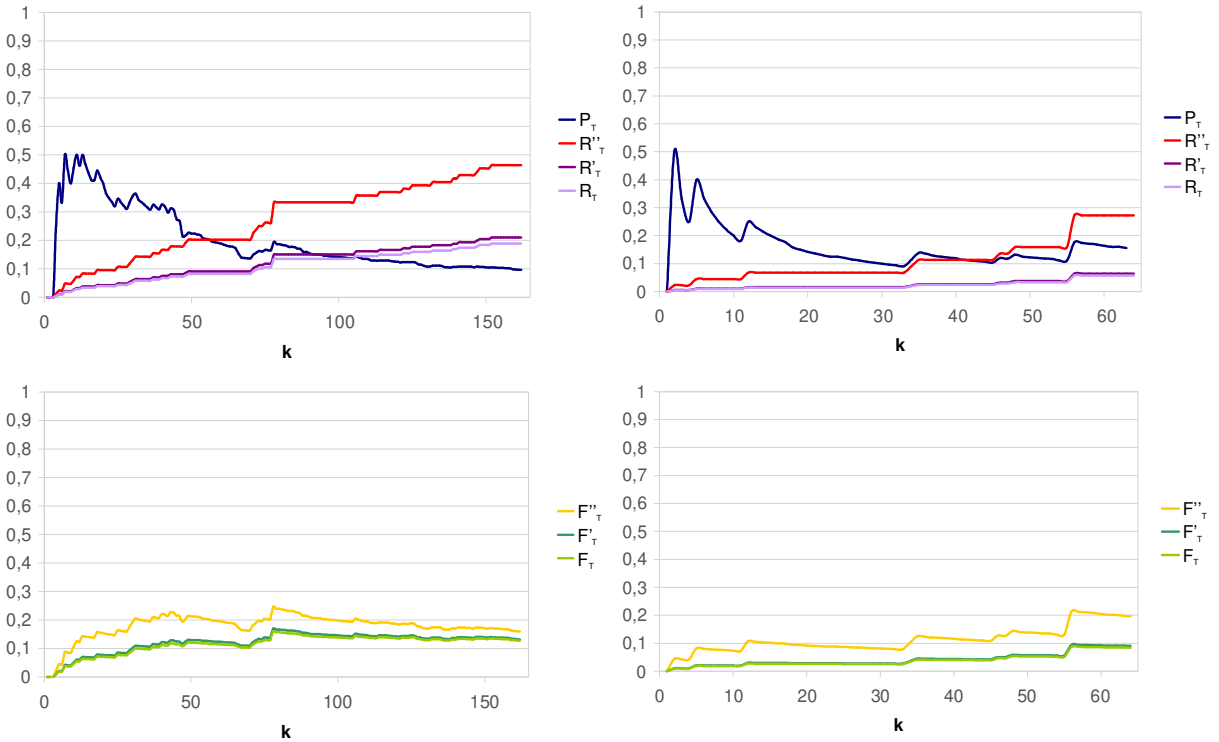
C.3 Vzťahy vygenerované latentnou sémantickou analýzou (LSA)

Na grafoch (Obr. 23, Obr. 24) uvádzame pre doplnenie závislosť presnosti a úplnosti od počtu vygenerovaných vzťahov aj pre ostatné varianty ich počítania. Môžeme vidieť rozdiely v úplnosti v prípade, že sa počíta len vzhľadom na relevantné doménové termy objavené v texte (R'_T), vzhľadom na všetky relevantné doménové termy (R_T) alebo vzhľadom len na termy, medzi ktorými sme objavili hierarchický vzťah (R''_T). Obdobne sú znázornené aj všetky F-metriky (F_T, F'_T, F''_T).

Grafy zaznamenávajú výsledky pre oba kurzy (Funkcionálne a Logické programovanie) z oboch variantov metódy (množinový a PageRank) pri aplikovaní na vzťahy s váhou väčšou ako $w = 0{,}9$.



Obr. 23. Presnosť, úplnosť (hore) a F-metrika (dole) prvých *k* vygenerovaných vzťahov pre kurz Funkcionálneho programovania. Vľavo je množinový variant metódy, vpravo PageRank variant ($w = 0{,}9$).



