FAKULTA INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Název:	Nástroj pro analýzu poslaneckých projevů
Student:	Jan Horyna
Vedoucí:	Ing. Jaroslav Kuchař, Ph.D.
Studijní program:	Informatika
Studijní obor:	Webové a softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2020/21

Pokyny pro vypracování

Cílem práce je analyzovat projevy poslanců v Poslanecké sněmovně za pomocí technik NLP. Součástí práce bude návrh a implementace modulů, které získají surová data z webu, následně je zpracují a poté prezentují zpracované informace.
- Seznamte se s problematikou extrakce informací z webu a textu.
- Prozkoumejte již existující zdroje dat a nástroje, které mohou pomoci se základním zpracováním a analýzou dat.
- Navrhněte, implementujte a otestujte proces získávání a ukládání dat.
- Navrhněte a implementujte způsob, který využívá zejména existující přístupy či nástroje na analýzu textu, a aplikujte ho na poslanecké projevy.
- Vytvořte modul pro prezentaci získaných informací.
- Na výsledném řešení zobrazte reálné informace o projevech poslanců v Poslanecké sněmovně.

Seznam odborné literatury

Dodá vedoucí práce.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
děkan

V Praze dne 7. února 2020

FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE

Bakalářská práce

Nástroj pro analýzu poslaneckých projevů

Jan Horyna

Katedra softwarového inženýrství
Vedoucí práce: Ing. Jaroslav Kuchař, Ph.D.

31. května 2020

Poděkování

Velmi děkuji vedoucímu své bakalářské práce Ing. Jaroslavu Kuchařovi Ph.D. za skvělé vedení, poskytnutí odborného vysvětlení k dané problematice, lidský přístup a výborné rady, které mne vždy nasměrovaly ke správnému řešení. Dále bych chtěl poděkovat kamarádu Petrovi Větrovskému za konzultování a podporu. Vyzdvihl bych moji maminku Mgr. Petru Horynovou, která mi ochotně pomohla s korekturou, za což jí moc děkuji. Děkuji také rodině a přátelům, kteří mě podporovali, někteří i výsledky práce zkoušeli a poskytli cennou zpětnou vazbu.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo"), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, avšak pouze k nevýdělečným účelům. Toto oprávnění je časově, teritoriálně i množstevně neomezené.

V Praze dne 31. května 2020 . . . . . .

České vysoké učení technické v Praze
Fakulta informačních technologií

Odkaz na tuto práci

Horyna, Jan. *Nástroj pro analýzu poslaneckých projevů.* Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2020.

Abstrakt

Dnešní doba se označuje jako doba informační. Informace jsou velmi cenné, je jich ale tolik, že dochází k informačnímu zahlcení. V politice mají navíc různé strany a hnutí na stejnou věc různé názory, takže se množství informací ještě násobí. Tato práce má za cíl vytvořit nástroj, který občanům pomůže zorientovat se v tom, jak jednotliví poslanci a strany mluví v Poslanecké sněmovně Parlamentu ČR, a uvést tyto projevy do kontextu projevů ostatních stran nebo i jich samých a jejich minulosti.

Pro splnění tohoto úkolu byly vytvořeny tři části implementované v jazyce Java. První z nich má za cíl získat data poslaneckých projevů z webu Poslanecké sněmovny a předzpracovat je pro projekt druhý. Druhý projekt přijme poslanecké projevy, které zpracuje a zanalyzuje pomocí nástroje MorphoDiTa. Výstup pak ukládá spolu s dalšími informacemi jako jsou statistiky a podobně do databáze, která byla vytvořena pro tuto práci. Třetí projekt zodpovídá za prezentaci informací z databáze pro uživatele. Prezentace je zde realizována pomocí nástroje Vaadin, poskytující různé textové i grafové pohledy na zpracovaná data. Poslance, osoby, strany nebo i celá volební období lze porovnávat pro získání lepšího kontextu.

Klíčová slova poslanecké projevy, zpracovaní přirozeného jazyka, dolování dat z webu, analýza textu, český jazyk, webová aplikace

Abstract

Nowadays we live in the Information Age. Information is very valuable but there is so much of it therefore the risk of information overload does more often appear. In politics it is more complicated due to many political parties and each party has its own opinions. This thesis has an ambition to build a tool which purpose is to help citizens to orientate in speeches of members and parties in the Chamber of deputies Parliament of the Czech republic in the context of other speeches.

Three Java projects have been built in order to accomplish this task. The first has the purpose to download data from the parliament's website and prepare it for the second project which receives the data as an input. Subsequently it processes speeches and analyzes them in the tool MorphoDiTa. Finally the project transfers analyzed data together with other information and statistics to for this reason developed database. The third project performs information from the database to users. The presentation is implemented in the tool Vaadin which visualizes analyzed data in text and graph format. Members of parliament, parties or whole election periods can be compared to obtain improved context.

Keywords speeches of deputies, natural language processing, web mining, text analysis, czech language, web application

Obsah

Seznam obrázků

Úvod

Žijeme v době informačního zahlcení. K informacím se často už ani nesnažíme dostat, ale ony k nám ve velkém množství pronikají samy a my už je dokonce musíme filtrovat. Není tedy moc času na jejich analyzování a upřednostňujeme, když zpracování a zasazení do kontextu provede někdo jiný, kdo se danému tématu věnuje podrobněji.

Nedílnou součástí veřejného dění je politika. Právě z oblasti politiky máme mnoho informací a ještě jsou stejné zprávy vykládány z různých stran, jak to vykládají různé politické skupiny. Můžeme se v tom tedy ztrácet (což možná některým i vyhovuje). Nejdůležitějším politickým orgánem u nás je Poslanecká sněmovna Parlamentu České republiky, kde se hlavně rozhoduje o podobě zákonů. Strany a hnutí zde mají přístup k tomu, aby se jejich poslanci vyjadřovali k návrhům zákonů a jiným věcem. Právě tyto poslanecké projevy jsou ideálním zdrojem pro analytické zpracování, na které se zaměřuje tato práce.

Podstatou této práce je tedy vytvořit nástroj na analýzu poslaneckých projevů, který by pomohl lidem vyznat se v politické situaci. Práce a její výsledek v podobě nástroje je tedy určen široké veřejnosti, která si chce rozšířit obzory a kontext projevů, které poslanci pronášejí.

Téma mé bakalářské práce jsem si zvolil, protože si myslím, že výše uvedený nástroj v rámci české politické scény neexistuje a mohl by být prospěšný všem lidem v našem státě. Osobně se o politiku snažím zajímat a podobný nástroj mi chybí, takže k jeho vytvoření mám i tuto vlastní motivaci.

Práce jako celek se věnuje analýze, návrhu a implementaci řešení nástroje, který získá, zpracuje a nakonec zobrazí informace o projevech poslanců. První kapitola se věnuje cílům, kterých je třeba dosáhnout k úspěšnému dokončení této práce. Druhá kapitola se zaměřuje na teoretickou část, jde tedy o zmapování zdrojů a jejich teoretické popsání pro pozdější praktické použití v této práci. Třetí kapitolou je praktická část. Zde je probrána konkrétní analýza a implementace nástroje. Je tady také popsáno použití nástrojů, které byly

představeny v druhé kapitole. Třetí kapitola je následně rozdělena podle jednotlivých segmentů nástroje, kde se každá část stará o konkrétní úkol. První část se stará o stahování dat, druhá zodpovídá za analýzu dat a uložení jejich zpracované formy do úložiště, z kterého je pak třetí část prezentuje pro uživatele.

Kapitola 1

Cíl práce

Hlavním cílem této bakalářská práce je analyzovat projevy poslanců pronesené v Poslanecké sněmovně Parlamentu České republiky. Pro splnění hlavního cíle musí být splněny následující dílčí cíle.

Cílem teoretické části je analyzovat existující řešení v oboru zpracování poslaneckých projevů. Následujícím dílčím cílem je seznámit se s postupy v oblasti extrakce dat z webu a textu. Dalším dílčím cílem je prohledání existujících zdrojů dat a nástrojů, které data z těchto zdrojů dokáží zpracovat. Posledním dílčím cílem teoretické části je nalezení vhodného nástroje na prezentaci dat v podobě textu a grafů.

