VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

SHLUKOVÁNÍ SLOV PODLE VÝZNAMU

WORD SENSE CLUSTERING

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **VILIAM SAMUEL HOŠTÁK**
AUTHOR

VEDOUCÍ PRÁCE — **Doc. RNDr. PAVEL SMRŽ, Ph.D.**
SUPERVISOR

BRNO 2017

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií Akademický rok 2016/2017

Zadání bakalářské práce

Řešitel: **Hošták Viliam Samuel**
Obor: Informační technologie
Téma: **Shlukování slov podle významu**
Word Sense Clustering
Kategorie: Umělá inteligence

Pokyny:

1. Prostudujte metody měření sémantické podobnosti slov.
2. Seznamte se s existujícími jazykovými nástroji, které mohou být použity ke zkvalitnění odhadu sémantického zařazení slova.
3. Shromážděte data potřebná pro průběžné testování jednotlivých fází řešení problému.
4. Na základě získaných poznatků navrhněte a realizujte systém, který dokáže k zadanému slovu automaticky najít a zobrazit slova sémanticky příbuzná
5. Vyhodnoťte realizované řešení a porovnejte zvolený přístup s jinými metodami
6. Vytvořte stručný plakát prezentující práci, její cíle a výsledky

Literatura:

- dle doporučení vedoucího

Pro udělení zápočtu za první semestr je požadováno:

- funkční prototyp řešení

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Smrž Pavel, doc. RNDr., Ph.D.**, UPGM FIT VUT
Datum zadání: 1. listopadu 2016
Datum odevzdání: 17. května 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
612 66 Brno, Božetěchova 2

doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Táto práca sa zaoberá sémantickou podobnosťou slov. Popisuje a porovnáva existujúce modely, ktoré sa aktuálne pre tento účel používajú. Rozoberá návrh a implementáciu vytvoreného systému na predspracovanie textového korpusu, vytváranie sémantických modelov a vyhľadávanie sémanticky príbuzných slov. Vytvorený systém umožňuje prácu s distribučnými sémantickými modelmi Word2vec, FastText a GloVe.

Abstract

This thesis deals with semantic similarity of words. It describes and compares existing models that are currently used for this purpose. It discusses the design and implementation of the system for corpus preprocessing, semantic modelling and retrieval of semantically related words. The system that has been created supports the use of distributional semantic models Word2vec, FastText and Glove.

Kľúčové slová

spracovanie prirodzeného jazyka, sémantická podobnosť, Word2vec, FastText, GloVe, Gensim

Keywords

natural language processing, semantic similarity, Word2vec, FastText, GloVe, Gensim

Citácia

HOŠTÁK, Viliam Samuel. *Shlukování slov podle významu.* Brno, 2017. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Doc. RNDr. Pavel Smrž, Ph.D.

Shlukování slov podle významu

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána doc. RNDr. Pavla Smrža, Ph.D. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....
Viliam Samuel Hošták
16. mája 2017

Poďakovanie

Ďakujem pánovi doc. RNDr. Smržovi, Ph.D. za poskytnutú odbornú pomoc a vedenie práce.

Obsah

1 Úvod **3**

2 Predspracovanie textového korpusu **5**
2.1 Korpus . . . 5
2.2 Tokenizácia . . . 5
2.3 Stemovanie . . . 6
2.4 Lematizácia . . . 6
2.5 Spracovanie frekventovaných slov . . . 6

3 Distribučná sémantika **7**
3.1 Charakteristika . . . 7
3.2 Reprezentácia slov . . . 8
3.3 Distribučné sémantické modely . . . 8
3.3.1 Modely založené na počte . . . 8
3.3.2 Prediktívne modely . . . 10

4 Architektúra skúmaných modelov **12**
4.1 Word2vec . . . 12
4.1.1 Model Continuous Bag of Words . . . 14
4.1.2 Model Skip-Gram . . . 14
4.1.3 Aproximácie funkcie softmax . . . 15
4.1.4 Hierarchický softmax . . . 15
4.1.5 Negatívne vzorkovanie . . . 16
4.1.6 Podvzorkovanie frekventovaných slov . . . 16
4.2 FastText . . . 16
4.3 GloVe . . . 17

5 Návrh a implementácia **19**
5.1 Požiadavky a použité technológie . . . 19
5.1.1 Použité knižnice . . . 19
5.2 Architektúra systému . . . 20
5.3 Predspracovanie vstupu . . . 21
5.4 Použité korpusy . . . 22
5.4.1 CWC-2011 . . . 22
5.4.2 All.vert . . . 22
5.4.3 Wikipédia . . . 23
5.5 Automatická identifikácia fráz . . . 23
5.6 Trénovanie modelov . . . 24

5.6.1 Word2vec a FastText . . . 24
5.6.2 GloVe . . . 25
5.7 Vyhodnotenie sémantických modelov . . . 26
5.7.1 Slovné analógie . . . 26
5.7.2 Slovník synoným . . . 27
5.7.3 Krycie mená . . . 28
5.8 Sémantická podobnosť slov . . . 29

6 Vyhodnotenie **30**
6.1 Lematizácia . . . 30
6.2 Charakteristika vytvorených modelov . . . 31
6.3 Vyhodnotenie Wikipédie . . . 32
6.4 Vyhodnotenie All.vert . . . 34
6.5 Vyhodnotenie CWC-2011 . . . 36
6.6 Spojenie korpusov . . . 38
6.7 Analýza výsledkov . . . 38
6.7.1 Krycie mená . . . 39
6.7.2 Slovník synoným . . . 40
6.8 Zhrnutie výsledkov . . . 41

7 Záver **43**

Literatúra **44**

Kapitola 1

Úvod

Spracovanie prirodzeného jazyka (Natural language processing, NLP) je rozsiahla a rýchlo sa rozvíjajúca oblasť z oborov informatiky, umelej inteligencie a lingvistiky. Pod pojmom prirodzený jazyk sa myslí jazyk, ktorý je používaný pri komunikácií medzi ľuďmi, na rozdiel od umelých jazykov, ako sú programovacie jazyky alebo matematické zápisy. NLP označuje automatickú, počítačom vykonanú analýzu alebo syntézu písaného alebo hovoreného prirodzeného jazyka. Medzi jeho hlavné úlohy patrí napríklad strojový preklad, automatická sumarizácia textov, rozpoznávanie reči alebo text-to-speech [3].

Jedna z oblastí spracovania prirodzeného jazyka je taktiež distribučná sémantika. Táto oblasť bola vďaka svojim atraktívnym vlastnostiam za posledných dvadsať rokov cieľom rozsiahleho výskumu. Jednou z dôležitých vlastností distribučnej sémantiky je to, že vychádza z hypotézy, ktorá tvrdí, že slová nachádzajúce sa v podobnom kontexte zvyknú mať podobný význam. Na základe tejto hypotézy dnes existuje mnoho rôznych distribučných sémantických modelov, ktoré umožňujú určovať význam slov alebo viacslovných fráz. Uplatnenie majú v rôznych úlohách NLP, ako napríklad extrakcia entít, analýza sentimentu, rozpoznávanie pomenovaných entít či vyhľadávanie informácií.

V tejto oblasti existuje viacero výskumov, ktoré sú zamerané na chovanie distribučných sémantických modelov pre anglický jazyk. Pre ďalšie jazyky, ako napríklad čeština, naopak, veľa výskumov vykonaných nebolo. Práve preto je táto práca zameraná na český jazyk. Jej cieľom je porovnanie a vyhodnotenie vybraných distribučných sémantických modelov a popis implementácie systému, ktorý na základe daných modelov určí k zadanému slovu ďalšie sémanticky podobné slová.

Cieľom tejto práce je zoznámiť sa s metódami na určovanie a meranie sémantickej podobnosti slov a implementácia systému, ktorý dokáže k zadanému slovu automaticky nájsť a zobraziť sémanticky podobné slova. Za týmto účelom boli vybrané a použité modely Word2vec, ich rozšírená verzia v podobe knižnice FastText a model GloVe.

Kapitola 2 sa zaoberá rôznymi technikami predspracovania textových korpusov. Tieto techniky sú používané pred samotnou tvorbou sémantických modelov, ktoré dokážu významným spôsobom ovplyvniť. Kapitola 3 je venovaná obecnej charakteristike distribučnej sémantiky ako celku a kategorizácií distribučných sémantických modelov. Kapitola 4 obsahuje podrobnejší popis princípu činnosti sémantických modelov, ktoré boli v tejto práci použité za účelom vytvorenia požadovaného systému. Následne je v kapitole 5 popísaný návrh a implementácia samotného systému. Táto časť obsahuje popis architektúry celého systému, testovacích sád a použitých korpusov. Kapitola 6 je venovaná vyhodnoteniu a porovnaniu vytvorených modelov v závislosti na rôznych vybraných parametroch trénovania.

V záverečnej kapitole sú zhrnuté výsledky a prínos tejto práce, pričom sú v nej taktiež diskutované rôzne návrhy na ďalšie rozšírenia.

Kapitola 2

Predspracovanie textového korpusu

Predspracovanie vstupného neanotovaného textového korpusu, taktiež nazývané ako textová normalizácia, hrá dôležitú úlohu pri technikách určovania sémantickej podobnosti slov. Tento krok dokáže zjednodušiť výsledný model a výrazne ovplyvniť jeho presnosť. Predspracovanie je nutné vykonať pred samotnou tvorbou distribučných sémantických modelov a často sa skladá z niekoľkých operácií ako tokenizácia, lematizácia, stemming či odstránenie bezvýznamových slov.

2.1 Korpus

V dnešnej dobe termín korpus označuje reprezentatívnu kolekciu textov alebo časti textov daného jazyka, dialektu alebo inej podmnožiny jazyka, pričom táto kolekcia je čitateľná strojom. Typicky obsahuje rozsiahlu zbierku vzoriek textu, ktoré pokrývajú rôzne oblasti jazyka za účelom vysokej lingvistickej rozmanitosti. Korpus primárne slúži ako základ pre lingvistickú analýzu a deskriptívnu lingvistiku.

Korpusy sa delia na anotované a neanotované. Neanotovaný korpus reprezentuje jednoduchý, nespracovaný text, ktorý neobsahuje žiadne ďalšie doplňujúce informácie. Anotovaný korpus naopak obsahuje pomocné informácie vo forme kódov a značiek, ktoré reprezentujú napríklad slovný druh či gramatické kategórie. Tieto pomocné informácie slúžia na zvýšenie jeho užitočnosti pri rôznych výskumoch [5].

2.2 Tokenizácia

Tokenizácia je proces konvertovania postupnosti znakov na postupnosť tokenov, ktoré zahŕňajú slová alebo frázy. Výsledné tokeny sú často normalizované tak, aby sa skladali len z malých písmen.

Dôležitou úlohou je správna identifikácia znakov, ktoré oddeľujú jednotlivé tokeny. Obecne sa jedná napríklad o interpunkčne znamienka a biele znaky. V niektorých prípadoch to však nemusí platiť. Príkladom takejto situácie je spojovník, ktorý môže niekedy oddeľovať dva validné tokeny a v iných prípadoch môže byť súčasťou jedného tokenu. Podobná situácia nastáva pri bodke, ktorá môže byť súčasťou skratky, alebo môže plniť funkciu ako koniec vety [7].

Ďalšia problematika, ktorá spadá pod tokenizáciu, je identifikácia viacslovných fráz. V niektorých prípadoch je žiadané identifikovať frázy jedným tokenom, ako napríklad „New_York“, miesto dvoch tokenov „New“ a „York“. Osobitne totiž dané slová majú odlišný

význam a ak chceme túto skutočnosť odraziť vo výslednom sémantickom modeli, musia byť frázy určené v rámci predspracovania.

2.3 Stemovanie

Ďalšia časť predspracovania korpusu je stemovanie. Využitie tejto metódy je voliteľné a jej cieľom je identifikácia kmeňa jednotlivých slov. Kmeň označuje takú časť slova, ktorá vznikne po algoritmickom odstránení prefixov a sufixov [10]. Slová, ktoré sa v korpuse pôvodne vyskytujú v rôznych formách, sú pomocou stemovania normalizované na jednotný tvar, čo vedie k zníženiu počtu unikátnych tokenov.

2.4 Lematizácia

Lematizácia je technika podobná stemovaniu. Na rozdiel od stemovania využíva lematizácia slovníky a morfologickú analýzu a snaží sa odstrániť iba prefixy a sufixy, ktoré vznikli skloňovaním daného slova. Jej cieľom je teda určenie základného, slovníkového tvaru slova, ktorý je označovaný ako lema [1].

Lematizácia a stemovanie dokážu významným spôsobom znížiť pamäťovú náročnosť sémantických modelov, a teda aj urýchliť ich tvorbu. Tieto metódy taktiež zvyšujú presnosť modelov v určovaní sémantiky slov. Často sú využívané práve pri práci s jazykmi, ktoré majú komplexný morfologický systém, ako napríklad čeština. Napriek týmto výhodám ich použitie v niektorých prípadoch nie je vhodné.

2.5 Spracovanie frekventovaných slov

Niektoré slová sa vo vstupných textových korpusoch vyskytujú oveľa častejšie ako ostatné. Podľa Zipfovho zákona je frekvencia slov v korpusoch prirodzeného jazyka nepriamo úmerná ich poradiu v počte výskytov. Slovo s najvyššou frekvenciou sa teda vyskytuje približne dvakrát toľko ako druhé najfrekventovanejšie slovo, trikrát toľko ako tretie najfrekventovanejšie slovo atď. [8]. Práve veľmi časté slová hrajú pre daný jazyk viac syntaktickú ako sémantickú úlohu. Tieto slová teda majú nízku informačnú hodnotu pre určovanie sémantickej podobnosti slov a sú pomerne málo podstatné pre analýzu. Príkladom takýchto slov môžu byť spojky a predložky. Potenciálnym problémom je ich neúmerný vplyv na výsledný sémantický model. Existuje niekoľko spôsobov riešenia tohto problému. V rámci predspracovania sa spomínané slová často filtrujú na základe predpripraveného zoznamu častých slov nazývaného stoplist. Alternatívne riešenie je použitie váhových funkcií na zníženie vplyvu frekventovaných slov pri samotnej tvorbe modelu [16].

Kapitola 3

Distribučná sémantika

V tejto kapitole je analyzovaná tematika zhlukovania slov podľa významu, ktorou sa zaoberá distribučná sémantika. Bližšia pozornosť je venovaná vývoju distribučnej sémantiky, kategorizácie distribučných sémantických modelov a reprezentácií slov, ktorú tieto modely využívajú.

3.1 Charakteristika

Distribučná sémantika je empirická oblasť NLP, ktorá je zameraná na zistenie a modelovanie sémantiky slov na základe analýzy a porovnávania ich distribúcie v rozsiahlych textových korpusoch [20]. V tomto kontexte si môžeme definovať sémantickú podobnosť ako metriku, ktorá určuje vzdialenosť významu dvoch lexikálnych jednotiek [6]. Distribučné sémantické modely sú v dnešnej dobe populárne a úspešné v určovaní sémantickej podobnosti slov vo viacerých úlohách NLP. Práve ich dobré výsledky sú dôvodom, prečo distribučný prístup býva najčastejšie zvoleným prístupom k sémantike v oblasti NLP.

Základom distribučnej sémantiky je distribučná hypotéza, ktorá bola sformulovaná Zellingom Harrisom [9]. Táto hypotéza uvádza, že stupeň podobnosti medzi dvoma jazykovými výrazmi je funkciou podobnosti jazykových kontextov, v ktorých sa dané výrazy môžu objaviť. Z distribučnej hypotézy vyplýva, že analýzou dostatočného počtu kontextov daného slova je možné určiť minimálne niektoré jeho významové vlastnosti. Práve vďaka tomu, že distribučná hypotéza navrhuje takúto jednoduchú a praktickú metódu na odvodenie významu slov, mala veľký dopad na NLP. Význam slov je teda možné určiť automatickou analýzou kontextov slov v obyčajných textových korpusoch, na rozdiel od iných prístupov, ktoré vyžadujú rozdielne zdroje, ako napríklad anotované texty.