Obr. 24. Presnosť, úplnosť (hore) a F-metrika (dole) prvých *k* vygenerovaných vzťahov pre kurz Logického programovania. Vľavo je množinový variant metódy, vpravo PageRank variant ($w = 0{,}9$).

Príloha D: Príspevok prijatý na IIT.SRC 2013

Tento príspevok bol prezentovaný na študentskej vedeckej konferencie IIT.SRC 2013, ktorá sa konala 23. 4. 2013. Príspevok bol uverejnený v zborníku:

Vrablecová, P.: Relationship Discovery from Educational Content. In: Proc. of the 9th Student Research Conference in Informatics and Information Technologies (IIT.SRC 2013), STU Bratislava, 55-60.

Relationship Discovery from Educational Content

Petra VRABLECOVÁ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova 3, 842 16 Bratislava, Slovakia
petra.vrablecova@gmail.com

Abstract. The domain model is an essential part of adaptive learning system. It expresses the semantics of educational content in the form of metadata. We consider it to be a lightweight ontology, i.e., a set of terms and relations. Manual domain model building is a challenging task for teachers, hence there is an effort to automate it. We propose a method for *automated* acquisition of metadata from educational content, aimed at relationships discovery between terms. We exploit existing methods for relationship discovery from text and adopt them for the educational domain. Our work is promising contribution to the growing field of automated domain model acquisition.

1 Introduction

Abstraction, modularization or building of hierarchies are basic tools for human beings to understand, classify, categorize all sorts of things regardless of complexity. We try to achieve this kind of thinking in machines, too, so the cooperation with them is as meaningful, helpful and efficient for us as possible. To accomplish this behavior we have to supply machines with knowledge humans are able to acquire by modalities and common sense – semantics.

Our work focuses on the area of education, specifically *adaptive learning*. Adaptive learning system stores the semantics of its educational content in a domain model. The domain model is needed as a basis for tracking users' progress in learning and adaption of the content accordingly. It is represented by metadata, in our case a lightweight ontology consisting of set of relevant domain terms (RDT) and relationships between them. Terms represent the semantics of the educational content, which is presented in form of learning objects such as explanations, exercises. E.g., a chapter about file handling would have assigned terms like "write" or "read". RDTs are interconnected by various types of relationships, e.g., is-a, related-to, type-of. Manual creation of the complete and correct domain model is a demanding task for the educational content author (teacher). There are attempts to automate it. Many generic methods for automatic acquisition of metadata have been developed by now. But too few methods focus on area of education. We explored existing approaches, took note of educational content specifics and designed a method for discovery of relationships between RDTs. Our work aims to facilitate the process of domain model acquisition.

---

* Master degree study programme in field: Software Engineering
Supervisor: Dr. Marián Šimko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 55–60.

2 Related work

There are many generic methods for automatic acquisition of metadata from text. They usually focus only on a part of metadata like terms, concepts, or relationships between them. Natural language processing is widely used in these methods. We focus on methods for relationship discovery. There can be distinguished hierarchical and non-hierarchical relationships. There are two main approaches to relationship discovery – statistical and linguistic.

Statistical methods are based on data mining and machine learning algorithms. Their main advantage is the language independence. On the other hand, to provide good results a big dataset is needed. To discover non-hierarchical relationships the distributional hypothesis (e.g., LSA method) and collocations of words in text are mostly used. Techniques for discovery of hierarchical relationships are usually based on clustering [2], e.g. latent Dirichlet allocation (LDA) [16], formal concept analysis (FCA) [3]. Term subsumption is used to discover hierarchical relationships based on conditional probability of term occurrence in corpus [11].

Linguistic methods are the most often used methods. They depend on the language of the text and require at least basic knowledge about its syntax. But they can provide better results because the discovery rules can be tailored for certain cases of relationship occurrence in text. Techniques based on syntactic dependencies like verb frames [2] and lexical-syntactic patterns [7] or usage of semantic dictionaries like WordNet [10] can be used for both types of relationships.