Cílem praktické části je navrhnout, implementovat a otestovat nástroj, který umožní zpracovat původní poslanecké projevy, vytvořit z nich zanalyzovanou formu a poté prezentovat získané informace v podobě webové aplikace. Otestování systému proběhne na reálných datech z Poslanecké sněmovny.

Kapitola 2

Teoretická část

2.1 Poslanecká sněmovna Parlamentu ČR

Česká republika se řadí mezi parlamentní republiky. Parlament zde má velmi důležitou roli. Parlament má zákonodárnou moc, stará se tedy o vytváření a schvalování zákonů. Ukotvení parlamentu v našem právním systému můžeme nalézt už v nejdůležitějším zákonu našeho státu, v Ústavě České republiky [1]. Moci zákonodárné se věnuje Hlava druhá. Parlament se skládá ze dvou částí – Poslanecké sněmovny a Senátu.

Poslanecká sněmovna Parlamentu České republiky je zákonodárný orgán, zákony zde projednává 200 volených zástupců – poslanců. Poslanci jsou voleni na dobu 4 let a jedná se o volbu přímou. Poslancem se může stát občan České republiky, který dosáhl věku 21 let a má právo volit [1]. Poslanci se starají o podobu přijímaných zákonů a také vyslovují důvěru vládě. Vláda zastává v našem systému moc výkonnou.

Poslanci se scházejí v různých sněmovních orgánech – výbory, podvýbory, komise a další. V poslaneckých klubech se scházejí poslanci jedné politické příslušnosti nebo společného zájmu [2]. Všichni poslanci se scházejí na schůzích sněmovny. Tyto schůze jsou svolávány předsedou Poslanecké sněmovny. Na těchto schůzích poslanci zákony projednávají, vyjadřují se k nim a nakonec o nich i hlasují. Tyto plenární schůze se řídí Jednacím řádem [3]. Jednací řád stanovuje, jak mohou jednotliví poslanci vystupovat na plénu. Tato pravidla k vystupování platí pro všechny poslance stejně a právě díky těmto pravidlům jsou tyto projevy vhodné pro analyzování.

2.1.1 Popis dat

Poslanecká sněmovna generuje mnoho dat a všechna zveřejněná data jsou v digitální knihovně [4] této instituce. Digitální knihovna je hodně obsáhlá a lze zde nalézt i zápisy ze Sněmů království Českého. Tato práce ale využívá novější

záznamy a to stenoprotokoly [5] z Poslanecké sněmovny ČR („*Stenoprotokoly (těsnopisecké zprávy): doslovné záznamy z jednání schůzí Parlamentu*“ [6]).

Stenoprotokoly jsou rozděleny podle volebních období, schůzí a jednacích dnů. Hlavní struktura stenoprotokolu se skládá ze 2 částí. První částí jsou jednací body. Body jsou jasně viditelné a oddělují témata, která poslanci rozebírají. Druhou částí jsou promluvy jednotlivých poslanců, ministrů a dalších osob, které mohou na plénu vystoupit. Samotný výstup řečníka se skládá z jeho jména a jeho řeči.

Ke stenoprotokolům jsou ještě dostupné seznamy (tabulky) osob, poslaneckých mandátů, stran,... [7]. Tyto seznamy následně slouží jako základ databáze pro uchovávání dat v rámci této práce. Seznamy jsou totiž jistou formou výpisu z tabulek databáze, která se používá v Poslanecké sněmovně. Lze zde tedy identifikovat různé vazby mezi entitami a podobně.

2.2 Jiné existující zdroje dat

Existuje mnoho prací o zpracování textu jako takového. Minimum se jich ale věnuje zpracování politických projevů v českém prostředí.

2.2.1 Hlídač státu

Hlídač státu je nezisková organizace, která si klade za cíl kontrolovat hospodaření státu, smlouvy, registry a tak podobně. Dalším důležitým prvkem tohoto známého projektu je získaná data propojovat a prezentovat ve srozumitelné podobě pro všechny občany [8].

Na Hlídači státu jsou poslanecké projevy zpracovány na stránce *Databáze Stenozáznamy Poslanecké sněmovny Parlamentu ČR* [9]. Databáze poskytuje velké množství dat rozdělených do jednotlivých projevů. Ke každému projevu jsou navíc dostupné doplňkové informace jako délka v minutách, odkaz na řečníka, zdroj a další. Tento zdroj nakonec nebyl použit v této práci, protože forma zpracování dat nebyla v detailech úplně vhodná.

Pro příklad lze uvést zmiňování se řečníků navzájem. Řečníci se ve svých projevech zmiňují a tato informace je zaznamenána u projevu. Problém ale je u různých řečníků se stejným jménem. Zcela konkrétně bývalý prezident Václav Klaus a jeho syn poslanec Václav Klaus mladší jsou vždy zmiňováni jako dvojice, protože nelze určit, kterého z nich řečník myslel. Oproti tomu v této bakalářské práci se zaznamenává zmínka řečníků navzájem jen na poslance ve stejném volebním období, což možnost záměny hodně snižuje. Konkrétně shoda příjmení u více různých osob se v historii poslanců Poslanecké sněmovny vyskytuje v 81 případech, ale shod ve stejném volebním období je 47. Hlídač státu se také v rámci zmínek nezaměřuje jen na poslance, ale obecně na politiky, takže je možné, že je shod ještě více. V prezentační části tohoto projektu je také navíc funkce, která upozorní uživatele na to, když vybere někoho z poslanců ze shodným příjmením. Dále Hlídač státu využívá

pro identifikaci poslance nebo jiné osoby jako identifikátor textový řetězec. Pro tuto práci se lépe pracovalo s číselnými identifikátory, které jsou pevně dané už v tabulkách na stránkách Poslanecké sněmovny [7]. Na tyto identifikátory lze tedy rovnou navázat další informace z těchto tabulek, kde už jsou přímo definované třeba návaznosti na kandidátky, na kterých byli poslanci voleni, což v datech od Hlídače státu není. Při teoretickém převodu jednoho typu identifikátorů na druhý by byl znovu problém u lidí, kteří mají stejná jména. Na Hlídači státu se také zaměřují na všechny projevy v rámci pléna Poslanecké sněmovny. Tato práce se zaměřuje jen na projevy poslanecké, jejich vyfiltrování by ale asi byl řešitelný problém. Jedním z hlavních důvodů vzniku dvou různých přístupů je ale to, že zpracování na Hlídači státu se vytvářelo ve stejné době jako tato práce a první výstupy projektu Hlídače státu byly přístupné až v době psaní této práce. I proto nebylo tak docela možné z výstupů Hlídače státu přímo vycházet.

Zpracování domény projevů v Poslanecké sněmovně vypracované Hlídačem státu je ale celkově zajímavé a výhledově se nabízí možnost propojit s výstupy této práce. V této práci byl třeba inspirací pohled na výše zmiňované zmínky poslanců mezi sebou nebo detaily při členění dat v datových strukturách.

2.3 Web mining

Web mining (dolování dat z webu) [10] je vytěžování potřebných dat z webových dokumentů a služeb. Důležité je, že tento proces je automatizovaný. V případě této práce jde o zpracovávání HTML (Hypertext Markup Language) stránek.

2.4 Zpracovaní přirozeného jazyka (NLP)

NLP (Natural Language Processing), v češtině zpracování přirozeného jazyka, je obor, který se pohybuje na rozhraní matematiky, informatiky a lingvistiky [11]. Cílem tohoto oboru je vytvořit nástroje, které budou rozumět lidské řeči. Rozumět ve smyslu, že alespoň v omezené míře budou chápat lidský jazyk, jeho kontext a další důležité aspekty.

2.4.1 Sentiment

Analýza sentimentu [12], často také jako „dolování názoru“ (opinion mining), je obor, který se zabývá extrahováním názoru nebo emocí autora k textu. V tomto oboru narážíme na spoustu problému. Každý člověk používá trochu jiný slovník a stejná slova jsou pro různé lidi jinak důrazná nebo tvrdá a používají je v jiných situacích. Problémy se samozřejmě násobí, když se přidají ještě různé jazyky, kde mohou být ještě ne úplně jednoznačné překlady.