Popularita distribučnej sémantiky v priebehu posledných desaťročí rástla aj vďaka dostupnosti rôznych textových korpusov a zlepšenia technologickej úrovne počítačov. Práve vďaka vyššej výkonnosti počítačov a pokroku strojového učenia v posledných rokoch sa stalo možným trénovanie omnoho kvalitnejších modelov na skutočne rozsiahlych korpusoch. V dnešnej dobe existuje mnoho rôznych druhov metód a matematických techník na tvorbu distribučných sémantických modelov, ktoré zachytávajú rôzne typy sémantickej podobnosti a dokážu pracovať s rôznymi lexikálnymi jednotkami ako sú slová, frázy, či dokonca celé vety. Sémantické modely si však do dnešnej doby zachovali aspoň jednu spoločnú vlastnosť, a to tú, že nejakým spôsobom vychádzajú z distribučnej hypotézy [6].

3.2 Reprezentácia slov

Každý matematický systém očakáva ako vstup nejakú formu numerickej informácie. Systémy na spracovanie obrazu a zvuku pracujú so vstupom vo forme vysokodimenzionálnych dátových sád, ktoré sú zakódované ako vektory. V prípade obrazových dát sa môže jednať napríklad o jednotlivé intenzity pixelov a v prípade zvukových dát napríklad o koeficienty spektrálnej hustoty. Takáto reprezentácia dát je pre obraz a zvuk intuitívna a nesie v sebe všetky potrebné informácie na vykonanie rôznych úloh, ako napríklad rozpoznanie objektov alebo reči. V prípade systémov spracovania prirodzeného jazyka sú však slová tradične reprezentované ako unikátne diskrétne atomické symboly. Reprezentácia vo forme unikátnych symbolov je síce pre slová intuitívna, ale neposkytuje systému žiadne užitočné informácie o sémantických vzťahoch, ktoré potenciálne existujú medzi jednotlivými symbolmi [12]. Práve vďaka tomuto je pre určovanie sémantickej podobnosti potrebná odlišná reprezentácia slov, ktorá by odrážala ich podobnosť alebo rozdielnosť. Toto umožňuje reprezentácia slov vo forme vektorov, ktoré sú umiestnené v rámci spojitého vektorového priestoru. Rozdiel reprezentácií slov je zobrazený v tabuľke 3.1.

	Unikátny symbol	Vektor
hotel	ID245	[0.60, -0.10, 0.13, ...]
motel	ID118	[0.70, -0.09, 0.07, ...]
...	...	[...]

Tabuľka 3.1: Porovnanie reprezentácie slov

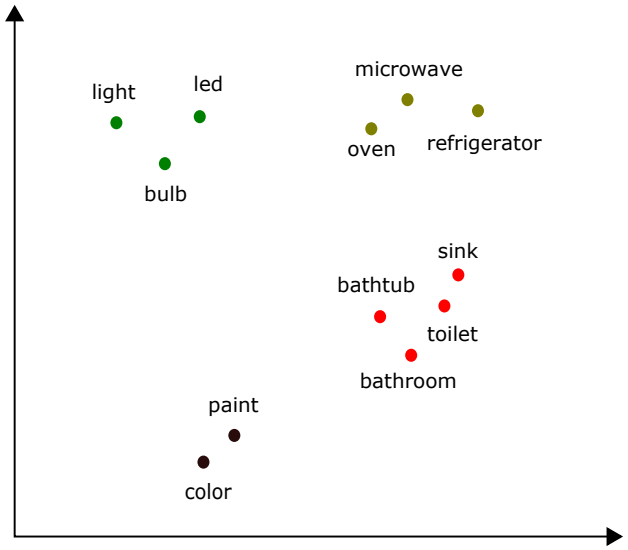
3.3 Distribučné sémantické modely

Distribučné sémantické modely, často taktiež nazývané modely vektorového priestoru alebo modely sémantického priestoru, reprezentujú slová alebo frázy zo vstupného korpusu vo forme vektorov v rámci mnohodimenzionálneho vektorového priestoru, ako je ilustrované na obrázku 3.1.

Tieto vektory sú odvodené na základe analýzy kontextov, v ktorých sa nachádzajú odpovedajúce lexikálne jednotky v korpuse. Distribučné sémantické modely sú preto vysoko závislé na charaktere vstupného korpusu, od ktorého sú vektory odvodené. V súčasnosti sú preferované korpusy, ktoré sú čo najväčšieho rozsahu a po obsahovej stránke heterogénne. Kontext môže v rôznych modeloch odpovedať oknu istého počtu slov, vete, alebo celému dokumentu. Vektory lexikálnych jednotiek aproximujú ich význam, a teda sémanticky podobné slová sa vo vektorovom priestore nachádzajú blízko seba. Dôležitá vlastnosť distribučných sémantických modelov, ktorá z tohto vyplýva, je to, že sémantická podobnosť medzi dvoma alebo viacerými slovami môže byť presne kvantifikovaná z hľadiska geometrickej vzdialenosti medzi vektormi, ktoré ich reprezentujú. V dnešnej dobe existujú rôzne prístupy, ktoré sa dajú rozdeliť do dvoch hlavných kategórií. Tieto kategórie modelov sú označované ako modely založené na počte a prediktívne modely [2].

3.3.1 Modely založené na počte

Modely založené na počte (count-based models) existujú už niekoľko desaťročí a majú bohatú históriu vývoja. Základom tejto kategórie modelov je takzvaná matica spoluvýskytu.



Obr. 3.1: Vektorový priestor s dvoma dimenziami

V tejto matici sú zachytené štatistické informácie o spoluvýskyte, ktoré sú získané analýzou vstupného textového korpusu. Spomínaná matica existuje v dvoch podobách a to matica typu slovo-slovo a matica typu slovo-kontext [15].

V matici spoluvýskytu typu slovo-slovo riadky aj stĺpce reprezentujú jednotlivé slová, ktoré sa nachádzajú v slovníku daného modelu. Je tvorená frekvenciami spoločného výskytu slov v rámci rozsahu kontextového okna o pevnom počte slov, ktoré je postupne posúvané nad vstupným korpusom. Prvok m_{ij} spoluvýskytu matice M teda udáva koľkokrát sa v rámci kontextového okna spolu vyskytovali slová, ktoré reprezentuje riadok i a stĺpec j. Typický predstaviteľ modelov, ktoré používajú maticu typu slovo-slovo je model HAL.

V matici spoluvýskytu typu slovo-kontext riadky reprezentujú slová a stĺpce reprezentujú jednotlivé kontexty zo vstupného korpusu, pričom kontext môže byť veta, odstavec, dokument alebo kontextové okno slov. Prvok m_{ij} matice spoluvýskytu M v tomto prípade udáva koľkokrát sa slovo reprezentované riadkom i nachádzalo v kontexte reprezentovanom stĺpcom j. Najznámejším príkladom modelov, ktoré takúto maticu používajú, je model LSA. Rozdiel týchto typov matíc je ilustrovaný na obrázku 3.2.

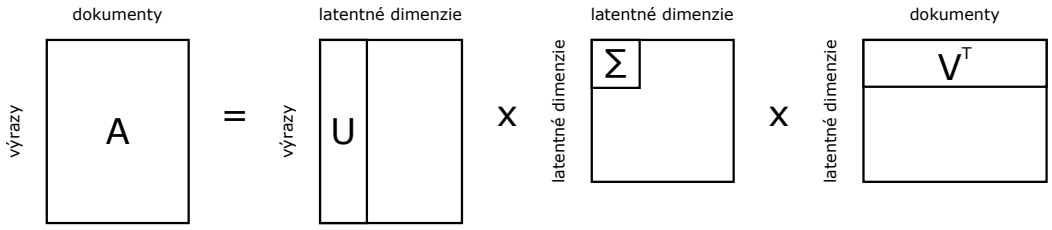
Výrazy	Dokumenty			
	D1	D2	D3	...
pes	6	0	2	...
auto	0	11	4	...
koleso	0	5	0	...
...	...	...	...	...

Výrazy	Výrazy			
	pes	auto	koleso	...
pes	1	0	0	...
auto	0	0	5	...
koleso	0	5	0	...
...	...	...	...	...

Obr. 3.2: Porovnanie matíc spoluvýskytu

Tvorba modelov založených na počte sa skladá typicky z troch krokov. Ako už bolo spomínané, prvým krokom je zber počtu spoluvýskytov analýzou kontextov vo vstupnom korpuse. V druhom kroku sú nazberané frekvencie slov transformované váhovou funkciou na asociačné skóre, čím sa zamedzuje príliš vysokému vplyvu najfrekventovanejších slov. Takéto slová majú často pre daný jazyk primárne syntaktickú, a nie sémantickú rolu, a tento

krok teda vedie k zvýšeniu informatívnosti kontextu [16]. Výsledná matica spoluvýskytu poskytuje váženú explicitnú reprezentáciu distribúcie slov v kontextoch. Takáto matica je však riedka a navyše pri väčších korpusoch vedie k vysokodimenzionálnym vektorom. Ako tretí krok sa preto často pomocou matematických techník vykonáva faktorizácia matice za účelom redukcie dimenzií, ktorá výrazne znižuje výpočtovú a pamäťovú náročnosť modelu a odstraňuje šum. Významná vlastnosť redukcie dimenzií je taktiež to, že umožňuje odhaliť skryté, nepriame vzťahy medzi slovami, ktoré sa priamo nevyskytujú spolu v rovnakom kontexte. Faktorizácia matice spoluvýskytu sa prvýkrát objavila pri modeli LSA, čo tento model v jeho dobe robilo revolučným. Matematická technika, ktorá sa pri modeli LSA používa sa nazýva Singular Value Decomposition a jej princíp je uvedený na obrázku 3.3. Výsledkom tejto transformácie sú latentné sémantické dimenzie, ktoré nahradzujú stĺpce pôvodnej matice. Ďalšie matematické techniky, ktoré sa na redukciu dimenzií najčastejšie používajú, sú napríklad Non-Negative Matrix Factorization alebo Latent Dirichlet Allocation [6].



Obr. 3.3: Singular Value Decomposition

3.3.2 Prediktívne modely

V posledných rokoch sa objavila nová generácia distribučných sémantických modelov, a to skupina nazývaná prediktívne modely (predictive models). Prediktívne modely vychádzajú z neurónových pravdepodobnostných jazykových modelov, ktoré sú používané na inicializáciu komplexnejších metód NLP v rôznych architektúrach založených na neurónových sieťach. Tieto metódy priamo nesúvisia s určovaním sémantickej podobnosti, a preto sa pôvodne ich efektivita v tomto ohľade brala skôr ako zaujímavý vedľajší efekt [2].

Na rozdiel od modelov založených na počte, ktoré sú typické vážením a transformáciami matice spoluvýskytu na základe rôznych kritérií, prediktívne modely fungujú na princípe trénovania neurónových sietí na optimálnu predikciu susedného slova alebo niekoľkých slov. Pri tomto procese sú iteratívne priamo upravované vektorové reprezentácie slov, ktoré majú formu hustých nízkodimenzionálnych vektorov reálnych čísel. Takáto reprezentácia sa často označuje anglickým termínom word embeddings, ktorý je odvodený z faktu, že jednotlivé slová sú vložené do nízkodimenzionálneho lineárneho priestoru, ktorého dimenzie predstavujú latentné charakteristiky slov [6]. Tento spôsob tvorenia distribučných sémantických modelov je v dnešnej dobe populárny vďaka tomu, že sa dá veľmi efektívne aplikovať na rozsiahle textové korpusy a výsledné modely zároveň dosahujú pozoruhodné výsledky v sémantických testoch.

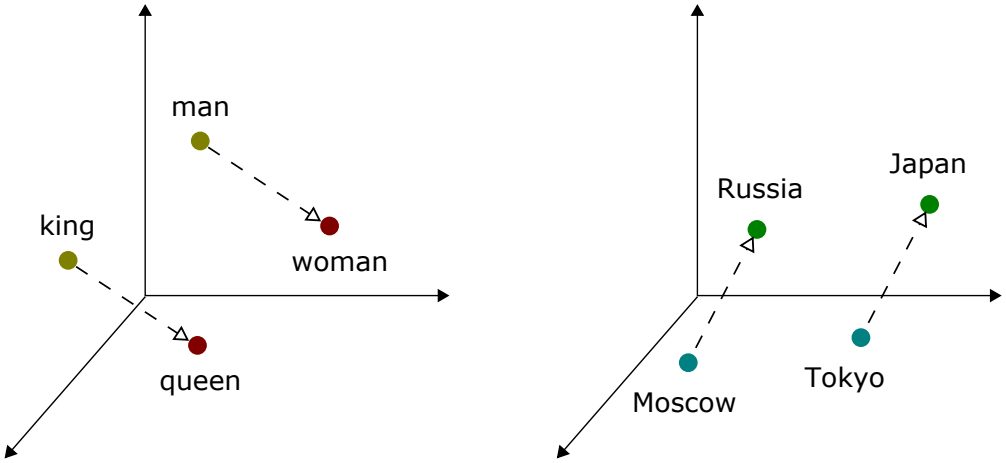
Zaujímavá vlastnosť vektorov slov prediktívnych modelov je ich schopnosť zachytiť nielen sémantickú podobnosť medzi slovami, ale aj rôzne ďalšie lingvistické vzory medzi viacerými dvojicami slov. Podobnosti, ktoré je možné zachytiť, môžu reprezentovať napríklad vzťah pohlavia v dvojiciach „man:woman“ a „king:queen“, či vzťah hlavného mesta v dvo-

jiciach „Russia:Moscow“ a „Japan:Tokyo“, ako je uvedené na obrázku 3.4. Takéto vzťahy odráža rozdiel vektorov slov pre danú dvojicu:

$$vektor(woman) - vektor(man) \approx vektor(queen) - vektor(king) \tag{3.1}$$

Pomocou jednoduchej vektorovej aritmetiky je teda možne zistiť približnú vektorovú reprezentáciu slova a, ktoré má podobný vzťah k slovu b, ako má slovo c ku slovu d. Túto vlastnosť si môžeme demonštrovať na analógií medzi dvojicami slov „man:woman“ a „king:queen“ [11]:

$$vektor(queen) \approx vektor(woman) - vektor(man) - vektor(king) \tag{3.2}$$



Obr. 3.4: Vizualizácia vzťahov v dvoch dimenziách

Kapitola 4

Architektúra skúmaných modelov

V tejto práci je použitých niekoľko úspešných a známych distribučných sémantických modelov, ktoré produkujú kvalitné nízkodimenzionálne vektory slov. Vybrané modely zahŕňajú model GloVe, sadu modelov Word2vec a jej upravenú verziu vo forme knižnice FastText. Hlavný dôvod výberu týchto modelov je ich vysoká presnosť a ich optimalizácia na prácu s rozsiahlymi slovníkmi a analýzu textových korpusov, ktoré obsahujú rádovo až bilióny slov. V tejto kapitole bude popísaná architektúra a princíp fungovania zvolených modelov.

4.1 Word2vec

Word2vec je rodina prediktívnych modelov, ktorá je pomenovaná podľa softwarového balíka, ktorý tieto modely implementuje. Základný princíp týchto modelov je trénovanie neurónovej siete na predikciu cieľového slova na základe susedných slov zo zvoleného kontextového okna, alebo naopak predikciu susedných slov na základe cieľového slova. Natrénovaná neurónová sieť sa však na predikciu už ďalej nepoužíva. Hlavným cieľom tohto procesu je naučenie váh projekčnej vrstvy, ktoré predstavujú samotné hľadané vektory slov. Predpokladá sa totiž, že vektory váh, ktoré sú natrénované na predikciu, dokážu taktiež úspešne reprezentovať význam jednotlivých slov.