Only a few works deal with automatic acquisition of metadata for adaptive systems. The authors of MOT adaptive system present method for acquisition of relationships between concepts [4]. Relationship between concepts is created and labeled according to their most common attribute. In [12] is described a method for automatic prerequisite and outcome relationships identification between concepts extracted from a sequentially ordered set of learning objects on C programming language. The disadvantage is the necessity of sequential order of learning objects because it digresses from traditional book or a tree structure of e-courses. An interesting example is the adaptive vocabulary acquisition system ELDIT [1], where methods and techniques of natural language processing were employed in order to create relationships between examples of vocabulary entries and vocabulary entries. The OBAMA-tool [13] aggregates the most of existing freely available tools, techniques and procedures to achieve the semiautomatic building of domain model. The WordNet dictionary is used for relationship identification.

The tool CourseDesigner [14] uses for relationship discovery a vector approach similar to LSA. It is the first tool that also considers the structure of educational content – concepts assigned to learning objects and applies graph algorithms (spreading activation, PageRank algorithm) to improve the results of vector approach. A method for automated hierarchical relationship discovery using the linguistic approach was presented in [15]. This work relies on the specifics of educational content – high occurrence of explanation and determination phrases.

In our work we decided to take advantages of the less explored statistical approach – language independence, no need for syntax knowledge. We assume that educational content has an unambiguously defined narrowed vocabulary which is a precondition for better results of statistical methods. We will make use of the LSA, term subsumption and application of graph algorithms on the educational content's structure to discover relatedness and hierarchical relationships. Our method is described in detail in the next section. We employ the system for educational content management for the evaluation.

3 Relationship discovery

The educational content of adaptive learning system consists of set of learning objects (LO) – any entity, digital or non-digital, that may be used for learning, education or training [9]. We consider mainly text documents. They usually form a hierarchy (a tree or a book structure), i.e., are linked through LO-LO relationships implying their relatedness. The purpose of our method is the auto-

mated creation of a lightweight ontology which will be assigned to the set of LO. A lightweight ontology is considered to be a set of relevant domain terms (RDT) and relationships of different types between them. RDTs are assigned to LOs through RDT-LO relationship that implies the semantic connection between the term and the content of the LO (e.g., the term is a keyword).

Relationship discovery process (see Figure 1) consists of three steps: (a) LO preprocessing, (b) extraction of relevant domain terms, (c) discovery of relationships between terms.

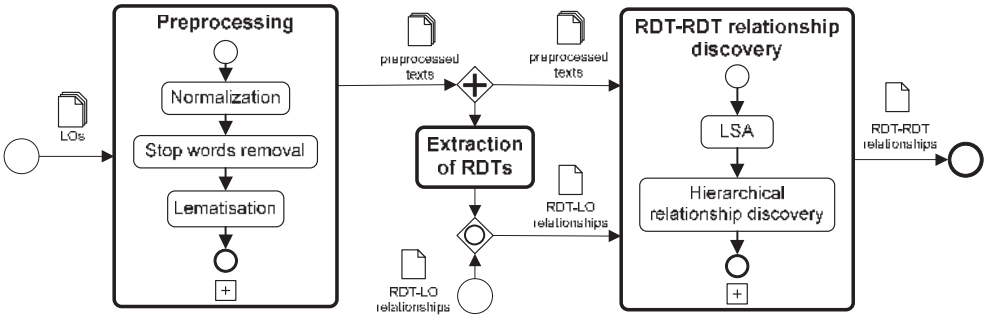


Figure 1. Relationship discovery process.

The input of preprocessing is a set of learning objects. The text of LO is normalized, purged from stop words and lemmatized. The preprocessing procedure depends on the language of the learning objects. Since our relationship discovery method is language independent, in case of other language of learning objects, only the preprocessing procedure needs to be replaced.

Besides preprocessed texts a set of RDTs and RDT-LO relationships are needed. We extract both from preprocessed texts using the standard approach based on tf-idf measure. There can be also used already existing data; the only limitation is the necessity of terms' occurrence in text.

Relationship discovery begins with the construction of a net of RDTs. In this step LSA is applied on the preprocessed texts and assigned RDTs. LSA computes the context similarity of terms, i.e., the similarity of words that surrounds a term in the text. If the similarity is significant, a relationship is created. The output is a set of relationships between related terms. For example in course on programming relationships between terms "function" and "print" or "human" and "user" would appear because these words occur in similar contexts in the learning objects.

LSA is usually used for discovering synonyms in text, therefore we assume that there might exist a hierarchical (is-a) relationship between very related terms. In the next step we propose two variants to determine whether the found relationship is hierarchical.