Dalším problémem může být kontext. Jako příklad si lze vzít třeba větu „*Rve se za jejich práva*“. Kdyby se rozebíralo pouze slovo po slově a slova byla převedena na základní tvary, tak takové nejsilnější slovo je „*Rvát (se)*“. To by ukazovalo jednoznačně jakýsi negativní sentiment. V kontextu celé věty už to ale nezní tak negativně a pak ještě záleží v jakém kontextu byla použita celá věta. Bohužel kontextové hledání sentimentu je složité, a proto používáme i bezkontextové hledání sentimentu.

Příkladem bezkontextového hledání sentimentu je přístup hledání ve slovníku. Na začátku je vytvořen slovník, kde ke každému slovu ve slovníku je přiřazen nějaký sentiment podle toho, jakou používáme reprezentaci sentimentu. Lze použít interval reálných čísel, binární označení negativních a pozitivních slov nebo jiná značení. Když poté probíhá analýza, tak se analyzuje po jednotlivých slovech v textu a pro každé slovo se kontroluje, zda není ve slovníku. Pokud je, přiřadí se mu příslušný sentiment.

2.4.1.1 Slovník SubLex

Slovník SubLex Czech 1.0 [13] obsahuje přes 4500 slov. Každé slovo má přiřazen pozitivní nebo negativní sentiment. Nástroj je licencován pod licencí, která umožňuje dílo dále sdílet a upravovat [14].

2.4.2 Redukce dimenzionality v rámci NLP

Pojem redukce dimenzionality (dimensionality reduction) [15] označuje proces, který má za úkol zmenšit (redukovat) složitost nějaké složité domény za předpokladu zachování důležitých informací. V rámci práce s dlouhými texty to lze upřesnit jako proces, který se snaží z textu získat to důležité. K tomuto cíli vedou různé metody a všechny mají své využití pro konkrétní problémy. Některé z těchto metod jsou popsány v následujících podsekcích.

2.4.2.1 Tokenizace

Tokenizací (tokenization) [16] je označen proces, kde nějaký delší kus textu, složený ze slov, rozdělujeme na slova, interpunční znaménka případně další. V tomto případě jde tedy o „word tokenization“ – tokenizace na slova. Existuje ale třeba i „sentence tokenization“ – tokenizace na věty. Tokenizace jako vstup dostane text a jako výstup odevzdá seznam tokenů [17].

2.4.2.2 Lemma

Český jazyk je známý svými mnohými tvary. Máme docela hodně jmenných pádů a například slovesa mají také mnoho osob, časů, čísel a podobně. V rámci zpracování přirozeného jazyka to může dělat problémy a pro většinu operací by bylo vhodnější, kdyby všechna slova byla v základním tvaru. Tedy pro příklad podstatné jméno – 1. pád a jednotné číslo, sloveso v infinivu a tak

podobně. Tento proces se označuje jako lemmatizace (lemmatizing) [16], [18]. Lemmatizaci je podobný proces zvaný stematizace (stemming) [19]. Stematizace stejně jako lemmatizace zpracovává jednotlivá slova. Rozdíl je ale ve výstupu tohoto procesu, u lemmatizace je výstupem základní tvar slova, stematizace poskytuje kořen slova. Tyto dva postupy jsou si tedy dost podobné. V této práci je stematizace zmíněna hlavně z důvodu širšího kontextu a v této práci se aktivně nevyužívá.

2.4.2.3 Morfologický tag

Morfologický tag (morphological tag) se často uvádí ve dvojici s lemmatem a tvoří spolu jako dvojice výstup morfologické analýzy, kde lemma označuje výše popsaný základní tvar a tag má uloženy informace o tvaru původním. Informace jsou ovšem na rozdíl od tvaru původního jednoduše strojově zpracovatelné. Tag je typicky textový řetězec o pevně dané délce, kde každý znak odpovídá některému prvku morfologické analýzy. Například 1. znak může uchovávat informaci o slovním druhu, 2. o čísle, 3. o osobě a tak dále.

2.4.2.4 Klíčová slova

Klíčové slovo (keyword) je slovo, které zastupuje a vystihuje význam jiného slova, věty nebo projevu – „*Keyword – a word that serves as a key, as to the meaning of another word, a sentence, passage, or the like.*“ [20]. Klíčová slova tedy lze použít třeba v případě, kdy se analyzuje dlouhý text a je potřeba zachytit z něj to nejpodstatnější. Člověk, který rozumí určitému textu, pro něj dokáže určit klíčová slova. V rámci počítačového zpracování je to složitější. Z počátku by se mohlo zdát, že klíčovými slovy by mohla být nejčastěji se vyskytující slova. To bohužel není tak docela pravda. Nejčastější slova v nějakém projevu jsou často jen jazykovou podporou pro slova, která reálně nesou význam. Často se třeba opakují spojky, zájmena, předložky a podobně, ale většinou tato slova žádný význam nemají. Tato „dopňková“ slova se označují jako stopslova (stopwords).

2.5 Nástroje a technologie

V této sekci jsou popsány nástroje, které byly v práci použity.

2.5.1 Jazyk Java

Celá praktická část této práce je psaná v jazyce Java, tudíž je vhodné tento široce rozšířený nástroj alespoň krátce představit.

Java [21], [22] je jazyk, který vznikal v 90. letech 20. století. Java si bere inspiraci z jazyka C++, některé věci zjednodušuje a jiné vylepšuje. Velkou

výhodou Javy je snadná přenositelnost a nezávislost na platformě. Pro programy napsané v Javě se využivá JVM (Java Virtual Machine). Programy napsané pro Javu se tedy nepřekládají přímo do strojového kódu pro příslušný typ procesoru, ale místo toho se použije takzvaný mezikód (bytecode), který je právě spustitelný v JVM.

Spustitelnost na velkém množství zařízení je tedy velkým kladem, je ale vykoupena určitou pomalostí oproti konkurenci, která mezikód nevyužívá. Firma Sun, která stojí za vznikem Javy, tuto nevýhodu ale do určité míry vyřešila, když vyvinula Just In Time Compiler (JIT). Java je také oblíbená pro dostupnost široké palety knihoven, které usnadňují vývoj aplikací.

2.5.2 Knihovna jsoup

Knihovna jsoup [23] je napsaná pro použití v jazyce Java. Tato knihovna má na práci zjednodušit práci se zpracováním HTML stránek. Označuje se také jako HTML parser, což znamená, že lze s tímto nástrojem HTML soubory snadno procházet přes jednotlivé tagy.

2.5.3 Knihovny Apache Commons

Apache Commons [24] v sobě zahrnuje soubor mnoha knihoven pro jazyk Java, které usnadňují mnoho různých zaměření, ať už jde třeba o práci se soubory, textovými řetězci, čísly a tak podobně. Obecně se Apache Commons zaměřují na ty nejčastěji používané funkce, které ale nejsou součástí samotného jazyka Java. Všechny knihovny Apache Commons podléhají Apache License, version 2.0 [25], což znamená, že je lze používat jako takzvaný *svobodný software* [26], [27]. V tomto projektu jsou použity následující knihovny:

Commons IO [28] – Tato knihovna pomáhá ve čtení vstupů a vypisování výstupů – IO (Input Output). Využití má v práci se soubory, kde lze načítat soubory různých formátů, archivované soubory zde lze rozbalovat a v neposlední řadě umí tato knihovna stahovat i soubory z internetu.

Commons Lang [29] – Lang v názvu této knihovny značí language (česky – jazyk). Zaměření je zde tedy na textové řetězce a práci s nimi. Knihovnu lze použít například k hledání podobnosti textových řetězců pomocí editační (Levenshteinovy) vzdálenosti [30]. Tato funkce je použita i v této práci.