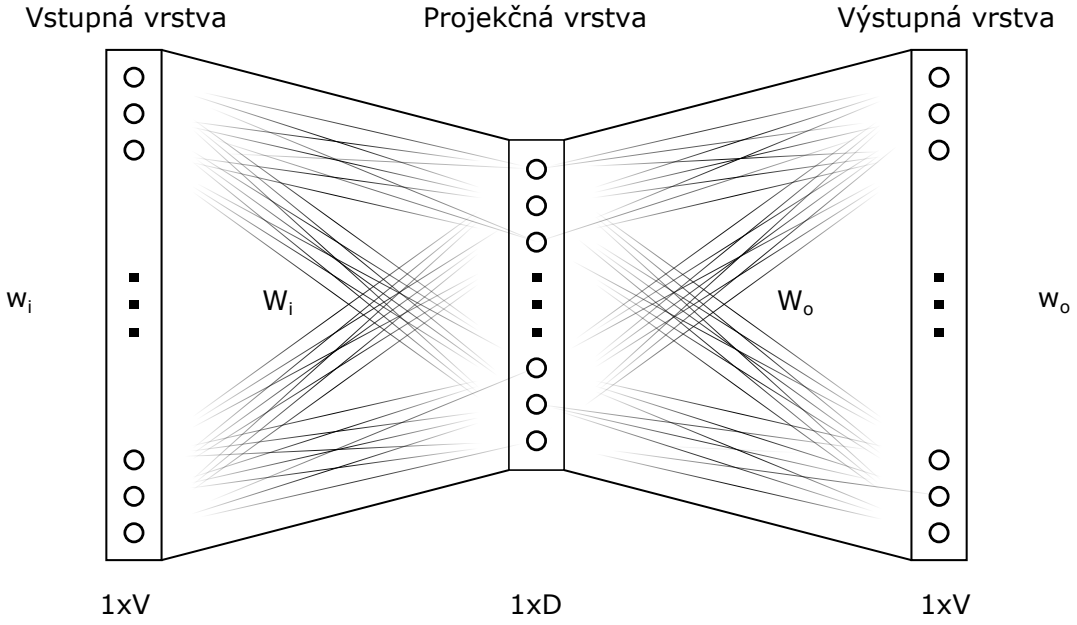
Architektúra použitej neurónovej siete je založená na modeli, ktorý je označovaný ako Feedforward Neural Net Language Model [13]. Použitá sieť je plne spojená a skladá sa zo vstupnej, projekčnej a výstupnej vrstvy. Vstupná vrstva má veľkosť odpovedajúcu počtu slov nachádzajúcich sa v slovníku na trénovanie. V tejto vrstve je vstupné slovo zakódované kódom 1 z V, pričom V odpovedá veľkosti slovníka. Toto znamená, že vstupom do neurónovej siete je vektor x s V dimenziami, v ktorom je dimenzia odpovedajúca pozícií vstupného slova v slovníku nastavená do jednotky, pričom všetky ostatné dimenzie sú nulové.

x_1	x_2	...	x_j	...	x_{V-1}	x_V
0	0	...	1	...	0	0

Obr. 4.1: Vstupný vektor neurónovej siete pre slovo w_j

Počet neurónov v projekčnej vrstve odpovedá zvolenému počtu dimenzií výslednej vektorovej reprezentácie slov. Pri zvolenom počte dimenzií D, môžu byť váhy projekčnej vrstvy reprezentované maticou W_i o veľkosti $V \times D$. Každý riadok tejto matice je D dimenzionálny vektor váh, ktorý reprezentuje odpovedajúce slovo v slovníku. Výstupná vrstva sa skladá z rovnakého počtu neurónov, ako je veľkosť vstupnej vrstvy. Matica váh výstupnej vrstvy

W_o má teda veľkosť $D \times N$. V tomto prípade reprezentujú slová daného slovníka jednotlivé stĺpce matice W_o. Pred samotným trénovaním sú obe matice váh W_i a W_o inicializované na malé náhodné hodnoty.



Obr. 4.2: Zjednodušená architektúra neurónovej siete Word2vec

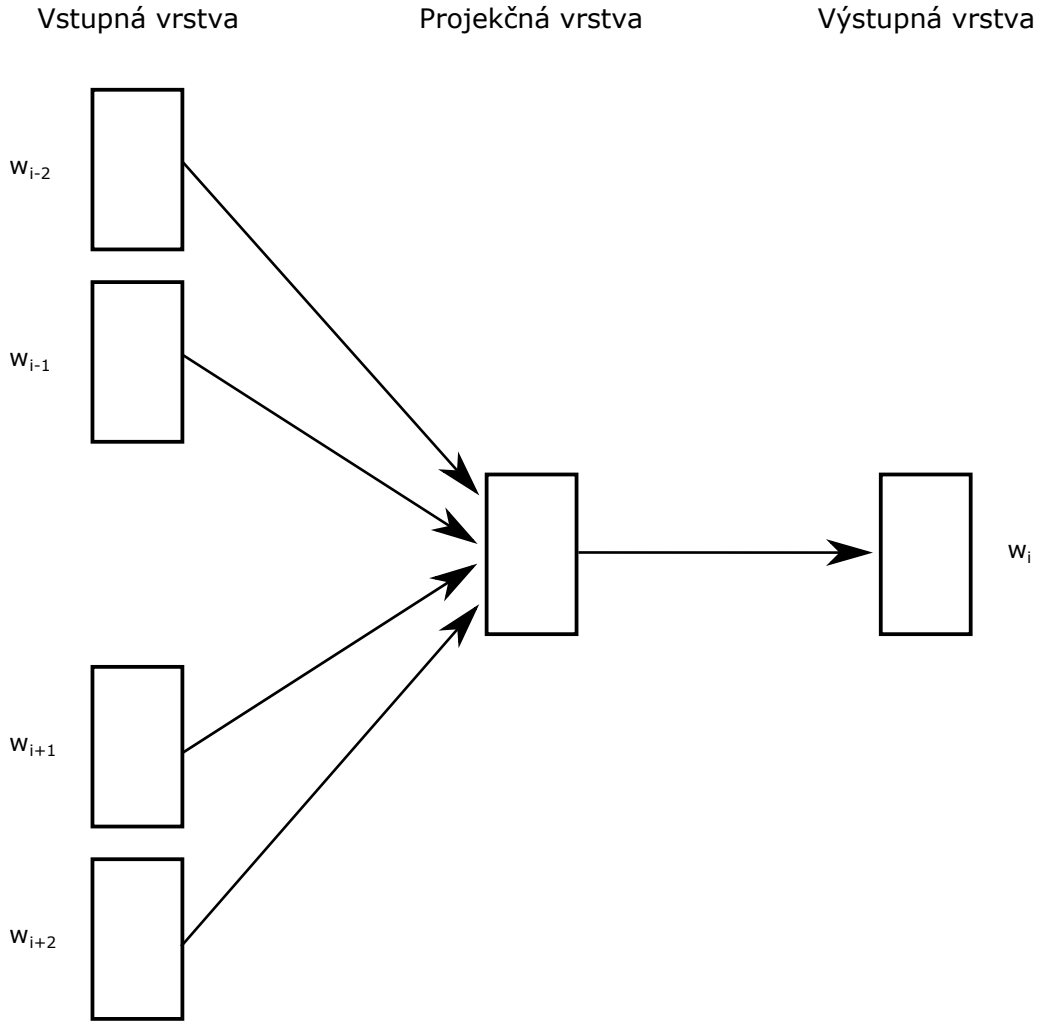
Pre jednoduchosť uvažujme situáciu, kde trénujeme neurónovú sieť na predikciu jedného kontextového slova w_o na základe vstupného slova w_i, ako je vidieť na obrázku 4.2. Cieľom je teda vypočítať pravdepodobnosť $p(w_o \mid w_i)$, ktorá znamená že slovo w_i sa nachádza v kontexte slova w_o. V prvom kroku trénovania je vstupný vektor x odpovedajúci vstupnému slovu w_i vynásobený maticou W_i. Vďaka kódovaniu 1 z V sa touto operáciou vyberie práve jeden riadok z matice W_i. Vybraný vektor váh v_{w_i}, ktorý reprezentuje význam vstupného slova w_i, je priamo odovzdaný do projekčnej vrstvy. Aktivačná funkcia projekčnej vrstvy je teda lineárna. V ďalšom kroku je vektor projekčnej vrstvy vynásobený maticou W_o. Výsledok tohto násobenia je V dimenzionálny výstupný vektor, ktorý udáva skóre podobnosti pre každé z V slov daného slovníka. Toto skóre však predstavuje iba výsledok skalárneho súčinu a nie hľadané pravdepodobnosti. Aby sme získali pravdepodobnosť $p(w_o \mid w_i)$ je nutné normalizovať výsledok skalárneho súčinu medzi vektorom v_{w_i} vstupného slova w_i a vektorom v'_{w_o} kontextového slova w_o. Za týmto účelom sa používa funkcia softmax:

$$p(w_o \mid w_i) = \frac{\exp({v'_{w_o}}^{\mathsf{T}} v_{w_i})}{\sum_{v=1}^{V} \exp({v'_v}^{\mathsf{T}} v_{w_i})} \tag{4.1}$$

V poslednom kroku sa vypočíta chyba oproti cieľu a pomocou spätnej propagácie sa upravia váhy v maticiach W_i a W_o. Pri tomto procese sa používa algoritmus stochastic gradient descent. Uvedená architektúra je zjednodušená verzia dvoch modelov rodiny Word2vec, ktoré sú nazývané Skip-Gram a Continuous Bag of Words [17].

4.1.1 Model Continuous Bag of Words

Model Continuous Bag of Words (CBOW) reprezentuje kontext ako C susedných slov v rámci kontextového okna okolo cieľového slova. Ako je možné vidieť na obrázku 4.3, týchto C slov je použitých ako vstup neurónovej siete pri jej trénovaní na predikciu daného cieľového slova. Hlavný rozdiel oproti už popísanej architektúre spočíva v úprave výpočtu vstupu projekčnej vrstvy. Miesto priameho kopírovania jednotlivých vektorov C vstupných kontextových slov z matice W_i sa dané vektory spriemerujú a tento výsledný vektor je použitý ako výstup projekčnej vrstvy [13].

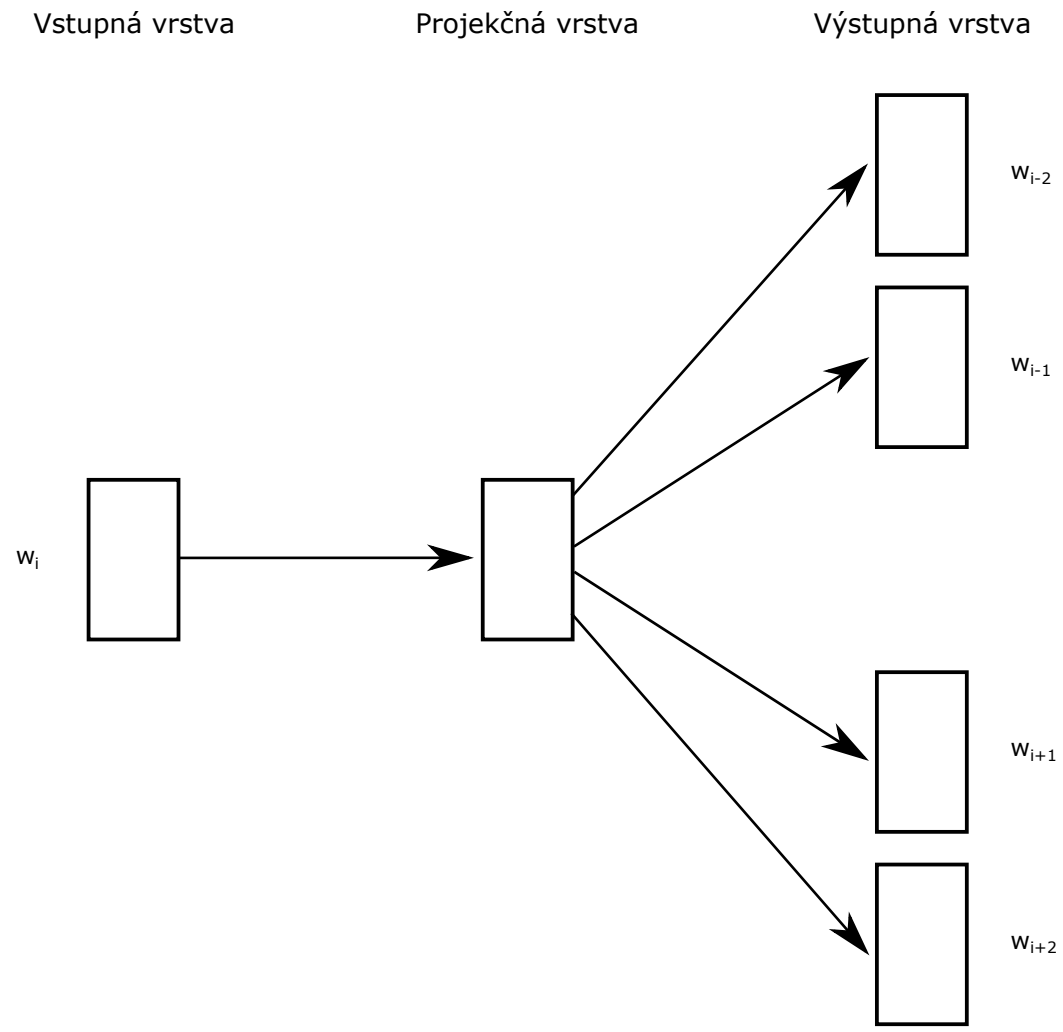


Obr. 4.3: Architektúra modelu CBOW

4.1.2 Model Skip-Gram

Model Skip-Gram (SG) funguje na opačnom princípe ako model CBOW. Pri tomto modeli cieľové slovo slúži ako vstup neurónovej siete pri trénovaní predikcie niekoľkých kontextových slov. V tomto prípade je miesto jedného vektora pravdepodobností vypočítaných C vektorov pre jednotlivé kontextové slová. Pre každý z týchto vektorov sa následne vypočíta chyba oproti cieľu vo forme chybového vektoru. Pri upravovaní váh pomocou spätnej pro-

pagácie je chyba predikcie reprezentovaná súčtom C chybových vektorov, ktoré odpovedajú daným kontextovým slovám [17]. Táto architektúra je uvedená na obrázku 4.4.



Obr. 4.4: Architektúra modelu SG

4.1.3 Aproximácie funkcie softmax

Keďže spomínané modely typicky používajú rozsiahle slovníky, výpočet klasickej funkcie softmax je drahá záležitosť. Toto spôsobuje hlavne jej menovateľ, pri ktorom sa musí vypočítať skalárny súčin medzi vektorom vstupného slova a vektorom každého z V slov daného slovníka. Táto situácia sa typicky rieši použitím rôznych aproximácií tejto funkcie.

4.1.4 Hierarchický softmax

Hierarchický softmax (hierarchical softmax, HS) je výpočtovo efektívna aproximácia funkcie softmax. Táto aproximácia používa binárny strom na reprezentáciu V slov daného slovníka. Jednotlivé slová tvoria listy daného stromu z čoho vyplýva, že existuje $V - 1$ vnútorných vrcholov. Listy reprezentujú pravdepodobnostné rozloženie slov a teda ich súčet sa rovná 1. Pre každý z nich existuje unikátna cesta z koreňa, ktorá je použitá na určenie

pravdepodobnosti reprezentovaného slova. Za týmto účelom má každý vnútorný vrchol priradené pravdepodobnosti prechodu do ľavého a pravého potomka. Tieto pravdepodobnosti sa určujú na základe vektorovej reprezentácie $V - 1$ vnútorných vrcholov, ktoré sú uložené v matici váh W_o výstupnej vrstvy neurónovej siete, namiesto vektorov slov v danom slovníku. Takáto reprezentácia umožňuje získať pravdepodobnostné rozloženie vyhodnotením $\log_2(V)$ vrcholov na rozdiel od klasickej funkcie, pri ktorej je nutné vyhodnotiť V vektorov slov. Word2vec používa Huffmanov binárny strom, pri ktorom je slovám s najvyššou frekvenciou priradený najkratší kód, čo má za následok zrýchlené trénovanie [17].