3.1 Hierarchical relationship discovery based on term subsumption

This variant follows the original work on term subsumption [11] that claims the existence of hierarchical relationship between terms that collocate in documents with the probability of at least 80 %. Then the term which occurs in more documents is labeled as the more general, i.e., superordinate. Since the relationship created by LSA can exist between terms that does not collocate in the same document, in our method we compare sets of learning objects to which are terms assigned (see Figure 2). The procedure for the pair of terms x and y can be described by following steps:

1. Get a set of learning objects with assigned term x.
2. Add the learning objects with assigned terms that are in strong LSA relationship with term x.
3. Construct the set of learning objects from steps 1 and 2 for term y.
4. Compare the sets. If the sets are not disjoint then label the term related to the bigger of sets as superordinate.

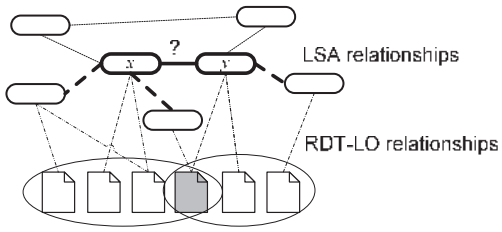


Figure 2. Determination of relationship type between terms x and y.

3.2 Hierarchical relationship discovery using PageRank algorithm

In this variant we apply the PageRank algorithm with priors on the graph of RDT-LO relationships and LO-LO relationships. The set of LO-LO relationships is another input for this method. The procedure of relationship identification between terms *x* and *y* consists of following steps:

1. Apply the PageRank algorithm on the graph with the learning objects assigned to term *x* as the starting nodes for the algorithm and get the sorted list of ranked terms.
2. Repeat the step 1 with the learning objects assigned to term *y* as the starting nodes.
3. Cut the lists' tails and keep only best-ranked terms (first *k* %).
4. Compare the lists. If both *x* and *y* are in both lists and if *x* is on higher position in both lists then label *x* as superordinate.

This technique is based on the Semantic GrowBag algorithm [5].

4 Evaluation

The goal of the evaluation is to find out whether the domain model built by our method is on the level of the manually built domain model. We compare our method result – a lightweight ontology – with the gold standard ontology. To evaluate the hierarchical relationship discovery techniques we compare the results of the algorithms they are based on with our results. The part of the evaluation is also an integration of our method to the system for educational content management.

4.1 Dataset

We perform the tests on the learning objects from the Functional and Logic programming course. The ontology from this course is the gold standard created by the group of domain experts – including the author of the course. The characteristics of the course are shown in Table 1.

Table 1. Functional and Logic programming course characteristics.

	Functional programming	Logic programming
# learning objects	79	42
# words	28,455	23,383
average length of learning object	360.19	556.74
# relevant domain terms	162	138
average relevant domain term length	1.70	1.41
average number of relevant domain terms per learning object	1.94	2.10

4.2 Experiments

In experiments we use the recall and precision measures against the gold standard. We also use these measures for the methods we exploited in our work – term subsumption, Semantic Grow-

back algorithm, to see whether techniques for hierarchical relationship discovery proposed by us give better results. In addition, we modify recall and precision measures with respect to the transitive nature of the hierarchical relationship by following approach in [15].

At the moment we still work on final results. We experiment with various setups of the method and look for the optimal combinations. The best recall and precision are so far 0.59 and 0.08 but we see a scope for further improvement. The preliminary results show that the method has a great potential to supplement methods based solely on linguistic processing (e.g., [15]).

4.3 Integration into COME^{2}T

The part of the evaluation is also the integration of our method into the system COME^{2}T (*COllaboration and MEtadata-oriented COntent Management EnvironmenT*) [6]. Its purpose is the management of the adaptive learning portal's content used to support educational process. This system already contains functionality for non-automated creation of a metadata in a form of lightweight ontology. Integration of our method into the system helps to automate the creation of metadata and ease the work for the authors of content (see Figure 3).