2.5.4 MorphoDiTa

MorphoDiTa je nástroj vyvinut Ústavem formální a aplikované lingvistiky MFF UK. Nástroj umožňuje mnoho z dříve popsaných způsobů redukce dimenzionality v rámci analýzy textu. V rámci toho projektu se pracuje konkrétně s těmito procesy: tokenizace, lemmatizace a tvorba tagů. Ukázka práce

je dostupná k vyzkoušení na adrese: [31]. Pro použití ve vlastních projektech je připraveno API nebo samostatná instalace. Dále používá MorphoDiTa následující strukturu tagu: [32]. Tento přehledný popis tagu zároveň může sloužit k lepšímu porozumění tagu jako takového. Tento nástroj je distribuován pod licencí Mozilla Public License Version 2.0 [33], což mimo jiné znamená, že ho je možné použít v této praci. Licence nepovoluje komerční využití, což výsledek této práce splňuje.

2.5.5 Databáze SQLite

V rámci této práce se využívá SQLite [34] databáze, která je uložena lokálně. SQLite patří mezi SQL(Structured Query Language) databáze. SQL je jazyk, který se používá nad relačními databázemi. SQLite bylo zvoleno hlavně díky jednoduchosti a snadné obsluze. Uložení dat v databázi bylo vybráno, protože třetí projekt bude data zobrazovat opakovaně a uložení v databázi je vzhledem k rychlostem vhodné k tomuto účelu.

2.5.6 Framework Vaadin

Vaadin je framework pro tvorbu webových aplikací. Nejlépe vystihuje tento nástroj asi popis z úvodní webové stránky Vaadinu[35]: *„Vaadin is an open source web framework that helps Java developers build great user experiences with minimal effort.“* Mezi hlavní vlastnosti tedy patří otevřenost a jednoduchost. Pro psaní v tomto frameworku se používá jazyk Java. Kód psaný v Javě je ale následně překládán do JavaScriptu.

Jednoduchost však s sebou nese nepřímo i nevýhodu a tou je nižší přizpůsobitelnost než právě ve výše zmíněném JavaScriptu nebo v jiných nástrojích. Díky otevřenosti je možné si do Vaadinu přidávat další funkce, což vyhovuje velké komunitě, která je aktivní a nástroj dále rozvíjí skrze různé knihovny.

2.5.6.1 Chart.js

Jednou z knihoven, kterou pro Vaadin komunita připravila je i portovaná verze [36] knihovny Chart.js [37]. Portovaná z toho důvodu, že originální Chart.js je k dipozici pro jazyk JavaScript. Knihovna se zaměřuje na vytváření grafů a dalších typů vizualizace dat. Všechny vytvořené vizualizace jsou vyvedeny v moderním stylu a jsou snadno přizpůsobitelné. Portovaná verze ubírá některé funkcionality, ale stále se jedná o výborný nástroj, který je dostupný k použití zdarma oproti grafům přímo od tvůrců frameworku Vaadin – Vaadin Charts [38].

2.5.7 Kumo

Další knihovnou pro vizualizaci dat je Kumo [39]. Tento nástroj se soustředí na zobrazení slov ve formě tzv. slovních mraků (word cloud). Kumo není vázané na žádný konkrétní framework a lze tuto knihovnu využít v kterémkoliv Java projektu.

2.5.8 JUnit

JUnit [40] je knihovnou pro testování programů napsaných v jazyce Java. Jak plyne již z názvu (unit – jednotka, díl), tato knihovna se zaměřuje na testování menších částí zdrojového kódu [41].

Kapitola 3

Praktická část

3.1 Architektura práce

Celá práce je rozdělena do tří Java projektů viz Obrázek 3.1. První projekt má na starosti stažení dat z webu, druhý projekt data zpracovává a analyzuje a třetí zpracovaná data prezentuje.

3.2 Databáze

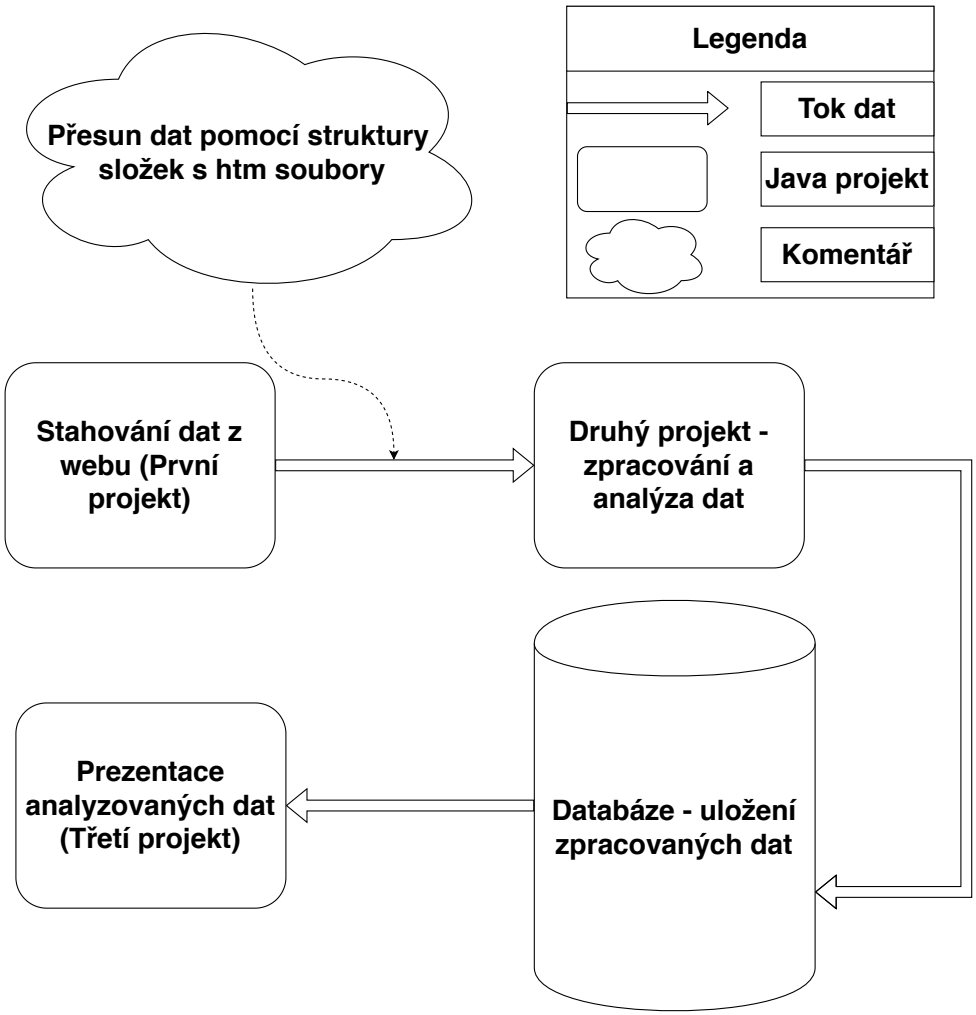
Databáze v tomto projektu je typu SQLite. Toto úložiště slouží k uchování analyzovaných dat z druhého programu. Data nahrává po analýze do databáze druhý projekt. Nahraná data poté využívá třetí program, který se opakovaně ptá databáze podle toho, co uživatel potřebuje zobrazit v prezentační části.

3.2.1 Entity

Pro větší přehlednost a ušetření slova „entita“ jsou entity psány s velkým písmenem na začátku slova.

Osoby uchovává data o lidech. Mezi nejdůležitější atributy této entity patří ID, jméno, příjmení a tituly před a za jménem. Dále jsou zde doplňkové atributy jako narození, úmrtí, poslední změna (myšleno v rámci systému poslanecké sněmovny) a pohlaví. Tato entita je důležitá hlavně pro následující entitu *Poslanec*.

Poslanec je „centrem“ celé databáze, protože se na ni váže mnoho jiných entit. Záznam v této entitě zastupuje poslanecký mandát v jednom volebním období. *Poslanec* má vazbu na *Osobu*, takže jedna *Osoba* může mít více *Poslanců* (poslaneckých mandátů v různých obdobích), ale *Poslanec* má právě jednu *Osobu*.