4.1.5 Negatívne vzorkovanie

Pri použití klasickej funkcie softmax je nutné pri každej iterácií aktualizovať všetky váhy matice W_o výstupnej vrstvy neurónovej siete. Negatívne vzorkovanie (negative sampling, NS) rieši tento problém zmenšením počtu váh, ktoré je nutné aktualizovať v každej iterácií na niekoľko vzoriek. Tieto vzorky zahŕňajú pozitívnu vzorku, ktorá je reprezentovaná slovom odhadovaným neurónovou sieťou pre daný vstup a zvolený počet negatívnych vzoriek. Hlavý cieľ tohto prístupu je zvýšiť podobnosť vstupného slova s pozitívnou vzorkou a naopak minimalizovať podobnosť s negatívnymi vzorkami. Negatívne vzorky sú vybrané zo slovníka na základe zvoleného pravdepodobnostného rozloženia. Word2vec za týmto účelom využíva vážené unigramové rozloženie [17].

4.1.6 Podvzorkovanie frekventovaných slov

Ako už bolo spomínané v kapitole 2, slová s vysokou frekvenciou výskytu často do výsledného modelu zavedú nežiaduci šum. Word2vec implementuje jednoduchý prístup podvzorkovania, ktorý zamedzuje nadmernému vplyvu častých slov. Pre každé slovo w_i zo vstupnej tréningovej sady je vypočítaná pravdepodobnosť jeho vyradenia pomocou nasledujúceho vzorca, kde parameter t označuje zvolený prah a $f(w_i)$ frekvenciu slova w_i:

$$p(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}} \tag{4.2}$$

Bolo potvrdené, že tento prístup urýchľuje proces učenia a výrazne zvyšuje presnosť vektorov menej častých slov [14].

4.2 FastText

Väčšina modelov reprezentuje každé slovo v slovníku ako unikátny vektor. Takáto reprezentácia funguje dobre pre jazyky, ktoré nie sú príliš morfologicky zložité. Pre morfologicky bohaté jazyky, ako je napríklad čeština, je však naopak veľmi limitujúce, že takáto reprezentácia neberie do úvahy vnútornú štruktúru slov. Knižnica FastText rozširuje modely rodiny Word2vec o novú reprezentáciu slov, ktorá je nazývaná ako bag of character n-grams. Jej princíp spočíva v priradení vektorovej reprezentácie k n-gramom znakov, ktoré sa vyskytujú v jednotlivých slovách. Pre každé slovo w teda existuje množina G_w obsahujúca n-gramy, ktoré sa v ňom nachádzajú. Pri zachovávaní n-gramov, ktoré majú minimálne 3 znaky a maximálne 6 znakov by pre slovo „auto“ množina G_w vyzerala nasledovne:

$$G_w = \{aut, uto, auto\} \tag{4.3}$$

Každému n-gramu g z tejto množiny je priradená vektorová reprezentácia z_g. Suma vektorov z_g slúži ako reprezentácia slova v projekčnej vrstve pri počítaní skóre podobnosti vo výstupnej vrstve neurónovej siete medzi vstupným slovom w a odhadovaným slovom c, ktorého vektor označíme ako v_c:

$$s(w, c) = \sum_{g \in G_w} z_g{}^{\intercal} v_c \tag{4.4}$$

Do množiny G_w je vždy pridané samotné slovo w, aby bolo možné taktiež učiť vektorové reprezentácie jednotlivých slov. Toto znamená, že množina všetkých n-gramov je nadmnožina slovníka daného modelu [4].

4.3 GloVe

GloVe [15] je jeden z najnovších modelov založených na počte, ktorý produkuje nízkodimenzionálny lineárny vektorový priestor. Hlavným cieľom návrhu GloVe bolo vytvorenie modelu, ktorý efektívne využíva informácie o spoluvýskyte slov v korpuse a zároveň zachytáva lingvistické vzory v podobe lineárnych vzťahov medzi vektormi slov, podobne ako prediktívne modely Word2vec.

Základná intuícia modelu je odvodená z pozorovania, že niektoré aspekty významu môžu byť získané z pomerov pravdepodobností spoluvýskytu slov, ako si môžeme ukázať na nasledujúcom príklade. Nech X je matica spoluvýskytu typu slovo-slovo s prvkami X_{ij}. Pravdepodobnosť toho, že slovo j sa nachádzajú v kontexte slova i sa dá vyjadriť ako $P(j \mid i) = X_{ij}/X_i$, pričom X_i značí počet výskytov ľubovoľného slova v kontexte slova i. Nech $i =$ ľad a $j =$ para. Vzťah medzi týmito slovami je možné pozorovať na základe pomeru pravdepodobností ich spoluvýskytu s rôznymi slovami k. Nech k je slovo, ktoré súvisí so slovom ľad, ale nie so slovom para, ako napríklad $k =$ tuhý. Pravdepodobnosť $P(\text{tuhý} \mid \text{ľad})$ bude relatívne vysoká a naopak $P(\text{tuhý} \mid \text{para})$ bude relatívne nízka. Pomer $P(\text{tuhý} \mid \text{ľad})/P(\text{tuhý} \mid \text{para})$ teda bude nadobúdať veľké hodnoty. Pre slovo $k = plyn$, ktoré súvisí so slovom para a nesúvisí so slovom ľad, bude pomer $P(\text{plyn} \mid \text{ľad})/P(\text{plyn} \mid \text{para})$ naopak veľmi malý. Pre slová k, ktoré súvisia s oboma alebo žiadnym zo slov i a j by tento pomer mal byť blízky jednotke. Tento príklad naznačuje, že pomer pravdepodobností spoluvýskytu kóduje istú formu významu a teda je vhodnejší na učenie vektorovej reprezentácie slov, ako pravdepodobnosť spoluvýskytu sama osebe.

Cieľ trénovania GloVe spočíva v naučení vektorov slov tak, aby sa skalárny súčin dvoch vektorov rovnal logaritmu pravdepodobnosti spoluvýskytu slov, ktoré reprezentujú. Toto môžeme popísať nasledujúcimi rovnicami, kde w_x značí vektorovú reprezentáciu slova x a b_x jeho skalárny posun (bias):

$$w_i{}^{\intercal} w_j = \log(P(j \mid i)) = \log(X_{ij}) - \log(X_i) \tag{4.5}$$

$$w_i{}^{\intercal} w_j + b_i + b_j = \log(X_{ij}) \tag{4.6}$$

Vzhľadom k tomu, že logaritmus pomeru sa rovná rozdielu logaritmov, pomery pravdepodobností spoluvýskytu slov sú spojené s rozdielmi ich vektorovej reprezentácie. Vďaka tejto vlastnosti, model GloVe zachytáva rôzne lingvistické vzory a dokáže úspešne riešiť úlohy na analógie podobne ako Word2vec. Aby sa zamedzilo nadmernému vplyvu príliš častých a príliš zriedkavých slov, autori použili regresný model, ktorý používa váženú metódu najmenších štvorcov. Navrhnutá váhová funkcia $f(X_{ij})$ má podobu

$$f(X_{ij}) = \begin{cases} (X_{ij}/X_{max})^{\alpha} & \text{pre } X_{ij} < X_{max} \\ 1 & \text{inak} \end{cases} \tag{4.7}$$

kde X_{max} je zvolená hranica počtu spoluvýskytu slov a α je najčastejšie hodnota 3/4. Učenie vektorovej reprezentácie slov modelu GloVe teda prebieha minimalizáciou chybovej funkcie J, ktorá má pri veľkosti slovníka V nasledujúcu formu:

$$J = \sum_{i,j=1}^{V} f(X_{ij})(w_i{}^{\intercal} w_j + b_i + b_j - \log(X_{ij}))^2 \tag{4.8}$$

Kapitola 5

Návrh a implementácia

V tejto kapitole bude popísaný návrh a implementácia vytváraného systému. Sú tu detailne popísané jednotlivé časti systému od predspracovania vstupných textov, cez trénovanie modelov, až po samotné určovanie sémanticky podobných slov. Ďalej sú v tejto kapitole popísané zvolené vstupné korpusy, testovacie sady a spôsoby vyhodnocovania modelov.

5.1 Požiadavky a použité technológie

Cieľom tejto práce je vytvoriť systém na určovanie sémanticky podobných slov. Architektúra vytvoreného systému by mala umožňovať jednoduchú prácu so vstupnými korpusmi rôznych formátov a veľkostí. Systém by mal byť schopný vytvárať nové modely Word2vec, FastText alebo GloVe a taktiež by mal byť schopný pracovať už s existujúcimi modelmi týchto architektúr. Na základe sémantického modelu by malo byť možné k zadaného slovu zobraziť slová sémanticky podobné. Systém by taktiež mal umožniť vyhodnotiť sémantický model na vybranej testovacej sade za účelom určenia jeho úspešnosti.

Ako implementačný jazyk navrhovaného systému bol vybraný jazyk Python. Python je vysokoúrovňový interpretovaný programovací jazyk. Je vyvíjaný ako open source projekt a je kompatibilný s väčšinou bežných platforiem ako Windows, Unix a Mac OS. Python ponúka dynamickú typovú kontrolu, automatickú správu pamäti a podporu viacerých programovacích paradigiem, vrátane objektovo orientovaného a imperatívneho štýlu. Výhodou tohto jazyka je bohatá škála štandardných knižníc, ktoré dokážu významným spôsobom zefektívniť prácu v rôznych oblastiach. Ďalší dôvod výberu jazyka Python sú verejne dostupné knižnice, ktoré sú optimalizované na prácu so sémantickými modelmi.

5.1.1 Použité knižnice

Prvou použitou knižnicou je Gensim[1], ktorá je licencovaná pod GNU LGPLv2.1. Toto znamená, že je zadarmo pre osobné i komerčné využitie. Jedná sa o jednu s najrobustnejších a najefektívnejších knižníc na tvorbu sémantických modelov z textových korpusov. Efektívnosť tejto knižnice spočíva v reimplementácií pôvodne pomalých úsekov kódu v jazyku C pomocou prekladača Cython a v použití paralelizmu. Jej architektúra umožňuje prácu s textami, ktorých veľkosť presahuje množstvo dostupnej operačnej pamäti, čo ju robí vhodnou na prácu s rozsiahlymi korpusmi. Knižnica Gensim obsahuje implementáciu modelov Word2vec, LSA, či LDA a navyše poskytuje nástroj na automatickú identifikáciu fráz vo

[1] https://github.com/RaRe-Technologies/gensim

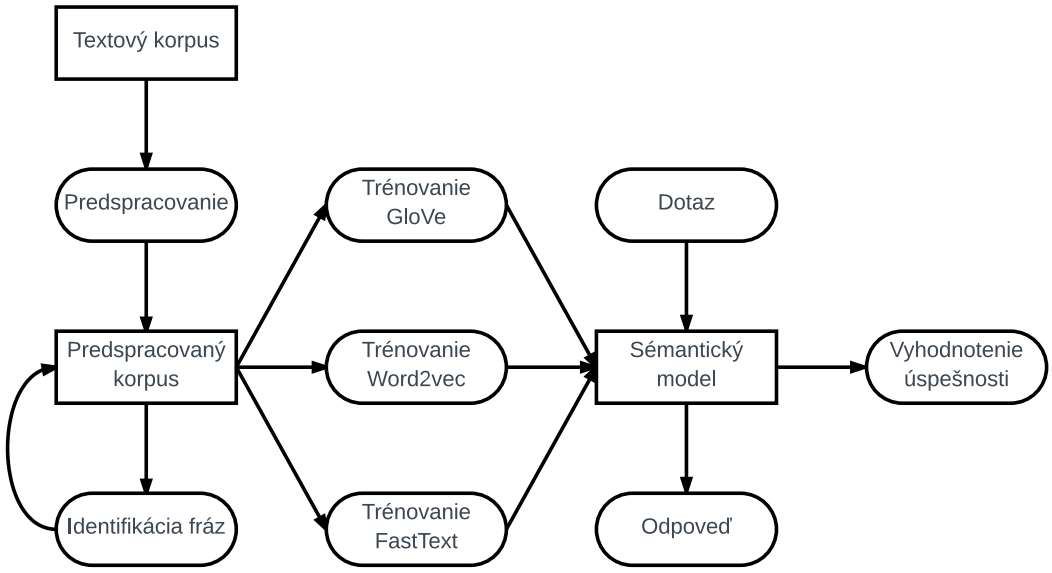
vstupnom korpuse. Implementuje taktiež funkcie na vyhľadanie sémanticky podobných slov k zadanému slovu v danom modeli a určenie sémantickej podobnosti dvoch slov.

Ďalšou z knižníc je FastText[2]. Ide o rozhranie jazyka Python pre oficiálnu knižnicu FastText vytvorenú v jazyku C++ skupinou Facebook Research. Jedná sa o knižnicu pre efektívne učenie slovných reprezentácií a klasifikáciu viet. Podobne ako Gensim, knižnica FastText vyžaduje Cython na preloženie C++ rozšírení. Táto knižnica je licencovaná pod BSD a teda je možné ju v tejto práci využiť.

Glove-python[3] je knižnica, ktorá implementuje model GloVe. Poskytuje prostriedky na tvorbu matice spluvýskytu zo vstupného korpusu aj na samotné učenie vektorovej reprezentácie slov. Podobne ako predchádzajúce knižnice sa pri optimalizáciách spolieha na Cython a paralelizmus. Licencovaná je pod Apache License 2.0.

MorphoDiTa[4] je open-source knižnica na analýzu textov českého jazyka. Táto knižnica poskytuje prostriedky na morfologickú analýzu, morfologickú generáciu, tokenizáciu a značkovanie, pričom vykazuje veľmi vysokú úspešnosť a rýchlosť spracovania až do 200 tisíc slov za sekundu. V navrhovanom systéme je použitá hlavne na morfologickú analýzu, ktorej cieľom je pre každý token v danej vete priradiť odpovedajúcu lemu a POS (part-of-speech) značku. Morfologická analýza je vykonaná v dvoch krokoch: v morfologickom slovníku sa najskôr vyhľadajú všetky možné lemy a POS značky pre dané slovo a následne je na základe algoritmu vybraná optimálna kombinácia výslednej lemy a POS značky. MorphoDiTa k svojmu behu potrebuje jazykové modely, ktoré sú voľne dostupné na stiahnutie z jej oficiálnej internetovej stránky.

5.2 Architektúra systému



Obr. 5.1: Architektúra systému

Práca celého systému sa dá rozdeliť na niekoľko na seba nadväzujúcich krokov. V prvom kroku je vstupný textový korpus predspracovaný a následne uložený vo formáte, ktorý je

[2]https://github.com/salestock/fastText.py
[3]https://github.com/maciejkula/glove-python/
[4]http://ufal.mff.cuni.cz/morphodita

možné jednoducho použiť pri tvorbe modelov. Predspracovaný korpus je v druhom kroku použitý na natrénovanie niektorého z podporovaných sémantických modelov: Word2vec, GloVe alebo FastText. V prípade modelu GloVe je taktiež vytvorená a uložená matica spoluvýskytu, ktorá je neskôr použitá pri analýze úspešnosti modelov. Vytvorený sémanticky model je nakoniec možné vyhodnotiť na vybraných testovacích sadách, alebo použiť na získanie rôznych informácií o sémantickej podobnosti slov. Na obrázku 5.1 je možné vidieť schému celého systému od načítania vstupného textového korpusu až po vyhodnotenie vytvorených modelov.

5.3 Predspracovanie vstupu

Táto práca je zameraná na porovnanie a vyhodnotenie modelov natrénovaných na lematizovaných vstupných korpusoch. Preto vytvorený systém v rámci predspracovania podporuje okrem tokenizácie a filtrovania slov na základe stoplistu aj lematizáciu. Táto funkcionalita je implementovaná v skripte *preprocessor.py*. Spomínaný skript očakáva ako vstup ľubovoľný textový korpus. Jeho práca sa dá popísať postupnosťou niekoľkých krokov, ktoré sú aplikované na jednotlivé riadky vstupného súboru. Pomocou knižnice MorphoDiTa sú v tomto riadku identifikované a tokenizované jednotlivé vety. Nad každou vetou je postupne vykonaná morfologická analýza, ktorá ku každému tokenu priradí lemu a POS značku, na základe ktorej sú z danej vety odfiltrované interpunkčné znamienka. Zo zoznamu zostávajúcich tokenov danej vety sú nakoniec voliteľne odfiltrované slova, ktoré sa nachádzajú v stopliste poskytnutom skriptu.