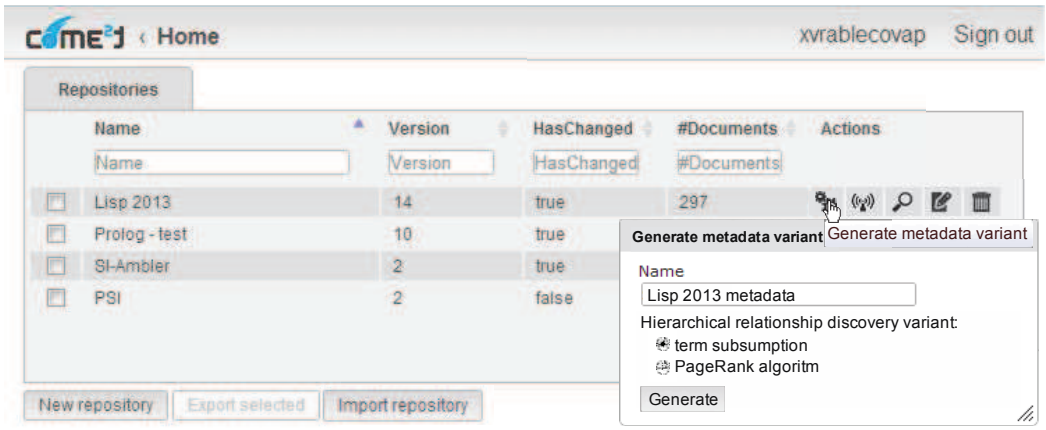


Figure 3. Design of integrated functionality in COME^{2}T system.A repository contains the learning objects from one adaptive educational course.

5 Conclusions

The domain model is important part of the adaptive learning system. It influences the quality of educational content adaptation to the learner. Unfortunately there is not much research on the topic of domain model acquisition and course authoring support. However there are many generic methods for metadata acquisition from text which form a solid basis for research in this area.

In this paper we presented a method for automatic discovery of relationships between terms in a lightweight ontology. We use statistical approach for metadata acquisition from text. The advantage of this approach is its language independency which allows us to apply this method in the future on other than Slovak texts. The uniform vocabulary of educational texts leads to better results of statistical approach. Our method focuses also on the discovery of the hierarchical relationships which comprise the core of the domain model. We took advantage of the specific structure of educational content (hierarchically organized) in this process. The project is at the moment in the phase of evaluation and preliminary results suggest that the most valuable contribution of the method is that it yields different kinds of relationships that cannot be discovered by applying linguistic approaches. As a part of evaluation the method is integrated into the educational content management system to support the course authoring and the automated domain model acquisition.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0971/11.

References

[1] Brusilovsky, P., Knapp, J., Gamper, J.: Supporting teachers as content authors in intelligent educational systems. *Int. J. of Knowledge and Learning*, (2006), vol. 2, no. 3/4, pp. 191-215.

[2] Cimiano, P.: *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*. Springer, (2006).

[3] Cimiano, P., Hotho, A., Staab, S.: Comparing conceptual, divisive and agglomerative clustering for learning taxonomies from text. In: *ECAI 2004: Proc. of the 16th European Conf. on Artificial Intelligence*, IOS Press, (2004), pp. 435–439.

[4] Cristea, A.I., de Mooij, A.: LAOS: Layered WWW AHS Authoring Model and their corresponding Algebraic Operators. In: *WWW 2003: Proc. The 12th Int. World Wide Web Conference*. Alternate Track on Education, Budapest, (2003).

[5] Diederich, J., Balke, W.: The Semantic GrowBag Algorithm: Automatically Deriving Categorization Systems. In: *ECDL 2007: Proc. of the 11th European Conf. on Research and Advanced Technology for Digital Libraries*, LNCS 4675. Springer, Berlin, (2007), pp. 1-13.

[6] Franta, M., Gajdoš, M., Habdák, M. et al.: Management of Lighweight Semantic Content for an Adaptive Web-Based (Learning) Portal. In: *IIT.SRC 2012: Proc. of the 8th Student Research Conference in Informatics and Information Technologies*, Nakladateľstvo STU, (2012), pp. 495-496.

[7] Hearst, M. A.: Automated discovery of WordNet relations. In Fellbaum, Ch., ed.: *WordNet: An Electronic Lexical Database*. The MIT Press, London, (1998), pp. 131-153.

[8] Heyer, G., Läuter, M., Quasthoff, U. et al.: Learning Relations using Collocations. In: *IJCAI 2001: Proc. of the 2nd Workshop on Ontology Learning OL'2001*, (2001).