Obrázek 3.1: Architektura praktické části

Organy obsahuje informace o orgánech různého typu. Entita si uchovává informace o jméně v češtině a angličtině, zkratku a data začátku a konce platnosti. Z pohledu této práce jsou zajímavé hlavně orgány ve smyslu volebních období. Dále jsou tu zaznamenané třeba i politické strany, volební kraje, poslanecké výbory, skupiny a mnoho dalších. Tato entita má jednoho „rodiče“ a tím je entita *Typ_organu* – zde je k *Orgánům* dodatečná informace ve smyslu jejich typu. Například všechna volební období entity *Organy*, mají vazbu na záznam v tabulce *Typ_organu*, který označuje Poslaneckou sněmovnu. Dále má *Poslanec* tři vazby na tři různé záznamy této entity, kde jedna vazba znázorňuje příslušnost *Poslance* k volebnímu období, druhá k politické straně a třetí k volebnímu kraji. Tato trojnásobná vazba vychází z datového modelu a dat, která poskytuje Poslanecká sněmovna.

Bod označuje jeden jednací bod jedné schůze Poslanecké sněmovny. Bod má textové znění, číslo schůze a datum. Entita je navázaná na entitu *Organy*, aby měla spojení na příslušné volební období.

Projev popisuje jednu nepřerušenou promluvu poslance. *Projev* je popsán hlavně samotným textem projevu. Je zde ale zaznamenán i počet slov, počet negativních slov, počet pozitivních slov a sentiment celého projevu. *Projev* se váže k jednomu *Bodu* a jednomu *Poslanci* – řečníkovi.

Slovo označuje část projevu. Každý projev je složen ze slov. Tato slova jsou uchovaná v základním tvaru v této samostatné entitě, aby se případná analýza nemusela dělat opakovaně. Kromě samotného slova je entita popsána morfologickým tagem (popisuje slovní druh, číslo a podobné věci), počtem výskytů (jde o počet výskytů stejného slova ve stejném tvaru v rámci jednoho projevu) a nakonec je zde uložen i sentiment. Entita *Slovo* je navázána jen na *Projev*.

Zminka uchovává informace o tom, když je v některém projevu zmíněné jméno jiného poslance. Zmíněný poslanec je vždy ze stejného volebního období, v kterém byl projev pronesen. Kromě vazeb na *Poslance* a *Projev* zde je ještě atribut, který uchovává informaci o tom, kolik zmínek konkrétního poslance v daném projevu bylo.

Poslanec_statistiky je entita, které uchovává statistické informace pro jednotlivé poslance. Je zde tedy vazba na jednoho *Poslance*. Uloženy jsou tady ještě položky sentiment a počet slov, v obou případech jde o celkové hodnoty spočítané ze všech projevů daného poslance.

Poslanec_statistiky_mesic uchovává statistiky o poslanci, v tomto případě za určitý měsíc. Uložen je tu tedy konkrétní měsíc, sentiment a tři druhy počtu slov – celkový, pozitivní slova a negativní slova. Tato entita se váže na entitu *Poslanec_statistiky*.

Poslanec_statistiky_zminky je poslední entita. Uchovává zmínky mezi poslanci. Nejde tu ale o zmínky v jednotlivých projevech jako u entity *Zminka*. Tady jde o to, kolikrát jeden poslanec zmiňoval jiného za celé volební období. Jako atribut je zde tedy počet zmínek. Poté tu máme dvě vazby. Vazba na entitu *Poslanec_statistiky* označuje toho, kdo dané zmínky říkal – řečník. Vazba na *Poslance* značí toho, kdo byl zmiňován – zmíněný.

3.3 Stahování dat z webu (První projekt)

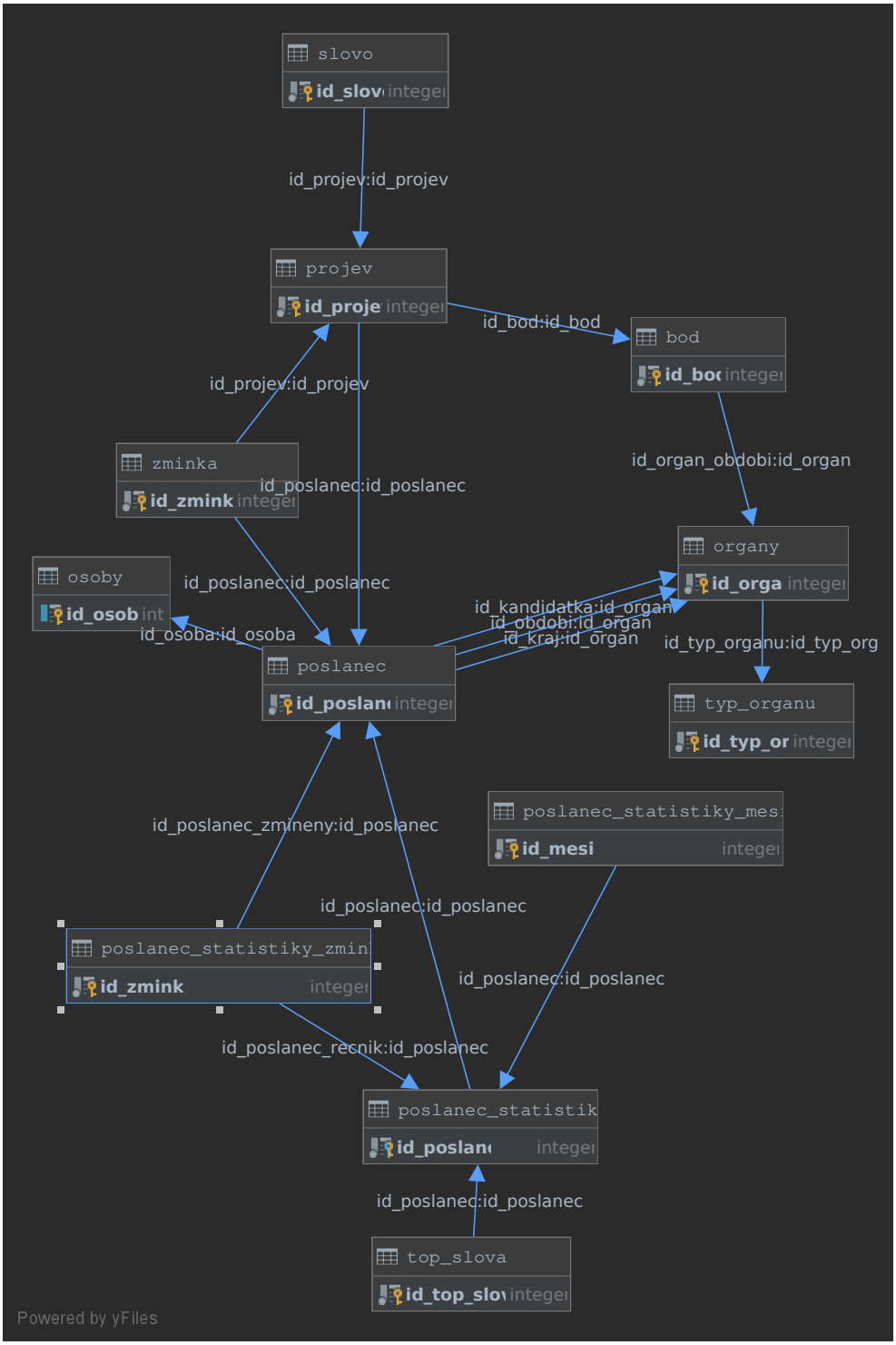
Novější volební období jsou poskytována na webu Poslanecké sněmovny v mnoha různých formách a formátech. Lze si stáhnout zvukové záznamy, těsnopisné zprávy ve formátu pdf a webové stránky (htm soubory) v klasické a zkomprimované verzi (komprimovaný archiv obsahuje htm soubory). Htm soubory jsou soubory stejného typu jako html, rozdíl je jen v příponě [42]. Zkomprimovaná verze má z pohledu této práce výhodu v tom, že se tyto soubory zaměřují čistě na projevy, takže z původní webové stránky zmizí různá menu, obrázky, hlavičky a podobně. Soubor je tedy přehlednější a lze ho lépe později zpracovat. Nevýhodou ale je, že u starších volebních období možnost komprimovaných souborů není. Aktuálně máme 8. volební období Poslanecké sněmovny Parlamentu ČR. Z těchto 8 období nabízí možnost komprimovaných souborů pouze 5 nejnovějších období. Pro stažení všech období je tedy potřeba podporovat komprimovanou i nekomprimovanou formu. Pro obě formy platí, že jedna schůze je typicky stránkovaná do více souborů a mimo to je ke schůzi ještě soubor (*index.htm*), který obsahuje soupis projednávaných bodů. Kromě souborů s projevy se stahují ještě seznamy s doplňkovými informacemi o osobách, poslancích, sněmovních orgánech a dalších.