Výstupný súbor je vo formáte, kde každý riadok obsahuje jednu predspracovanú vetu zo vstupného korpusu, pričom jednotlivé slová sú oddelené medzerou. Tento skript podporuje tri módy výstupu: bez lematizácie, základné lemy, alebo plné lemy obsahujúce dodatočné informácie vo forme komentára, ktoré tvoria s príslušnou lemou jeden token. Predspracovaný korpus je uložený na disk kvôli znovupoužiteľnosti pre viaceré sémantické modely. Pri spustení skriptu je možne zvoliť nasledujúce parametre:

- `-i, --input FILE`
  - Cesta ku vstupnému korpusu.
- `-o, --output_dir DIRECTORY`
  - Cesta k adresáru, kde bude uložený predspracovaný korpus.
- `-sw, --stopwords` FILE
  - Cesta k stoplistu, ktorý má formát jedno slovo na riadok.
- `-t, --tagger` FILE
  - Morfologický tagger, ktorý je potrebný k morfologickej analýze knižnice MorphoDiTa.
- `-l, --lemma {raw, full, none}`
  - Mód práce skriptu na predspracovanie. Pri zvolení možnosti none nebude vykonaná lematizácia vstupu, pri možnosti raw budú výstup tvoriť základné lemy a pri možnosti full budú základné lemy doplnené o komentár.

5.4 Použité korpusy

Na trénovanie sémantických modelov v tejto práci boli vybrané tri rôzne korpusy. Ako prvý korpus bola zvolená česká Wikipédia z dátumu 1.10.2016, ktorej dáta sú voľne prístupné na internete. Ďalšie korpusy, ktoré boli vybrané, sú dostupné na serveroch FIT VUTBR a nazývajú sa CWC-2011[18] a All.vert. Zvolené korpusy sa líšia obsahovo aj rozsahovo, pričom najmenší korpus tvorí Wikipédia a korpus CWC-2011 je najrozsiahlejší.

5.4.1 CWC-2011

Korpus CWC-2011 sa skladá z internetových článkov, blogov a diskusií. Ide o anotovaný korpus, ktorý už obsahuje okrem samotných slov aj ich lemy a POS značky. Keďže táto práca je zameraná na lematizované korpusy, CWC-2011 už nebolo treba predspracovať štandardným spôsobom a stačilo z neho iba extrahovať lemy jednotlivých slov. Toto zabezpečuje skript *cwcParser.py*, ktorý lemy uloží v rovnakom formáte ako skript *preprocessor.py*. Tento korpus má nasledujúci formát:

```
<s>
Slovo1    Lema    POS značka    Dodatočne informácie
Slovo2    Lema    POS značka    Dodatočne informácie
<g/>
!
<s>
 . . .
<s>
```

Nepárová značka <s> označuje hranice vety a značka <g/> označuje, že nasledujúci znak je interpunkčné znamienko.

5.4.2 All.vert

Podobne ako CWC-2011 je aj tento korpus anotovaný. Skladá sa zo starých novinových článkov a kníh. Formát korpusu All.vert sa veľmi nelíši od CWC-2011 a teda podobne ako v jeho prípade stačí extrahovať požadované informácie. Túto funkcionalitu implementuje skript *allvertParser.py*. Formát korpusu All.vert si môžeme ukázať na nasledujúcom príklade:

```
<DOC>
<P>
<S>
Poradie slova    Slovo    Lema    POS značka    Dodatočne informácie
 . . .
</S>
</P>
</DOC>
```

Párová značka <DOC> označuje hranice dokumentu, značka <P> hranice odstavca a značka <S> označuje hranice vety.

5.4.3 Wikipédia

V pôvodnom korpuse Wikipédie je každý dokument reprezentovaný jedným XML elementom. Samotný súbor obsahuje veľké množstvo obrázkov, tabuliek, zdrojov a rôznych XML značiek popisujúcich vnútornú štruktúru Wikipédie. Za účelom extrakcie čistého textu z tohto korpusu bol použitý nástroj Wikipedia Extractor[5]. Extrahovaný text bol použitý ako jeden zo vstupných korpusov vytvoreného systému.

5.5 Automatická identifikácia fráz

Po predspracovaní korpusu nasleduje nepovinný krok, pri ktorom sú v danom korpuse identifikované frázy. Tento krok je zahrnutý v skripte trainPhrases.py. Ako základ identifikácie fráz je použitá trieda Phrases knižnice Gensim. Pomocou nej je možné z textového korpusu vytvoriť model, ktorý obsahuje frázy podľa zadaných kritérií. Vytvorený model môže byť následne použitý na transformovanie vstupného textu do podoby, kde sú jednotlivé identifikované frázy spojené v jeden token, pričom jednotlivé slová sú oddelené zvoleným znakom. Frázy sú identifikované štatistickou analýzou, pri ktorej sú nájdené slová, ktoré sa často vyskytujú vedľa seba a naopak zriedkavo v iných kontextoch [19]. Skóre, ktoré je použité k identifikácií, je vypočítané pomocou nasledujúcej rovnice:

$$\text{skóre} = \frac{(cnt(ab) - min)N}{cnt(a)cnt(b)} \tag{5.1}$$

V tejto rovnici $cnt(a)$ označuje počet výskytov slova a, $cnt(b)$ počet výskytov slova b, $cnt(ab)$ počet výskytov slov a a b vedľa seba, N počet slov v slovníku a min hraničný počet výskytu slov a fráz. Ak je počet výskytov menší ako min, sú na vstupe ignorované. Ak je vypočítané skóre väčšie ako zvolená prahová hodnota, spojenie slov a a b je akceptované ako fráza. Skript *trainPhrases.py* je možné spustiť s nasledujúcimi parametrami:

- `-i, --input FILE`
  - Cesta k predspracovanému korpusu.
- `-o, --output_dir DIRECTORY`
  - Cesta k adresáru, kde bude uložený korpus s frázami.
- `-n, --ngram {bigram, trigram}`
  - Mód práce skriptu. Pri zvolení možnosti bigram budú identifikované dvojslovné frázy a pri možnosti trigram aj trojslovné frázy.
- `-c, --count NUMBER`
  - Minimálny počet výskytov slov alebo fráz, pri ktorom sú zaradené do tvoreného modelu.
- `-t, --threshold NUMBER`
  - Zvolená prahová hodnota použitá pri rozhodovaní o akceptovaní fráz.

[5] https://github.com/bwbaugh/wikipedia-extractor

5.6 Trénovanie modelov

Po predspracovaní a voliteľnej identifikácií fráz nasleduje samotné trénovanie sémantického modelu. Vytvorený systém podporuje niekoľko typov modelov: rodinu modelov Word2vec, jej rozšírenú verziu v podobe knižnice FastText a model GloVe.

5.6.1 Word2vec a FastText

K trénovaniu modelov rodiny Word2vec je použitá trieda `Word2vec` z knižnice Gensim. Ako vstup očakáva korpus vo forme objektu, ktorý má neimplementovanú metódu `__iter__`, pričom v každej iterácií vracia jednu vetu z korpusu vo forme zoznamu tokenov. Tento objekt je implementovaný v module *corpusIterable.py*. Trénovanie modelov Word2vec je vykonané skriptom *trainWord2vec.py*, ktorý má nasledujúce parametre:

- `-i, --input FILE`
  - Cesta k predspracovanému korpusu.
- `-o, --output_dir DIRECTORY`
  - Cesta k adresáru, kde bude uložený model Word2vec.
- `-e, --epochs NUMBER`
  - Počet iterácií trénovania.
- `-d, --dimensions NUMBER`
  - Počet dimenzií výsledných vektorov, ktoré reprezentujú slová v slovníku.
- `-c, --count NUMBER`
  - Minimálny počet výskytov slova, pri ktorom je zaradené do modelu.
- `-m, --model {cbow, sg}`
  - Výber modelu Continuous Bag of Words alebo modelu Skip-Gram.
- `-a, --algorithm {hs, ns}`
  - Výber aproximácie funkcie softmax. Možnosť hs označuje hierarchical softmax a možnosť ns zase negative sampling.
- `-s, --samples NUMBER`
  - Počet negatívnych vzoriek pri aproximácií negative sampling.

Na trénovanie modelov pomocou knižnice FastText je určený skript *trainFasttext.py*, ktorý má zhodné parametre so skriptom *trainWord2vec.py*. Formát uloženého modelu tejto knižnice je taktiež zhodný s knižnicou Gensim.

5.6.2 GloVe

Tvorba modelu GloVe pozostáva z dvoch krokov. Najskôr je vytvorená matica spoluvýskytu analýzou vstupného korpusu a následne sú na jej základe naučené vektorové reprezentácie slov.

Pri tvorbe matice spoluvýskytu je použitá trieda `Corpus` knižnice glove-python. V tejto implementácií je použitá matica, kde sú spoluvýskyty slov vážené ich vzdialenosťou v rámci kontextového okna:

$$\text{spoluvýskyt} = \frac{1}{\text{vzdialenosť}} \tag{5.2}$$

Keďže táto trieda pri tvorení slovníka nepodporuje filtráciu slov pomocou minimálneho počtu výskytu, bola pri tvorbe slovníka použitá trieda `Dictionary` knižnice Gensim, ktorá túto funkcionalitu podporuje. Vytvorený slovník je odovzdaný triede `Corpus` cez jej konštruktor. Samotná matica je implementovaná pomocou triedy `coo_matrix` knižnice SciPy. Práca s touto maticou je pokrytá v module *cooccurrence.py*.

Samotné trénovanie modelu GloVe je vykonané skriptom *trainGlove.py*. Tento skript podporuje ako vstup textový korpus alebo rovno maticu spoluvýskytu vytvorenú pomocou knižnice glove-python. Ak je na vstupe korpus, matica je dodatočne vytvorená a uložená. Vytvorený skript taktiež dodatočne implementuje funkciu na uloženie modelu vo formáte, ktorý je podporovaný knižnicou Gensim. Táto knižnica je totiž použitá pri všetkých vyhodnocovacích skriptoch a na rozdiel od knižnice FastText, knižnica glove-python tento formát nepodporuje. Pri spustení tohto skriptu je možné zadať nasledujúce parametre:

- `-i, --input FILE`
  - Cesta k predspracovanému korpusu alebo matici spoluvýskytu.
- `-o, --output_dir DIRECTORY`
  - Cesta k adresáru, kde bude uložený model GloVe.
- `-om, --output_mdir DIRECTORY`
  - Cesta k adresáru, kde bude uložená matica spoluvýskytu.
- `-e, --epochs NUMBER`
  - Počet iterácií trénovania.
- `-d, --dimensions NUMBER`
  - Počet dimenzií výsledných vektorov, ktoré reprezentujú slová v slovníku.
- `-c, --count NUMBER`
  - Minimálny počet výskytov slova, pri ktorom je zaradené do modelu.
- `-m, --mode {matrix, corpus}`
  - Tento parameter určuje, či sa na vstupe nachádza korpus alebo matica spoluvýskytu.

5.7 Vyhodnotenie sémantických modelov

Po natrénovaní modelov je možné vyhodnotiť ich úspešnosť na vybraných testovacích sadách. Metrika, ktorá sa najčastejšie používa na určenie sémantickej podobnosti vektorov slov je kosínusová vzdialenosť. Táto vzdialenosť nadobúda hodnoty od -1 do 1, pričom čím bližšie sa táto hodnota blíži k 1, tým silnejšia je sémantická podobnosť medzi slovami. Pre vektory a a b, ktoré reprezentujú dve slová zo slovníka by sa kosínusová vzdialenosť medzi nimi vypočítala nasledovne:

$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\|\|\vec{b}\|} \tag{5.3}$$

Knižnica Gensim bola využitá ako základ všetkých vytvorených vyhodnocovacích skriptov, pretože priamo poskytuje niekoľko metód na prácu so sémantickou podobnosťou, ktoré túto metriku využívajú. Prvá metóda knižnice Gensim, ktorá bola v tejto práci použitá je `similarity`. Ide o priamočiare vypočítanie kosínusovej vzdialenosti medzi vektormi dvoch slov, ktoré sa nachádzajú v slovníku daného modelu. Druhá použitá metóda je nazvaná `most_similar`. Táto metóda vyhľadá a vráti slová, ktoré sú sémanticky najviac podobné vstupnému vektoru, ktorý je odvodený od dvoch vstupných zoznamov slov. V zozname `positive` sa nachádzajú slová, ktoré pozitívne prispievajú k významu a ich vektory sú vážené váhou 1. Zoznam `negative` zas tvoria slová, ktoré prispievajú k významu negatívne a sú teda vážené váhou -1. Vstupný vektor je vypočítaný ako vážený priemer vektorov slov v týchto zoznamoch.

Za účelom vyhodnotenia boli vybrané tri rôzne testovacie sady: slovné analógie, slovník synoným a krycie mená.

5.7.1 Slovné analógie

Na vyhodnotenie úspešnosti sémantických modelov v tejto oblasti boli využité slovné analógie navrhnuté pre česky jazyk v odbornom článku [16]. Táto dátová sada obsahuje 8705 sémantických a 13552 syntaktických otázok. Otázky sú vyhodnocované princípom ukázanom na príklade 3.2. Dokopy tvoria 4 sémantické a 6 syntaktických kategórií, pričom každá z nich obsahuje okolo 40 dvojíc slov, medzi ktorými je rovnaký vzťah. Navrhnuté dvojice väčšinou tvoria dve slová, no v niektorých kategóriách sa môžu vyskytnúť aj dvojslovné frázy. Sémantické otázky pokrývajú kategórie antonymá, rodinné vzťahy, mesto-štát a prezident-štát. Kategórie prídavné mená-gradácia, národnosť, podstatné mená-množné číslo, zamestnania, slovesá-minulý čas a zámená pokrývajú zas syntaktické otázky.

Autori odborného článku verejne sprístupnili túto dátovú sadu[6] spolu s vyhodnocovacím skriptom v jazyku Python pod licenciou Apache License 2.0. Tento vyhodnocovací skript bol v upravenej verzii použitý pri tejto práci na vyhodnotenie vytvorených modelov. Prvá modifikácia spočíva v rozšírení skriptu o lematizáciu slovných párov, pretože vyhodnocované modely boli trénované na lematizovaných korpusoch. Druhá modifikácia spočíva v pridaní módu, pri ktorom sú ignorované analógie s viacslovnými frázami za účelom objektívneho vyhodnotenia modelov, ktoré neboli na viacslovných frázach trénované. Skript taktiež používa knižnicu Gensim, a preto vstupný model musí byť vo formáte, ktorý je s ňou kompatibilný.

[6] https://github.com/Svobikl/cz_corpus

5.7.2 Slovník synoným

Ako druhá testovacia sada bol vybraný slovník synoným, ktorý je dostupný na školských serveroch FIT VUTBR. Každý riadok tohto súboru obsahuje jeden záznam v nasledujúcom formáte:

```
<ENTRY>
<HEAD>Slovo</HEAD>
<LINE>Synonymá k prvému významu daného slova</LINE>
<LINE>Synonymá k druhému významu daného slova</LINE>
</ENTRY>
```

Vyhodnotenie každého záznamu prebieha pomocou niekoľkých krokov. Najskôr je vo zvolenom modeli nájdených pomocou metódy `most_similar` n najviac sémanticky podobných slov ku slovu, ktoré sa nachádza v párovej značke <HEAD>. Následne je zistený počet p synoným z množiny všetkých synoným v párových značkách <LINE>, ktorý sa vyskytuje v nájdených najviac podobných slovách. Pre počet p sa výsledné skóre záznamu sa vypočíta ako:

$$\text{Skóre} = \frac{\text{p}}{\text{n}} \tag{5.4}$$

Úspešnosť modelu je reprezentovaná aritmetickým priemerom hodnôt tohto skóre jednotlivých záznamov.