[9] IEEE LTSC: Draft Standard for Learning Object Metadata. IEEE Standard 1484.12.1. IEEE, (2002), retrieved March 2013.

[10] Lindberg, D. A., Humphreys, B. L., McCray, A. T.: The Unified Medical Language System. *Methods of Information in Medicine*, (1993), vol. 32, no. 4, pp. 281-291.

[11] Sanderson, M., Croft, B.: Deriving concept hierarchies from text. In: *SIGIR 1999: Proc. of the 22nd Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, ACM, (1999), pp. 206-213.

[12] Sosnovsky, S., Brusilovsky, P., Yudelson, M.: Supporting Adaptive Hypermedia Authors with Automated Content Indexing. In: *A^3H 2004*: *Proc. of 2nd Int. Workshop on Authoring of Adaptive and Adaptable Educational Hypermedia*, (2004), pp. 380-389.

[13] Šaloun, P., Velart, Z., Klimanek, P.: Semiautomatic domain model building from text-data. In: *SMAP 2011: Proc. of 6th Int. Workshop on Semantic Media Adaptation and Personalization*, IEEE Computer Society, (2011), pp. 15–20.

[14] Šimko, M., Bieliková, M.: Automated Educational Course Metadata Generation Based on Semantics Discovery. In: *EC-TEL 2009: Proc. of 4th European Conf. on Technology Enhanced Learning*, LNCS 5794. Springer, Berlin, (2009), pp. 99-105.

[15] Šimko, M., Bieliková, M.: Discovering Hierarchical Relationships in Educational Content. In: *ICWL 2012: Proc. of 11th Int. Conf. on Web-based Learning*, LNSC 7558. Springer, Berlin, (2012), pp. 132-141.

[16] Yeh, J., Yang, N.: Ontology construction based on latent topic extraction in a digital library. In: *ICADL 2008: Proc. of the 11th Int. Conf. on Asian Digital Libraries*, LNCS 5362. Springer, Berlin,(2008), pp. 93-103.

Príloha E: Obsah dátového nosiča

Priložené médium obsahuje:

`dp_vrablecova.pdf` – diplomová práca
`iit.src_vrablecova.pdf` – článok ptijatý na IIT.SRC 2013

`/Implementacia/` – adresár s implementovanými službami v jazyku Ruby
`/Implementacia/preprocessing/` – adresár so službou na predspracovanie
`/Implementacia/rdt_generator/` – adresár so službou na generovanie RDT z dokumentov
`/Implementacia/lsa/` – adresár so službou na generovanie vzťahov pomocou LSA
`/Implementacia/hierarchical_relationships/`
– adresár so službami na generovanie hierarchických vzťahov
`/Implementacia/scripts/` – adresár so skriptami použitými na testovanie
`/Implementacia/lemmaformtag.txt` – korpus slovenských slov
`/Implementacia/load_word_database.rake`
– skript na načítanie korpusu slovenských slov do databázy

`/Testovanie/` – adresár s výsledkami získanými počas testovania
`/Testovanie/testy.xls` – súbor so zaznamenanými testami
`/Testovanie/lisp-isa-manual-all-pre-pv.xlsx`
– vzťahy vygenerované lingvistickým prístupom, s ktorými sme porovnávali výstup našej metódy
`/Testovanie/results/`
– adresár s výstupmi testovania použitými pri zostrojovaní grafov a porovnávaní výsledkov
`/Testovanie/results/compare_lingv/`
– výsledky porovnania s lingvistickým prístupom, pre každý test spoločne nájdené vzťahy a odlišné vzťahy
`/Testovanie/results/graph_data/`
– CSV súbory s dátami pre vykreslovanie grafov presnosti, úplnosti, F-metriky pre každý test
`/Testovanie/results/graphs/`
– obrázky a excel súbor s vytvorenými grafmi uvedenými v práci
`/Testovanie/results/optimal w/`
– CSV súbory s vygenerovanou presnosťou, úplnosťou a F-metrikou pre každú hodnotu param. *w*
`/Testovanie/results/rdr/`
– doplnené a spoločné vzťahy medzi dokumentmi a termami získané pri obohacovaní vzťahmi
`/Testovanie/results/relationships/`
– CSV súbory s vygenerovanými vzťahmi pre každý test