Samotný program pracuje tak, že jako vstup obdrží cílovou cestu, kam si uživatel přeje data stáhnout, a seznam období, která chce stáhnout. Období je zde ve formátu *„rok“ps*, kde se za rok dosadí rok začátku příslušného volebního období (zatím nebyly sněmovní volby dvakrát v jednom roce).

3.3.1 Komprimovaná verze

Program se vždy snaží nejdříve získat komprimovanou verzi a to hlavně ze dvou důvodů. Zaprvé je stažení jednoho souboru rychlejší než stahování mnoha souborů. Zadruhé v komprimované verzi jsou soubory očištěny o data, která nás nezajímají, jak je popsáno již výše.

Stahování jednoho volebního období probíhá po jednotlivých schůzích, kde jedna schůze je jeden komprimovaný archiv ve formátu zip. Jelikož není dopředu známé, kolik má určité volební období schůzí (období nemá pevně daný počet schůzí), tak program zkouší stahovat od schůze s číslem 1 a postupně číslo zvyšuje. Když se dojde k webové adrese, která neexistuje, tak je



Obrázek 3.2: Schéma databáze

to bráno tak, že další schůze už neexistuje (příklad formátu odkazu na archiv *www.psp.cz/eknih/2006ps/stenprot/zip/001schuz.zip – 2006ps* označuje volební období a *001schuz* vyjadřuje 1. schůzi). Program by bylo možné vylepšit tak, že by první z webu Poslanecké sněmovny získal seznam odkazů na poslanecké schůze a podle toho poté schůze stahoval. Tento postup by řešil problém s nejistotou, jestli jsou staženy všechny schůze.

Archiv se stáhne z webu Poslanecké sněmovny pomocí knihovny Apache Commons IO viz 2.5.3. Přímo pro stažení se použije funkce *copyURLToFile(final URL source, final File destination) z balíčku FileUtils*, kde jako source bude odkaz na stažení požadovaného souboru (zip archivu) a destination je cesta na dočasný soubor – temp/temp.zip. Z dočasného souboru je pak archiv rozbalen do příslušné složky, která odpovídá formátem složkám z obrázku 3.3 (jde o složky se jmény 001schuz, 002schuz, 003schuz, ...).

3.3.2 Nekomprimovaná verze

Když se nepodaří stáhnout některou schůzi v komprimované verzi, tak se přistoupí k nekomprimované verzi. Toto rozhodnutí je provedeno na úrovni schůze a ne na úrovni celého období a to z toho důvodu, že i u období, které poskytuje komprimované schůze, mohou být schůze, kde nastane například problém s rozbalování zip archivu.

V tomto případě je potřeba stahovat soubory schůzí jednotlivě. Jde o jednotlivé části schůze (stránky) a *index.htm* soubor (příklad formátu odkazu na jednu část schůze *www.psp.cz/eknih/2017ps/stenprot/023schuz/s023015.htm – 2017ps* je označení pro období, *023schuz* označuje 23. schůzi a *s023015* označuje 15. část).

Program tedy stahuje jednotlivé části schůze postupně. Znovu chybí informace o tom, na kolik částí je členěna která schůze. Opět se tedy postupně zvyšuje číslo části schůze, dokud daná část existuje. Nevýhodou je, že kdyby nějaká část chyběla nebo byl odkaz nějak nespolehlivý, tak potenciálně program nějaká data nestáhne. Program tedy předpokládá, a v aktuálních datech tomu tak je, že části jednotlivých schůzí jsou číselně správně za sebou.

3.3.3 Výstup programu

Pro přenos dat mezi prvním a druhým projektem se používá složka s různými souborovými formáty. Tato složka obsahuje složky s jednotlivými volebními obdobími a poté seznamy osob, poslanců a dalších entit. Tyto seznamy jsou z webu poslanecké sněmovny [7]. Příklad struktury ukazuje Obrázek 3.3, v realitě jsou ovšem složky mnohem obsáhlejší.

V příkladu je možné vidět na nulté úrovni soubory s koncovkou .unl, které uchovávají výše zmíněné seznamy. Soubory s koncovkou .unl jsou používány často jako podpůrný formát pro mapy GPS navigací značky Garmin [43]. Pro použití v této práci je vhodnější ale popis z webových stránek Policie ČR [44].

Unl je zde chápán jako textový formát, kde každý řádek představuje jeden záznam předem definovaného druhu (osoba, věc,...) a jednotlivé atributy jsou odděleny znakem „|“.

Dále jsou zde složky, které se soustředí na jednotlivá období (příklad 2013ps, 2017ps – číslo značí rok začátku volebního období). Ve složce označující období jsou dále uloženy složky s jednotlivými schůzemi (příklad 001schuz, 002schuz a podobně – číslo zde v názvu složky označuje číslo schůze). Ve složce se schůzí pak najdeme vždy *index.htm*, což je soubor, v kterém jsou znění projednávaných bodů. Mimo *index.htm* zde jsou ještě další .htm soubory, které uchovávají informace o samotných projevech, projevy jsou stránkovány do jednotlivých souborů, aby případný jeden soubor nebyl moc velký (příklad *s001002.htm*, kde první trojčíslí ve jméně označuje číslo schůze a druhé trojčíslí označuje stránku – část).

3.4 Zpracování a analýza dat (Druhý projekt)

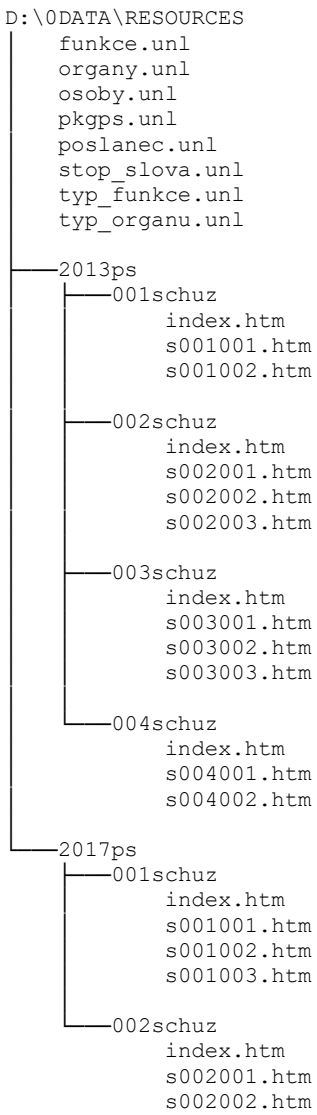
Program, který vznikl z tohoto projektu, má na starosti data stažená předchozím programem analyzovat a nahrát do databáze. Z této databáze jsou následně informace využívány na prezentaci.

3.4.1 Popis běhu programu

Program obdrží jako vstup označení požadovaného období, které se má zpracovat, a cestu ke složce s daty. Nejdříve program smaže z databáze všechny informace o nahrávaném období. V databázi už může dané období existovat a může být například v neaktuální verzi. Je potřeba ho tedy smazat. Nabízelo by se řešení informace v databázi jen rozšířit. Zde by však mohlo dojít k problémům, protože i když se většinou starší data neupravují, tak na to nelze spoléhat a ke změnám dojít může. U novějších dat dochází po vydání ještě k jazykovým korekcím, které by také mohly způsobit nekonzistenci ve zpracovávaných datech.