Vyhodnotenie pomocou slovníka synoným implementuje skript *evalSynonyms.py*. Skript podporuje ako lematizáciu, tak mód na ignorovanie záznamov, ktoré obsahujú viacslovné frázy. Výstup tvoria dva súbory, ktoré sa líšia sufixom. V prvom súbore sa nachádzajú hodnoty skóre pre jednotlivé záznamy. V druhom súbore sú v rámci analýzy výsledkov uložené nájdené najviac podobné slová ku slovu v párovej značke <HEAD>. Skript podporuje nasledujúce parametre:

- `-i, --input FILE`
  - Cesta k testovacej sade.
- `-o, --output FILE`
  - Cesta k textovému súboru, kde bude uložený výstup.
- `-m, --model FILE`
  - Cesta k sémantickému modelu, ktorý je uložený vo formáte kompatibilnom s knižnicou Gensim.
- `-d, --dictionary FILE`
  - Cesta k morfologickému slovníku knižnice MorphoDiTa.
- `-t, --topn NUMBER`
  - Počet vyhľadaných najviac podobných slov.
- `-s, --sort`
  - Prepínač na zoradenie jednotlivých záznamov podľa výsledného skóre.

- `-u, --unigram`
  - Prepínač na zapnutie módu, pri ktorom sú ignorované viacslovné frázy.

5.7.3 Krycie mená

Posledná testovacia sada je inšpirovaná spoločenskou hrou Krycie mená. Jej cieľom je určiť, či vytvorené sémantické modely môžu byť použité ako pomôcka pri hraní tejto hry, alebo využité k implementácií automatického spoluhráča alebo oponenta.

V tejto hre figuruje modrý a červený tím špiónov, pričom jeden z hráčov oboch tímov je nominovaný za hlavného špióna. V každej hre je náhodne vybraných 25 krycích mien, ktoré označujú 8 modrých agentov, 8 červených agentov, 7 náhodných okoloidúcich, 1 nájomného vraha a 1 dvojitého agenta. Hlavní špióni poznajú identitu všetkých agentov a striedajú sa v poskytovaní jednoslovnej nápovedy, na základe ktorej sa ich tím snaží uhádnuť krycie mená agentov, ktorí majú farbu danému tímu. Tím, ktorý uhádne všetkých agentov svojej farby vyhráva.

V použitej testovacej sade jednotlivé riadky reprezentujú určitý stav hry Krycie mená. Každý riadok sa skladá z troch stĺpcov, ktoré sú oddelené tabulátorom. Prvý stĺpec obsahuje nápovedu hlavného špióna. Druhý stĺpec predstavuje krycie mená, ktoré by správne mali byť určené na základe danej nápovedy. V tretom stĺpci sa nachádzajú krycie mená, ktoré sú z počiatočných 25 krycích mien aktuálne v hre.

Vyhodnotenie prebieha na základe vypočítania priemernej presnosti (average precision) riadku. Za týmto účelom je najskôr určená presnosť každého zo správnych krycích mien v druhom stĺpci. Pre vypočítanie presnosti je pomocou metódy similarity vypočítaná podobnosť medzi nápovedou v prvom stĺpci a krycími menami v treťom stĺpci a tento stĺpec následne podľa zistenej podobnosti zoradiť. Presnosť každého správneho krycieho mena sa totiž vypočíta ako

$$\text{presnosť} = \frac{\text{očakávaná pozícia v treťom stĺpci}}{\text{reálna pozícia v treťom stĺpci}} \tag{5.5}$$

pričom sa očakáva, že N krycích mien v druhom stĺpci obsadí N prvých pozícií v zoradenom treťom stĺpci, v ktorom tieto pozície reprezentujú najviac sémanticky podobné krycie mená k zadanej nápovede z tejto množiny. Priemerná presnosť riadku je vypočítané pomocou nasledujúcej rovnice:

$$\text{AP} = \frac{1}{N}\sum_{k=1}^{N}\text{presnosť}_k \tag{5.6}$$

Vyhodnotenie na tejto testovacej sade je implementovaná v skripte *evalCodeNames.py*. Tento skript taktiež podporuje rozšírený mód výstupu, ktorý poskytuje dodatočné informácie na analýzu výsledkov. Tieto informácie pre každú dvojicu nápoveda a krycie meno zahŕňajú vážené počty ich spoluvýskytov, ktoré sú získané z matice modelu GloVe a vybraný počet najviac podobných slov k priemeru vektorov tejto dvojice, ktorý určuje ich sémantický smer. Pri spustení skriptu môžu byť zadané nasledujúce parametre:

- `-i, --input FILE`
  - Cesta k testovacej sade.
- `-o, --output FILE`

  - Cesta k textovému súboru, kde bude uložený výstup.

- `-m, --model FILE`
  - Cesta k sémantickému modelu, ktorý je uložený vo formáte kompatibilnom s knižnicou Gensim.
- `-d, --dictionary FILE`
  - Cesta k morfologickému slovníku knižnice MorphoDiTa.
- `-t, --topn NUMBER`
  - Počet vyhľadaných najviac podobných slov.
- `-s, --sort`
  - Prepínač na zoradenie jednotlivých záznamov podľa výslednej priemernej presnosti.
- `-v, --verbose`
  - Prepínač na zapnutie rozšíreného výstupu.
- `-c, --co_matrix`
  - Cesta k matici spoluvýskytu.

5.8 Sémantická podobnosť slov

Vytvorený systém ponúka dva rôzne skripty na určovanie sémanticky podobných slov. Skript *mostSimilar.py* využíva metódu `most_similar` na vypísanie zadaného počtu sémanticky najviac podobných slov k zadaným vstupným slovám. Keďže táto práca je zameraná na lematizované vstupné korpusy, tento skript rozširuje funkcionalitu metódy `most_similar` o lematizáciu vstupných slov. Vstup je zadaný prostredníctvom štandardného vstupu a najviac podobné slová sú vypísané na štandardný výstup. Skript podporuje nasledujúce parametre:

- `-m, --model FILE`
  - Cesta k sémantickému modelu, ktorý je uložený vo formáte kompatibilnom s knižnicou Gensim.
- `-d, --dictionary FILE`
  - Cesta k morfologickému slovníku knižnice MorphoDiTa.
- `-t, --topn NUMBER`
  - Počet vyhľadaných najviac podobných slov.

Druhý vytvorený skript je *hint.py*. Tento skript umožňuje zoradiť zadanú množinu slov podľa ich podobnosti k vybranému slovu. Jeho pôvodný účel je ako pomôcka pri hre Krycie mená popísanej v predchádzajúcej podkapitole. Na štandardom vstupe očakáva jedno slovo, ktoré reprezentuje nápovedu a ľubovoľný počet slov oddelených čiarkou, ktoré reprezentujú krycie mená agentov. Parametre tohto skriptu sa až na parameter `-t`, ktorý nie je využitý, zhodujú so skriptom *mostSimilar.py*.

Kapitola 6

Vyhodnotenie

V tejto kapitole sú popísané výsledky všetkých sémantických modelov vytvorených v tejto práci. V rámci vyhodnotenia bol skúmaný vplyv vybraných parametrov, lematizácie a rôznych architektúr na výslednú úspešnosť modelov. Úspešnosť bola porovnaná a analyzovaná na testovacích sadách popísaných v predchádzajúcej kapitole. Ďalej sú tu porovnané časové požiadavky na trénovanie jednotlivých modelov.

6.1 Lematizácia

Prvá časť vyhodnotenia sa zaoberá vplyvom lematizácie na vytvárané sémantické modely. Toto vyhodnotenie bolo vykonané na slovných analógiách, pretože autori tejto testovacej sady poskytli výsledky pre modely rodiny Word2vec, ktoré boli trénované na nelematizovanom korpuse tvorenom dátami českej Wikipédie, a v ktorom boli identifikované dvojslovné frázy rovnakým algoritmom ako v tejto práci. Pri trénovaní modelov CBOW a SG autori použili algoritmus negative sampling a kontextové okno o veľkosti 10 slov. Výsledky sú dostupné pre modely s rôznymi počtami dimenzií vektorov a trénovacích epoch [19].

Word2vec				
Kategórie	Bez lematizácie		S lematizáciou	
	CBOW	SG	CBOW	SG
Antonymá (pods. mená)	8,53%	6,19%	15,86%	14,15%
Antonymá (príd. mená)	15,45%	5,69%	36,53%	40,30%
Antonymá (slovesá)	2,86%	0,18%	5,98%	5,09%
Štát-prezident	0,09%	0,27%	1,78%	4,19%
Štát-mesto	25,94%	9,98%	66,58%	74,87%
Rodinné vzťahy	15,68%	6,67%	30,62%	30,27%
Podstatné mená (množ. č.)	60,56%	23,95%	-	-
Zamestnania	6,82%	2,53%	54,21%	49,67%
Slovesá (minulý čas)	48,53%	8,77%	-	-
Zámená (množ. č.)	7,80%	0,79%	-	-
Prídavné mená (gradácia)	20,00%	7,50%	-	-
Národnosť	0,34%	0,08%	49,24%	47,31%

Tabuľka 6.1: Vyhodnotenie lematizácie pre modely Word2vec

Za účelom porovnania boli v tejto práci vytvorené obdobné modely na korpuse tvorenom lematizovanou českou Wikipédiou. Pri ich tvorbe bol zvolený variant s desiatimi trénovacími epochami a počtom dimenzií 300. Porovnanie modelov v závislosti na lematizácií je zobrazené v tabuľke 6.1. Ako je možné vidieť na jednotlivých kategóriách, lematizácia výrazne zvyšuje presnosť sémantických modelov pre český jazyk. Kategórie podstatné mená, slovesá a zámená, ktoré sú zamerané na syntaktické analógie medzi jednotnými a množnými číslami slov, sú z dôvodu použitia lematizácie ignorované, keďže táto odstráni syntaktické rozdiely, ktoré sa tieto kategórie snažia sledovať. Rovnaká situácia nastáva aj v prípade syntaktickej kategórie prídavné mená.

6.2 Charakteristika vytvorených modelov

Okrem modelov spomenutých v predchádzajúcej podkapitole bolo vytvorených ďalších 34 rôznych modelov. Na základe výsledkov v tabuľke 6.1 boli natrénované na čisto lematizovaných korpusoch. V týchto korpusoch však v rámci predspracovania neprebehla identifikácia viacslovných fráz na rozdiel od predchádzajúceho experimentu. Ako už bolo spomínané, v tejto práci boli využité 3 rôzne korpusy, ktorých veľkosti v jednotke počet slov je možné vidieť v tabuľke 6.2.

Korpus	Počet slov
Wikipédia	90 212 850
All.vert	539 740 976
CWC-2011	2 650 854 889

Tabuľka 6.2: Porovnanie veľkosti korpusov

Všetky vytvorené modely majú niekoľko spoločných charakteristík. Veľkosť kontextového okna bola pre všetky modely nastavená na hodnotu 10, ako počet dimenzií vektorov slov bola zvolená hodnota 300 a minimálny počet výskytov slov bol pre Wikipédiu nastavený na 5, pre ostatné korpusy kvôli ich veľkosti bol nastavený na 10. Trénovanie bolo vykonané na školských serveroch athena 7 až 11.

Pri modeli GloVe boli analyzované dve varianty, ktoré sa líšia počtom trénovacích epoch. Podľa [17] je totiž počet trénovacích iterácií parameter, ktorý výrazne ovplyvňuje presnosť modelov. V tejto práci boli skúmane modely s 15 a 30 epochami. Čas potrebný pre vytvorenie matíc spoluvýskytu a pre samotné trénovanie modelov je zobrazený v tabuľke 6.3.

GloVe			
Korpus	Matica spoluvýskytu	15 iterácií	30 iterácií
Wikipédia	7m	21m	40m
All.vert	38m	59m	1h 56m
CWC-2011	5h 1m	2h 46m	5h 40m

Tabuľka 6.3: Doby tvorby matíc spoluvýskytu a trénovania modelov GloVe

V prípade Word2vec bol skúmaný model CBOW aj SG, pričom pri každom boli testované ich kombinácie s oboma aproximáciami funkcie softmax (HS a NS). Pri týchto modeloch bol nastavený počet iterácií na 10, pri použití NS bol nastavený počet negatívnych

vzoriek na 5 a pre podvzorkovanie slov bol zvolený prah 1×10^{-4}. Doby trénovania týchto modelov sú uvedené v tabuľke 6.4.

Word2vec				
Korpus	CBOW		SG	
	HS	NS	HS	NS
Wikipédia	29m	26m	44m	34m
All.vert	2h 57m	2h 33m	4h 18m	3h 24m
CWC-2011	15h 33m	14h 34m	18h 21m	17h 3m

Tabuľka 6.4: Doby trénovania modelov Word2vec

Pri trénovaní modelov FastText boli použité rovnaké parametre ako v prípade klasického Word2vec. Jediný rozdiel je v pridaní parametrov minimálneho a maximálneho počtu znakov n-gramov, ktoré sú špecifické pre modely FastText. Minimálny počet bol nastavený na 3 a maximálny počet na 6. Vďaka charakteru týchto modelov je ich trénovacia doba omnoho vyššia ako v prípade Word2vec, čo je možne vidieť v tabuľke 6.5.

FastText				
Korpus	CBOW		SG	
	HS	NS	HS	NS
Wikipédia	58m	52m	1h 34m	1h 8m
All.vert	5h 40m	5h 13m	8h 55m	6h 57m
CWC-2011	21h 6m	18h 48m	33h 27m	25h 40m

Tabuľka 6.5: Doby trénovania modelov FastText

6.3 Vyhodnotenie Wikipédie

V tejto podkapitole sú zhrnuté a analyzované výsledky modelov, ktoré boli trénované na lematizovaných dátach českej Wikipédie. Rovnako ako pri modeloch z 6.1 sú v rámci slovných analógií ignorované syntaktické kategórie podstatné mená, slovesá, zámená a prídavné mená. Keďže ide o variantu korpusu, v ktorej neboli identifikované viacslovné frázy, je ignorovaná aj kategória prezident-štát, ktorej každá otázka obsahuje token tvorený menom a priezviskom prezidenta. Výsledky tejto sady sú typu top 1, čo znamená, že v modeli bolo na základe otázky na analógiu vyhľadané práve jedno slovo. Pri sade krycie mená bola vypočítaná priemerná presnosť prevedená na percentá a v prípade slovníka synoným boli testované varianty top 10 a top 20.

V tabuľke 6.6 sú zobrazené výsledky pre modely GloVe. Až na kategóriu antonymá (podstatné mená) GloVe dosahuje oveľa nižšiu úspešnosť oproti modelom Word2vec a FastText. Najvýraznejší rozdiel je v syntaktických kategóriách, v ktorých dosahuje niekoľkonásobne horšie výsledky. Zvýšenie počtu iterácií v niektorých kategóriách viedlo k marginálne lepším a v iných, naopak, k horším výsledkom.

Pre rodinu modelov Word2vec sa výsledky nachádzajú v tabuľke 6.7. Na testovacích sadách slovné analógie a slovník synoným mal najlepšiu úspešnosť model CBOW NS, ktorý v niektorých kategóriách dosahuje viac ako dvojnásobnú úspešnosť oproti modelu GloVe.

GloVe		
Kategórie	15 iterácií	30 iterácií
Antonymá (pods. mená)	16,64%	18,99%
Antonymá (príd. mená)	13,47%	10,05%
Antonymá (slovesá)	3,93%	3,75%
Štát-mesto	48,40%	50,28%
Rodinné vzťahy	31,64%	29,78%
Zamestnania	8,59%	7,74%
Národnosť	8,81%	7,48%
Krycie mená	48,67%	50,98%
Slovník synoným (Top 10)	7,25%	8,10%
Slovník synoným (Top 20)	9,55%	10,62%

Tabuľka 6.6: Vyhodnotenie modelov GloVe pre Wikipédiu

Pre testovaciu sadu krycie mená mal najlepšie výsledky model SG HS, ktorý zvíťazil nad CBOW NS o niečo menej ako 3%.