V dalším kroku dojde k načtení do databáze všech dat, která nepotřebují důslednější zpracování. Data jsou čerpána ze složky s daty 3.3 a jedná se o unl soubory. Unl soubory jsou zde použity jako výpis sněmovní databáze, alespoň to tak vypadá podle členění souborů. Jde o data pro entity: *Typ_organu*, *Organy*, *Osoby* a *Poslanec*. Následně dojde k odebrání všech záznamů z tabulky *Osoby*, které nemají vazbu na žádnou entitu typu *Poslanec*. Seznam osob totiž obsahuje velké množství lidí, kteří nikdy neměli poslanecký mandát. Pro lepší představu data za všechna volební období: osob je celkem 6592, poslaneckých mandátů je celkem 1730 a osob s alespoň jedním mandátem je 1027. Touto operací se tedy zbavíme velkého počtu osob, s kterými se v práci vůbec nepracuje a jejich záznamy jsou pro tuto práci tedy zbytečné.

Nakonec proběhne zpracování samotných projevů a přidružených dat. Nejdříve se načtou projednávané body, které se uloží do databáze. Poté se zpracují



Obrázek 3.3: Příklad struktury dat mezi 1. a 2. projektem

a do databáze načtou projevy, kterým program přiřadí vazby na příslušné body a poslance. V rámci načítání projevů se vytvoří a do databáze uloží i příslušné entity slov a zmínek na jiné poslance. Posléze se prochází postupně poslanec po poslanci a ke každému se vytváří a ukládají další údaje. Konkrétně jde o informace, které jsou uloženy v entitách *Poslanec_statistiky*, *Top_slova*, *Poslanec_statistiky_mesice*, *Poslanec_statistiky_zminky*, viz Obrázek 3.2.

3.4.2 Podoba vstupních dat

Jak bylo už nastíněno v podkapitole 3.3, podoba dat z Poslanecké sněmovny je v různých volebních obdobích dost odlišná. U projevů a jednacích bodů se sice vždy jedná o htm soubory, ale jejich struktura už bývá značně rozdílná. Za předpokladu, že tato data nikdo v poslední době nijak neupravoval, je to ale logické a pochopitelné. V nejstarších obdobích jsou používané zastaralé html tagy (příklad: starší verze – *<CENTER>*, novější verze – *<p align=center>*). Rozdílné také bývá umístění dat ve struktuře souboru, nelze se na ni tedy spoléhat a je nutno postupovat tag po tagu. Změna proběhla i u číslování projednávaných bodů. V prvních obdobích byly používány římské číslice, v novějších jsou ale číslice arabské. Pak je zde více částí souboru, které se tváří jako validní data a přitom validní nejsou a jejich filtrování není vždy úplně snadné.

Tyto problémy jednotlivě nevypadají nijak neřešitelně, ale je jich mnoho a navzájem se kombinují, takže je docela složité zachytit všechny možné scénáře. Další věcí je, že by program měl umět reagovat i na nějaké drobnější budoucí úpravy. Nakonec tedy není kladen velký důraz na formátování textových vstupů právě z důvodu častých změn. Jako příklad lze uvést třeba právě číslování projednávaných bodů. V některých případech se používá k označení jednacího bodu text s pořadovým číslem a v jiných případech zase ne. Pro sjednocení znění bodů byla na začátku vypracování práce snaha o mazání číslování. Některá období ale používají k číslování číslice arabské a jiná zase číslice římské. Problém se tedy tedy nakonec řeší tak, že když je potřeba znění jednacího bodu porovnávat, tak se neporovnává přesná shoda, ale využívá se co nejbližší podobnosti. Podrobněji popsáno v sekci Zpracování bodů 3.4.5. Program si tedy umí poradit s většinou problémů, ale některé těžko předvídatelné kombinace mohou skončit například načtením textu, který není projev, ale tento a podobné scénáře se stávají v minimu případů.

3.4.3 Mazání existujících dat

Odebírání již existujících dat z databáze probíhá tak, že se jako první vyhledá entita příslušného období. Poté se postupně z databáze mažou všechny entity typu *Poslanec* a *Bod*, které jsou navázané na vybrané období. Ostatní entity určené k mazání jsou mazány kaskádovitě přes vazby viz Obrázek 3.2.

3.4.4 Zpracování informací o osobách a orgánech

Informace o osobách a sněmovních orgánech jsou popsány v unl souborech viz Obrázek 3.3. Struktura takového unl souboru je v tomto případě vždy stejná. Jeden řádek obsahuje data, která se vážou k jednomu záznamu. Například soubor *osoby.unl* má na každém řádku informace o jedné osobě. Informace jsou dále odděleny znakem „|“ na jednotlivé atributy. Konkrétně jsou využity soubory: *organy.unl* (entita *Organy*), *osoby.unl* (entita *Osoby*), *poslanec.unl* (entita *Poslanec*) a *typ_organu.unl* (entita *Typ_organu*).

Čtení dat z unl souborů zajišťuje třída *UNLFileReader*, která dostane při vytváření instance cestu k souboru a poté po „zavolání“ instanční metody *getLineList* vrací seznam textových řetězců, které reprezentují původně jeden řádek ve výchozím souboru. Seznam je poté zpracován z textových řetězců do podoby konkrétních datových typů. Všechny zmiňované entity mají atributy následujících typů:

Text – U atributů vyjádřených textem dojde k odstranění přebytečných mezer. Jedná se o atributy jako jméno, příjmení, adresa, e-mail a tak podobně.

Číslo – U číselných atributů je nejdříve zkontrolováno, jestli je převod (parsování) z textu na číslo možný. Poté se provede převod přes funkci integrovanou přímo v základní výbavě jazyka Java. Číselný styl atributu se používá hlavně u identifikačních čísel jednotlivých entit a u zaznamenávání množství (slov, zmínek a podobně). V případě neplatnosti čísla se použije hodnota *null*.

Datum – Zpracování data je asi nejsložitější. Za prvé je potřeba vzor (pattern), podle kterého je datum zapsáno (příklad: dd.MM.yyyy). Poté dojde k vytvoření data jako instance třídy *java.util.Date* a nakonec se datum převede do instance třídy *java.sql.Date*. Datum se používá jako atribut u data narození, úmrtí a podobných údajů. V případě neplatného nebo prázdného data se použije hodnota *null*.

3.4.5 Zpracování bodů

Každý projev má podle schématu databáze (viz Obrázek 3.2) vazbu na jeden jednací bod. Schůze Poslanecké sněmovny ale probíhá nějakým úvodem, který nelze přiřadit k žádnému existujícímu bodu. Je to takový úvod, kde se například o programu a tedy i bodech schůze teprve hlasuje. Pro tento a podobné účely byl vytvořen navíc ke každé schůzi jeden bod, který tyto projevy spojuje a je pojmenován „—Provozní úkony—“.

Pro zpracování jednacích bodů program postupně iteruje po jednotlivých schůzích a ve struktuře složky se zdrojovými daty se zaměřuje na soubory *index.htm* viz Obrázek 3.3. Každý soubor obsahuje soupis jednacích bodů pro jednu schůzi – Obrázek 3.4.

```
<a name="b2"></a><b>II. Návrh organizačního výboru Poslanecké
↪  sněmovny na zkrácení zákonné lhůty 60 dnů k˜projednání
↪  návrhů zákonů podle sněmovních tisků 315 -
↪  321</b></a><br><br>
<a href="/eknih/1993ps/stenprot/009schuz/9-1.html#24">Projedn⌋
↪  ávání</a>, část č. 1 (18. května
↪  1993)<br>
<a href="/eknih/1993ps/stenprot/009schuz/9-3.html#426">Projed⌋
↪  návání</a>, část č. 2 (20. května
↪  1993)<br>
```

```
<a name="b22" id="b22"></a><b>22. Návrh na volbu předsedů
↪  stálých komisí Poslanecké sněmovny</b></a><br><br>
<a href="4-2.htm#q121">Projednávání</a>, část č. 1 (15.
↪  prosince 2017)<br>
<a href="4-2.htm#q190">Projednávání</a>, část č. 2<br>
<a href="4-3.htm#q230">Projednávání</a>, část č. 3 (19.
↪  prosince 2017)<br>
<a href="4-3.htm#q265">Projednávání</a>, část č. 4<br>
<a href="4-3.htm#q379">Projednávání</a>, část č. 5<br>
```

Obrázek 3.4: Příklady HTML tagů s definicemi jednacích bodů

Program prochází celý htm soubor přes jednotlivé tagy. Když narazí na tag, který identifikuje jako tag s jednacím bodem, tak se jím dále zabývá. Entita *Bod* potřebuje pro své vytvoření textové znění a datum. Text bodu se z tagu vyseparuje docela jednoduše a ani se nijak zásadně neupravuje. Případné úpravy by jen zbytečně zkomplikovaly načítání bodů v různých obdobích (příklad: různé číslování 3.4.2).