Word2vec				
Kategórie	CBOW		SG	
	HS	NS	HS	NS
Antonymá (pods. mená)	10,03%	19,06%	15,50%	12,59%
Antonymá (príd. mená)	38,91%	36,93%	36,41%	37,22%
Antonymá (slovesá)	4,82%	5,98%	5,54%	4,73%
Štát-mesto	46,18%	68,81%	68,72%	75,58%
Rodinné vzťahy	21,76%	37,96%	35,80%	38,43%
Zamestnania	32,58%	59,34%	35,86%	50,67%
Národnosť	28,79%	47,54%	37,22%	46,59%
Krycie mená	61,57%	63,71%	66,23%	65,04%
Slovník synoným (Top 10)	15,16%	15,69%	12,69%	10,02%
Slovník synoným (Top 20)	18,83%	19,78%	16,10%	12,42%

Tabuľka 6.7: Vyhodnotenie modelov Word2vec pre Wikipédiu

Modely FastText vďaka svojej architektúre očakávane dosahujú najlepšie výsledky v syntaktických kategóriách zamestnania a národnosť. Na testovacích sadách slovné analógie a krycie mená dopadol celkovo najlepšie model SG NS. Pre slovník synoným vyhráva CBOW HS, ktorý zvíťazil nad SG NS približne o 3%. Modely FastText dosiahli podobné výsledky ako klasický Word2vec, pričom FastText konzistentne vyhráva o niekoľko percent na všetkých testovacích sadách, okrem niektorých sémantických kategórií slovných analógií. Cena za toto zlepšenie je však oveľa vyššia doba trénovania. Výsledky modelov FastText sú zhrnuté v tabuľke 6.8.

FastText				
Kategórie	CBOW		SG	
	HS	NS	HS	NS
Antonymá (pods. mená)	11,74%	18,35%	17,35%	19,63%
Antonymá (príd. mená)	34,26%	17,19%	37,10%	29,91%
Antonymá (slovesá)	7,59%	10,90%	5,90%	6,07%
Štát-mesto	38,77%	50,71%	64,98%	65,33%
Rodinné vzťahy	28,40%	33,64%	39,97%	40,59%
Zamestnania	79,21%	86,70%	72,39%	83,67%
Národnosť	47,25%	67,14%	52,94%	71,02%
Krycie mená	62,43%	59,18%	66,59%	67,58%
Slovník synoným (Top 10)	15,52%	11,24%	13,53%	12,86%
Slovník synoným (Top 20)	20,70%	15,47%	17,54%	17,83%

Tabuľka 6.8: Vyhodnotenie modelov FastText pre Wikipédiu

6.4 Vyhodnotenie All.vert

Keďže česká Wikipédia je pomerne malý korpus, ďalší experiment bol vykonaný na korpuse All.vert, ktorý je približne šesťnásobne rozsiahlejší. Ďalší ich dôležitý rozdiel sa týka ich obsahu. Wikipédia je internetová encyklopédia, ktorá obsahuje články napísané výlučne v odbornom štýle, a teda nemusí zachytávať niektoré významy slov a pravdepodobne obsahuje užšiu slovnú zásobu. All.vert sa skladá zo starých novinových článkov a kníh a teda je pravdepodobné, že zachytáva väčšiu rovinu významu slov ako Wikipédia. Samotné vyhodnotenie modelov prebiehalo rovnakým spôsobom ako pre českú Wikipédiu.

Podobne ako v predchádzajúcom experimente, modely GloVe dosiahli najnižšiu úspešnosť. Použitie korpusu All.vert však spôsobilo nemalý nárast úspešnosti vo všetkých kategóriách okrem kategórie rodinné vzťahy. Najvýraznejšie zlepšenie bolo na testovacej sade slovník synoným a syntaktických kategóriách slovných analógií, kde bol zaznamenaný nárast úspešnosti až na dvojnásobok. Výsledky pre tieto modely sú zhrnuté v tabuľke 6.9.

GloVe		
Kategórie	15 iterácií	30 iterácií
Antonymá (pods. mená)	18,35%	16,86%
Antonymá (príd. mená)	20,85%	22,71%
Antonymá (slovesá)	6,79%	5,63%
Štát-mesto	59,80%	63,10%
Rodinné vzťahy	25,77%	25,31%
Zamestnania	16,67%	15,32%
Národnosť	24,05%	25,66%
Krycie mená	61,16%	62,29%
Slovník synoným (Top 10)	15,33%	16,30%
Slovník synoným (Top 20)	19,79%	21,10%

Tabuľka 6.9: Vyhodnotenie modelov GloVe pre All.vert

Rodina modelov Word2vec, podobne ako v prípade GloVe, zaznamenala pokles úspešnosti iba v kategórií rodinné vzťahy. Ostatné kategórie vyšli konzistentne lepšie, pričom bol zaznamenaný takmer dvojnásobný nárast úspešnosti na testovacej sade slovník synoným. Pre sady slovné analógie a slovník synoným znova zvíťazil model CBOW NS, pričom pre sadu krycie mená mal o 4% horšiu úspešnosť ako najlepší model SG NS. V tabuľke 6.10 sú zobrazené úspešnosti týchto modelov.

Word2vec				
Kategórie	CBOW		SG	
	HS	NS	HS	NS
Antonymá (pods. mená)	13,30%	23,04%	19,63%	21,27%
Antonymá (príd. mená)	32,12%	32,75%	36,35%	35,54%
Antonymá (slovesá)	8,04%	9,11%	9,82%	8,21%
Štát-mesto	28,79%	78,25%	71,57%	78,79%
Rodinné vzťahy	14,04%	31,17%	21,91%	35,49%
Zamestnania	46,89%	68,27%	53,96%	61,11%
Národnosť	40,81%	66,00%	53,50%	60,89%
Krycie mená	65,25%	66,75%	69,33%	70,55%
Slovník synoným (Top 10)	26,06%	28,43%	22,50%	22,84%
Slovník synoným (Top 20)	32,01%	34,58%	27,98%	29,00%

Tabuľka 6.10: Vyhodnotenie modelov Word2vec pre All.vert

Ako vidieť v tabuľke 6.11, modely FastText v rámci syntaktických kategórii dosiahli nárast úspešnosti v kategórií národnosť a naopak pokles úspešnosti v kategórií zamestnania oproti Wikipédií. V ostatných kategóriách dosiahli veľmi podobné výsledky ako modely Word2vec, pričom v tomto experimente Word2vec prekonal FastText takmer vo všetkých z nich.

FastText				
Kategórie	CBOW		SG	
	HS	NS	HS	NS
Antonymá (pods. mená)	12,95%	19,84%	18,71%	17,99%
Antonymá (príd. mená)	29,62%	13,70%	34,76%	31,30%
Antonymá (slovesá)	8,30%	6,52%	7,23%	4,55%
Štát-mesto	23,71%	58,11%	70,56%	75,58%
Rodinné vzťahy	14,20%	23,46%	22,38%	31,02%
Zamestnania	76,85%	83,84%	68,01%	73,06%
Národnosť	56,53%	76,33%	60,89%	72,73%
Krycie mená	64,85%	63,95%	69,48%	70,34%
Slovník synoným (Top 10)	24,60%	19,84%	22,33%	23,06%
Slovník synoným (Top 20)	31,57%	26,13%	28,22%	30,86%

Tabuľka 6.11: Vyhodnotenie modelov FastText pre All.vert

6.5 Vyhodnotenie CWC-2011

Tretí experiment bol vykonaný na korpuse CWC-2011, ktorý je najväčší zo skúmaných korpusov s 2,6 biliónmi slov. Keďže sa jedná o internetové blogy, diskusie a články, ktoré sú napísane širšou verejnosťou, predpokladá sa menšia celková kvalita korpusu. Vďaka jeho charakteru sa taktiež očakáva najširšia slovná zásoba v rámci skúmaných korpusov, ktorá je rozšírená o rôzne nespisovné a gramaticky nesprávne slová.

Pri tomto korpuse modely GloVe prvýkrát dosiahli výsledky porovnateľné s modelmi Word2vec a FastText. Na testovacej sade slovné analógie vo väčšine sémantických kategórií dokonca skončili s lepšou úspešnosťou ako ostatné modely trénované na CWC-2011. V syntaktických kategóriách bol zaznamenaný prudký nárast úspešnosti oproti predchádzajúcim korpusom, no stále to nestačilo na ostatné modely. Pre testovaciu sadu krycie mená prehral GloVe o približne 3% oproti najlepšiemu z modelov Word2vec a o 4% oproti najlepšiemu z modelov FastText z tohto experimentu. Pre slovník synoným je úspešnosť veľmi podobná modelom FastText a približne o 5% horšia ako pre modely Word2vec. Výsledky sú zhrnuté v tabuľke 6.12.

GloVe		
Kategórie	15 iterácií	30 iterácií
Antonymá (pods. mená)	18,92%	18,71%
Antonymá (príd. mená)	28,46%	33,16%
Antonymá (slovesá)	6,34%	5,80%
Štát-mesto	49,47%	53,65%
Rodinné vzťahy	50,93%	52,78%
Zamestnania	53,28%	54,29%
Národnosť	40,91%	40,44%
Krycie mená	69,27%	69,08%
Slovník synoným (Top 10)	19,07%	20,39%
Slovník synoným (Top 20)	24,38%	25,95%

Tabuľka 6.12: Vyhodnotenie modelov GloVe pre CWC-2011

Pre modely Word2vec bol na testovacej sade slovné analógie zaznamenaný pomerne výrazný pokles úspešnosti oproti korpusu All.vert pre všetky sémantické kategórie, až na rodinné vzťahy a antonymá (slovesá). Pre slovník synoným boli zistené rozdiely od predchádzajúceho experimentu menšie ako 1%. Na týchto testovacích sadách opäť vyhral model CBOW NS. Pre krycie mená si polepšili tieto modely priemerne o 2% a najlepšie sa umiestnil model SG NS. Tabuľka 6.13 zobrazuje úspešnosť modelov Word2vec trénovaných na korpuse CWC-2011.

V prípade modelov FastText bol podobne ako pri klasickom Word2vec zaznamenaný pokles úspešnosti oproti predchádzajúcemu experimentu na väčšine sémantických kategórií testovacej sady slovné analógie. Tento pokles bol ešte výraznejší ako v prípade Word2vec. Zníženie úspešnosti bolo taktiež zistené na sade slovník synoným, kde sa pohyboval od 3% do 6%. Nárast úspešnosti bol naopak zistený na testovacej sade krycie mená, pri ktorej bolo zaznamenané zlepšenie približne o 3% oproti korpusu All.vert. Výsledky pre modely FastText sú zhrnuté v tabuľke 6.14.

Word2vec				
Kategórie	CBOW		SG	
	HS	NS	HS	NS
Antonymá (pods. mená)	11,95%	15,01%	15,58%	13,80%
Antonymá (príd. mená)	25,96%	28,46%	28,16%	25,78%
Antonymá (slovesá)	8,84%	9,73%	7,68%	5,71%
Štát-mesto	23,26%	60,52%	40,02%	60,96%
Rodinné vzťahy	38,27%	48,15%	40,28%	44,75%
Zamestnania	65,32%	83,50%	60,27%	78,62%
Národnosť	51,33%	72,16%	50,85%	74,05%
Krycie mená	69,53%	69,83%	69,86%	72,65%
Slovník synoným (Top 10)	25,76%	27,92%	22,15%	23,77%
Slovník synoným (Top 20)	31,97%	34,69%	27,33%	29,19%

Tabuľka 6.13: Vyhodnotenie modelov Word2vec pre CWC-2011

FastText				
Kategórie	CBOW		SG	
	HS	NS	HS	NS
Antonymá (pods. mená)	7,97%	10,53%	14,08%	11,24%
Antonymá (príd. mená)	14,17%	6,10%	21,78%	16,67%
Antonymá (slovesá)	5,54%	3,66%	2,68%	1,16%
Štát-mesto	16,49%	35,03%	46,61%	62,92%
Rodinné vzťahy	33,49%	35,96%	41,05%	46,14%
Zamestnania	72,73%	87,37%	65,91%	78,96%
Národnosť	62,12%	75,76%	62,31%	82,39%
Krycie mená	67,64%	65,75%	71,99%	73,65%
Slovník synoným (Top 10)	18,90%	15,68%	20,05%	20,25%
Slovník synoným (Top 20)	25,25%	21,36%	25,80%	27,61%

Tabuľka 6.14: Vyhodnotenie modelov FastText pre CWC-2011

6.6 Spojenie korpusov

Cieľom posledného experimentu bolo maximalizovanie úspešnosti pre testovaciu sadu krycie mena. Jedným z najzákladnejších prístupov ako spresniť sémantické modely je zvýšenie objemu trénovacích dát. Za týmto účelom bol použitý korpus, ktorý vznikol spojením Wikipédie, All.vert a CWC-2011. Pre veľkosť trénovacích dát bol v tomto experimente skúmaný už iba jeden model a to Word2vec SG s použitím NS, ktorý dosiahol druhý najlepší výsledok na testovacej sade krycie mená. Tento model bol vybraný pre výrazne nižšiu dobu trénovania oproti modelu FastText SG NS, ktorý obsadil prvé miesto s náskokom iba 1%. Ďalšie spôsoby ako spresniť zvolený model sú podľa [14] a [13] zvýšenie dimenzií vektorov a zväčšenie počtu negatívnych vzoriek. V tomto experimente boli skúmané varianty s 300 a 400 dimenziami a s 5 a 10 negatívnymi vzorkami. Výsledky modelov SG NS sú uvedené v tabuľke 6.15.

Word2vec SG NS				
Kategórie	300 dimenzií		400 dimenzií	
	5 vzoriek	10 vzoriek	5 vzoriek	10 vzoriek
Antonymá (pods. mená)	14,15%	14,44%	13,80%	14,37%
Antonymá (príd. mená)	27,88%	26,13%	28,92%	28,28%
Antonymá (slovesá)	7,05%	6,07%	5,36%	6,25%
Štát-mesto	69,88%	69,07%	72,64%	72,19%
Rodinné vzťahy	47,67%	48,92%	49,08%	48,61%
Zamestnania	80,90%	81,06%	82,15%	79,97%
Národnosť	78,13%	79,17%	78,88%	79,17%
Krycie mená	72,70%	72,27%	72,77%	72,97%
Slovník synoným (Top 10)	25,23%	25,99%	25,89%	26,63%
Slovník synoným (Top 20)	31,01%	31,87%	31,66%	32,68%

Tabuľka 6.15: Vyhodnotenie modelov Word2vec SG NS pre spojené korpusy

Vo vytvorených modeloch bolo zaznamenané zvýšenie presnosti v každej kategórií oproti modelu SG NS trénovanom čisto na korpuse CWC-2011. Najmenší nárast bol nameraný práve na testovacej sade krycie mená, kde sa úspešnosť zvýšila len o desatiny percenta a teda sa nepodarilo prekonať ani model FastText SG NS, ktorý dosiahol najlepší výsledok so $73,65\%$. Dôvodom týchto výsledkov mohlo byť väčšie množstvo pomerne ťažkých sémantických vzťahov, ktoré sa modely nedokázali naučiť na žiadnom z použitých korpusoch. Príkladom takéhoto vzťahu môže byť nápoveda *šest*, na základe ktorej mali byť vybrané krycie mená *brouk* a *pistole*, alebo nápoveda *drak*, na základe ktorej mali byť vybrané krycie mená *podzim* a *louka*. Najväčší nárast úspešnosti bol zaznamenaný v kategórií štát-mesto, kde si tieto modely polepšili až od 9% do 12%. Zvýšenie počtu dimenzií a negatívnych vzoriek malo len malý dopad na výslednú úspešnosť.

6.7 Analýza výsledkov

V tejto podkapitole sú analyzované výsledky vybraných modelov pre testovacie sady krycie mená a slovník synoným. Táto analýza je vykonaná na základe dodatočných informácií, ktoré odpovedajúce vyhodnocovacie skripty poskytujú.

6.7.1 Krycie mená

Skript na vyhodnotenie krycích mien obsahuje rozšírený mód výstupu, pri ktorom sú poskytnuté informácie, ktoré môžu byť použité k vysvetleniu výberu krycích mien pre jednotlivé nápovedy. Tento mód je demonštrovaný na jednej vybranej nápovede z množiny riadkov, ktoré dosiahli menšiu priemernú presnosť ako 1. Tento výber bol uskutočnený pomocou generátora pseudonáhodných čísel. Analýza je vykonaná na modeli z najvyššou úspešnosťou a to FastText SG NS trénovanom na korpuse CWC-2011.

Nápoveda: dřevo, AP = 0,738		
Krycie mená	Kosínusová vzdialenosť	Vážený počet spoluvýskytov
uhlí	**0,5113**	**598,0666**
strom	**0,4332**	**329,4163**
hlína	0,4269	138,2167
kámen	0,4193	506,8330
tráva	0,3553	86,4500
maso	0,3360	36,4667
dveře	0,3108	244,2332
vzduch	0,2778	54,6000
ještěrka	0,2526	1,0333
mouka	0,2249	6,9333
lžíce	0,2210	5,3667
limonáda	0,1970	0,0000
země	0,1949	186,3167
smůla	**0,1889**	**20,7500**
pohádka	0,1849	3,6833
obchod	0,1703	66,2834
počítač	0,1577	11,0500
kouzlo	0,1572	11,3000
myš	0,1399	9,3000
míč	0,1271	2,9000
Španěl	0,1204	0,0000
hra	0,0981	21,0500
cizinec	0,0945	1,7833
Moskva	0,0837	0,3333
planeta	0,0662	2,4000

Tabuľka 6.16: Podobnosť nápovedy *dřevo* a daných krycích mien

V tabuľke 6.16 sú zobrazené výsledky pre nápovedu *dřevo*. Tučne vyznačené sú krycie mená, ktoré mali byť na jej základe vybrané. V tomto modeli správne krycie meno *smůla* obsadilo iba 14. priečku. Toto môže byť spôsobené pomerne nízkym váženým počtom spoluvýskytov, ktorý udáva že sa v danom korpuse tieto slová príliš často v rovnakom kontexte nenachádzali.

Rozšírený mód taktiež poskytuje informácie o sémanticky najviac podobných slovách pre priemer vektorov jednotlivých dvojíc tvorených nápovedou a krycím menom. Ide teda

o slová, ktoré sú podobné nápovede, aj kryciemu menu a určujú teda sémantický smer tejto dvojice. V tabuľke 6.17 sú zobrazené najviac podobné slová pre dvojicu *dřevo* a *smůla*. V tabuľke 6.18 ide o dvojicu slov *dřevo* a *hlína*, ktorá mala vysokú hodnotu spoluvýskytu, aj keď na prvý pohľad tieto slová spolu nesúvisia.

dřevokotel
dřevoštěpkový
naštípání
dřevjený
topivo
dřevomor
dřevotříska
smolnatý
briketky
dřívi

Tabuľka 6.17: Najviac sémanticky podobné slová pre priemer vektorov slov *dřevo* a *smůla*

zemina
dřevoštěpkový
hlína
mulčovací
štípaný
dřevotříska
kamení
briketky
dřevíčko
pískovec

Tabuľka 6.18: Najviac sémanticky podobné slová pre priemer vektorov slov *dřevo* a *hlína*

V prípade dvojice *dřevo* a *smůla* sú až na *dřevokotel* a *topivo* všetky slová relevantné. Slovo *dřevotříska* napríklad označuje časti dreva, ktoré sú spojené syntetickou smolou. Lisovanie brikiet funguje na základe spojenia častí dreva pomocou prírodnej smoly. Dokonca aj slová *dřevokotel* a *topivo* nepriamo súvisia s briketami a drevotrieskou, alebo dokonca môžu vychádzať z horľavosti ako dreva, tak aj smoly. Podľa týchto výsledkov sa javí, že model mohol zachytiť význam tejto dvojice aj napriek nízkemu umiestneniu tohto kryciehо mena. Nízke umiestnenie môže byť taktiež spojené s druhotným významom slova *smůla*, ktoré znamená nešťastie.

Skúmaním najviac podobných slov pre dvojicu *dřevo* a *hlína* je možné dospieť k záveru, že tento model zachytil ich podobnosť v oblasti záhradníctva. Všetky slová, okrem *dřevotříska* a *briketky*, totiž súvisia s dekoráciu záhradky, mulčovaním alebo tvorbou kompostu. V prípade slova *štípaný* sa predpokladá, že ide o časť výrazov *štípané dřevo* alebo *štípaný kámen*, ktoré s touto tematikou priamo súvisia. Je taktiež možné, že *dřevotříska* do tejto skupiny patrí, lebo prvý krok jej výroby je štiepanie. Vďaka analýze priemeru vektorov tejto dvojice je jasnejšie prečo sa nielen krycie meno *hlína*, ale aj ďalšie krycie mená *kámen* a *tráva*, ktoré taktiež súvisia s touto kategóriou, umiestnili na vysokých priečkach podobnosti pre nápovedu *dřevo*.

6.7.2 Slovník synoným

Najlepší výsledok typu top 10 na testovacej sade slovník synoným dosiahol model Word2vec CBOW NS trénovaný na korpuse All.vert s $28,43\%$. Dôvodom pomerne nízkej úspešnosti je to, že synonymá reprezentujú len jednu oblasť sémantickej podobnosti, ktorú skúmané distribučné modely zachytávajú. Pre drastické zvýšenie úspešnosti na tejto testovacej sade by musel byť použitý korpus špecializovaný pre tento účel. Analýza tejto testovacej sady je vykonaná vyhodnotením relevantnosti sémanticky najviac podobných slov pre 2 riadky vybrané pseudonáhodným generátorom čísel.

reklama, Top 10 = 0,0	
Synonymá	Kosínusová vzdialenosť
reklamní	0,6921
inzerce	0,6737
spot	0,6165
billboard	0,6028
propagace	0,5729
barnumský	0,5708
inzerent	0,5653
antireklama	0,5577
upoutávka	0,5561
proužkový	0,5499

Tabuľka 6.19: Najviac sémanticky podobné slová pre slovo *reklama*

zubní, Top 10 = 0,33	
Synonymá	Kosínusová vzdialenosť
dentální	0,6030
asciutta	0,5607
zubař	0,5542
zubařský	0,5283
stomatolog	0,5261
dermatologický	0,5240
protetika	0,5192
aquafresh	0,5164
paradentóza	0,5138
parodontóza	0,5055

Tabuľka 6.20: Najviac sémanticky podobné slová pre slovo *dřevo*

Pre slovo *reklama* v slovníku synoným boli zadané ako správne synonymá *vychvalování* a *doporučení*. Z týchto synoným sa ani jedno nenachádzalo v top 10 sémanticky najviac podobných slov podľa skúmaného sémantického modelu, čo je možné vidieť v tabuľke 6.19. Podľa osobného vyhodnotenia je však 7 z 10 určených slov definitívne relevantných k slovu reklama. Navyše anglické slovo *spot* sa veľmi často používa v spojení *reklamný spot* a teda sa dá tiež označiť ako relevantné. Podobne to je aj v prípade slova *barnumský*. Slovo *proužkový* je teda podľa môjho názoru jediné z nájdených slov, ktoré so slovom *reklama* vôbec nesúvisí.

Pre slovo *zubní* sa v slovníku nachádzali synonymá *stomatologický*, *zubařský* a *dentistický*. Sémanticky najviac podobné slová podľa daného modelu je možné vidieť v tabuľke 6.20. V tomto prípade je relevantných 7 z 10 slov: *dentální*, *zubař*, *zubařský*, *stomatolog*, *aquafresh*, *paradentóza* a *parodontóza*. Z týchto slov nemusí byť známe slovo *aquafresh*, ktoré označuje značku zubných pást. Zaujímavé je, že model sa v podstate trafil do všetkých synoným, pričom rozdiel oproti správnym slovám bol iba ich v tvare. Do sémanticky nesúvisiacich slov patrí *asciutta*, čo je talianske slovo, ktoré znamená suchý. Toto slovo sa však používa s spojení *pasta asciutta*, čo voľne preložené znamená cestoviny. Javí sa však, že sémantický model toto slovo mylne interpretoval ako zubnú pastu a preto sa nachádza vo vyhľadaných slovách. *Protetika* je názov predajcu zdravotných pomôcok a teda iba okrajovo súvisí s touto tematikou. Slovo *dermatologický* označuje úplne iný obor zdravotníctva.

6.8 Zhrnutie výsledkov

V rámci tejto práce bolo spolu vyhodnotených 36 sémantických modelov, ktoré boli trénované na troch rôznych korpusoch. V rámci testovania bolo dokázané, že lematizácia pre česky jazyk výrazne zlepšuje presnosť sémantických modelov a teda zvolené korpusy boli lematizované. Výhercovia jednotlivých kategórií sú uvedení v tabuľke 6.21.

Model GloVe dosiahol výrazne najlepšiu úspešnosť na korpuse CWC-2011. Z toho výsledku sa javí, že funguje lepšie pre rozsiahlejšie korpusy, kde môže efektívne využiť štatistické informácie z vytvorenej matice spoluvýskytu. Keďže doby trénovania GloVe modelu s 15 iteráciami sú okolo dvakrát nižšie ako pre Word2vec a tri až štyrikrát nižšie ako pre FastText, výsledky toho modelu sú pre korpusy All.vert a CWC-2011 pomerne uspokojivé.

Kategórie	Úspešnosť	Model
Antonymá (pods. mená)	23,04%	Word2vec CBOW NS - All.vert
Antonymá (príd. mená)	38,91%	Word2vec CBOW HS - Wikipédia
Antonymá (slovesá)	10,90%	FastText CBOW NS - Wikipédia
Štát-mesto	78,79%	Word2vec SG NS - All.vert
Rodinné vzťahy	52,78%	GloVe 30 iterácií - CWC-2011
Zamestnania	87,37%	FastText CBOW NS - CWC-2011
Národnosť	82,39%	FastText SG NS - CWC-2011
Krycie mená	73,65%	FastText SG NS - CWC-2011
Slovník synoným (Top 10)	28,43%	Word2vec CBOW NS - All.vert
Slovník synoným (Top 20)	34,69%	Word2vec CBOW NS - CWC-2011

Tabuľka 6.21: Najúspešnejšie modely pre jednotlivé kategórie

Rodina modelov Word2vec dosahuje celkovo najlepšie výsledky v sémantických kategóriách slovných analógií a na testovacej sade slovník synoným. V týchto kategóriách má celkovo najvyššiu úspešnosť model CBOW NS. Pre krycie mená naopak funguje najlepšie model SG NS, ktorý marginálne zaostáva na tejto sade oproti svojej alternatíve FastText SG NS.

Modely FastText konzistentne víťazia v syntaktických kategóriách slovných analógií, čo je vďaka ich architektúre očakávané. Najväčší rozdiel v týchto kategóriách bol pri korpuse Wikipédia, kde FastText niekoľkonásobne prekonáva ostatné modely. Pre sémantické kategórie slovných analógií, slovník synoným a krycie mená FastText dosahuje podobné výsledky ako Word2vec, pričom sa javí, že čím rozsiahlejší je korpus, tým architektúra modelov FastText dosahuje mierne horšie výsledky oproti klasickému Word2vec.

Kapitola 7

Záver

V rámci tejto práce bol v jazyku Python navrhnutý a implementovaný systém, ktorý dokáže k zadanému slovu nájsť a zobraziť slová sémanticky príbuzné. Tento systém sa skladá z niekoľko vzájomne prepojených modulov, ktoré slúžia na predspracovanie vstupného textového korpusu, trénovanie distribučných sémantických modelov, vyhodnotenie vytvorených modelov a vyhľadávanie najviac podobných slov. Systém v rámci predspracovania implementuje možnosť lematizácie pre český jazyk. Pri trénovaní umožňuje výber jedeného z modelov GloVe, Word2vec, alebo FastText. V rámci vyhodnotenia systém poskytuje skripty na určenie úspešnosti modelov s použitím testovacích sád krycie mená a slovník synoným, ku ktorým poskytuje základné vysvetľovacie režimy. Ďalej sú taktiež dostupné 2 rôzne skripty na určovanie sémantickej podobnosti slov.

Táto práca prináša vyhodnotenie a porovnanie populárnych distribučných sémantických modelov pre česky jazyk. Skúma ich úspešnosť v závislosti na rôznych architektúrach, parametroch a zvolených korpusoch na uvedených testovacích sadách.

Vďaka modularite systému je otvorený rôznym modifikáciám a vylepšeniam. Jedným z vylepšení môže byť využitie vybraných informácií, ktoré poskytujú POS značky behom trénovania modelov. Vďaka využitiu knižnice MorphoDiTa je možné túto modifikáciu veľmi jednoducho implementovať. Ďalšie rozšírenia by mohli mať podobu rozšírenia pre určovanie syntaktickej podobnosti celých viet alebo implementácie ďalších distribučných sémantických modelov. V rámci vyhodnotenia bolo taktiež určené, že sémantické modely dosahujú dostatočnú úspešnosť na to, aby mohli byť použité ako nápoveda pre spoločenskú hru krycie mená, alebo dokonca využité pri implementácií automatického spoluhráča alebo oponenta.

Literatúra

[1] Balakrishnan, V.; Lloyd-Yemoh, E.: Stemming and lemmatization: a comparison of retrieval performances. *Lecture Notes on Software Engineering*, ročník 2, č. 3, 2014: str. 262.

[2] Baroni, M.; Dinu, G.; Kruszewski, G.: Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *ACL (1)*, 2014, s. 238–247.

[3] Bird, S.; Klein, E.; Loper, E.: *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc., 2009, ISBN 978-0-596-51649-9, ix–x s.

[4] Bojanowski, P.; Grave, E.; Joulin, A.; aj.: Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016.

[5] Dash, N. S.: Corpus linguistics: An introduction. 2008, [Online; navštívené 12.04.2017].
URL http://www.ldcil.org/download/Corpus%20Linguistics.pdf

[6] Fabre, C.; Lenci, A.: Distributional Semantics Today Introduction to the special issue. *Traitement Automatique des Langues*, ročník 56, č. 2, 2015: s. 7–20.

[7] Fox, C. J.: Lexical Analysis and Stoplists. 1992, [Online; navštívené 12.04.2017].
URL http://orion.lcg.ufrj.br/Dr.Dobbs/books/book5/chap07.htm

[8] Ha, L. Q.; Sicilia-Garcia, E. I.; Ming, J.; aj.: Extension of Zipf's law to words and phrases. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, Association for Computational Linguistics, 2002, s. 1–6.

[9] Harris, Z. S.: Distributional structure. *Word*, ročník 10, č. 2-3, 1954: s. 146–162.

[10] Katariya, M. N. P.; Chaudhari, M.; Subhani, B.; aj.: Text preprocessing for text mining using side information. *International Journal of Computer Science and Mobile Applications*, ročník 3, č. 1, 2015: s. 01–05.

[11] Levy, O.; Goldberg, Y.; Ramat-Gan, I.: Linguistic Regularities in Sparse and Explicit Word Representations. In *CoNLL*, 2014, s. 171–180.

[12] Mandelbaum, A.; Shalev, A.: Word Embeddings and Their Use In Sentence Classification Tasks. *arXiv preprint arXiv:1610.08229*, 2016.

[13] Mikolov, T.; Chen, K.; Corrado, G.; aj.: Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[14] Mikolov, T.; Sutskever, I.; Chen, K.; aj.: Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, 2013, s. 3111–3119.

[15] Pennington, J.; Socher, R.; Manning, C. D.: Glove: Global Vectors for Word Representation. In *EMNLP*, ročník 14, 2014, s. 1532–1543.

[16] Riordan, B.; Jones, M. N.: Redundancy in perceptual and linguistic experience: Comparing feature-based and distributional models of semantic representation. *Topics in Cognitive Science*, ročník 3, č. 2, 2011: s. 303–345.

[17] Rong, X.: word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*, 2014.

[18] Spoustová, J.; Spousta, M.: CWC2011. 2012, LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University. URL http://hdl.handle.net/11858/00-097C-0000-0006-B847-6

[19] Svoboda, L.; Brychcín, T.: New word analogy corpus for exploring embeddings of Czech words. *arXiv preprint arXiv:1608.00789*, 2016.

[20] Widdows, D.; Cohen, T.: The semantic vectors package: New algorithms and public tools for distributional semantics. In *Semantic computing (icsc), 2010 ieee fourth international conference on*, IEEE, 2010, s. 9–15.