S datem je situace složitější. Různé jednací body se mohou projednávat v různých dnech. Dokonce i jeden bod se může projednávat ve více dnech (projednáváních). Datum je typicky zapsáno přímo u bodu (příklad formátu: „8. října 2002“). V tomto případě se bere jako datum bodu datum prvního projednávání daného bodu. V případě, že se načítání data z nějakého důvodu nepodařilo nebo se jedná o bod „—Provozní úkony—“, tak se volí jako jako datum bodu datum, které je uvedeno jako datum konané schůze. To může vypadat jako datum bodu, ale může být i daleko komplikovanější – například „27., 28 února, 1., 2., 6., 7., 20., 21. března 2018“. Program vybere vždy ale úplně první datum. V předchozím příkladu by to tedy bylo – „27. února 2018“. První datum se vybírá z toho důvodu, že hlavním bodem s tímto datem bude „—Provozní úkony—“ a projevy k tomuto bodu lze očekávat nejvíce právě na

začátku celé schůze.

Pro každý bod je nakonec vytvořena entita, která je uložena do databáze.

3.4.6 Zpracování projevů

Pro získání dat, která slouží jako základ k načtení projevů, je potřeba znovu procházet postupně složky s obdobími v příslušném volebním období. Tentokrát se program zaměřuje na soubory tohoto typu: *s001003.htm*, viz Obrázek 3.3. Projevy jedné schůze nejsou v jednom souboru, ale jsou stránkovány, proto je potřeba pro uchování projevů jedné schůze typicky více souborů.

Projev má vždy řečníka, text a bod, ke kterému se váže. Pak jsou zde další údaje, ale ty nejsou tak zásadní. Program prochází schůzi po jednotlivých htm souborech a hledá potřebné informace. Bez reakce prochází různé nepotřebné části souboru jako jsou hlavičky, menu nebo další části webových stránek. Zaměří se tedy jen na řečníky, body a pronášený text. Tyto tři části se v souborech různě střídají. Když narazí program na jednací bod, tak ví, že všechny následující projevy až po další bod se vážou k právě přečtenému bodu. Podobné je to s řečníkem. Pronášená řeč se může skládat z více částí. Program části shlukuje a přiřazuje je k aktuálním řečníkům.

3.4.6.1 Jednací bod

Zpracování bodu je docela přímočará záležitost. I zde jsou ovšem různé styly zapsání jednacího bodu. Program rozpozná html tag, ve kterém je uložen bod – příklady v Obrázku 3.5. Následně přečte jeho text a porovná ho se všemi body, které daná schůze má (tyto body získá z databáze). Z porovnávaných bodů program vybere ten, který má nejmenší editační vzdálenost vzhledem k přečtenému textu. Zároveň ale tato editační vzdálenost musí být dostatečně malá – program má nastavenou maximální přípustnou editační vzdálenost rovnou 10. Tento postup zajistí správné chování i v případech, kdy ten, kdo data zapisoval, udělal třeba překlep. Je zde ale i možnost, že je přečtený text moc odlišný od všech bodů v databázi pro tuto schůzi. Poté se využije bodu „—Provozní úkony—“. Program tedy žádný projev nezahodí z důvodu, že by k němu neexistoval bod.

3.4.6.2 Řečník

Získání správného řečníka je už značně komplikovanější – některé příklady jsou na Obrázku 3.6. V nejlepším případě je u řečníka odkaz na webovou stránku a v tomto odkazu je i ID osoby, která mluví. Pak už stačí jen z databáze zjistit, jestli daná osoba má v aktuálním volebním období poslanecký mandát. Jestliže poslanecký mandát nemá, tak je projev z pohledu této práce ignorován, protože nejde o poslance a práce se věnuje pouze poslaneckým mandátům.

```
<p align=center><b>28. <br>
Rozhodnutí vlády o přeletech a průjezdech ozbrojených
↪  sil <br>
jiných států přes území České republiky v roce 2018 <br>
/sněmovní tisk 14/ - prvé čtení </b></p>
```

```
<P ALIGN="CENTER">4.<BR>
Návrh poslanců Zdeňka Jičínského a dalších na vydání
↪  ústavního zákona<BR>
o˜lidovém hlasování (referendu) a o˜lidové zákonodárné
↪  iniciativě<BR>
/sněmovní tisk <a href="/sqw/historie.sqw?T=149&O=2">149</a>/
↪  - druhé čtení </P>
```

```
<p ALIGN="CENTER">11.</p>

<p ALIGN="CENTER">Návrh poslankyně Zuzky Rujbrové na vydání
↪  zákona, kterým se
mění</p>

<p ALIGN="CENTER">a doplňuje zákon České národní rady č.
↪  200/1990 Sb.,
o˜přestupcích,</p>

<p ALIGN="CENTER">ve znění pozdějších předpisů (sněmovní tisk
↪  <a href="/sqw/historie.sqw?T=126&O=2">126</a>) - prvé
čtení</p>
</b>
```

Obrázek 3.5: Příklady HTML tagů s jednacími body

```
<a href="/sqw/detail.sqw?id=6074">Místopředseda PSP Petr
↪  Fiala</a>
```

```
<a href="https://www.vlada.cz/cz/clenove-vlady/jaroslava-nemc
↪  ova-162045/">Ministryně práce a sociálních věcí ČR
↪  Jaroslava Němcová</a>
```

```
<A HREF="/sqw/p.sqw?P=535" id="r4">Poslanec Petr Nečas:</a>
```

Obrázek 3.6: Příklady HTML tagů s řečníky

Složitější je situace u řečníků, kteří takový odkaz nemají. K tomuto scénáři dochází ve starších obdobích. Stát se to také může, když je řečník členem Vlády ČR. Pak je u jeho jména odkaz na webové stránky v rámci vládního systému. U těchto lidí si program nejdříve připraví seznam všech poslanců. Poté prochází postupně všechny poslance a kontroluje, jestli jméno některého poslance není obsaženo v oslovení řečníka. V případě shody přiřadí poslance k projevu, v opačném případě je projev opět ignorován.

Osoby bez poslaneckého mandátu jsou většinou ministři nebo hosté (delegace ze zahraničí a podobně). Podle dat z posledního volebního období tyto osoby pronesly celkem 3,48% všech projevů. Poslanci pronesli zbytek všech projevů, tedy 96,52%. Poslanecké projevy tedy značně převládají. Zároveň se tato práce zaměřuje přímo na poslance a ne na všechny projevy pronesené v Poslanecké sněmovně. Projevy ostatních osob by byly možná dobré pro dokreslení situace a kontext, ale ztratila by se přehlednost a jednoznačné zakotvení domény. Proto se ignorují.

3.4.6.3 Projev

Projev je v souborech s daty nejčastěji zastoupený segment. Stává se, že jsou i soubory, kde nejsou žádné body ani řečníci, ale stále probíhá promluva jednoho řečníka. Získat projev v „surovém“ stavu není na první pohled nijak složité. Jsou zde ovšem znovu určité odlišnosti v různých poslaneckých schůzích – některé příklady jsou znázorněny na Obrázku 3.7. Program postupně spojuje text z tagů, které obsahují texty s projevy. V tomto procházení a skládání se přidává i jednací bod, když se mezi projevy nějaký vyskytne. To je z důvodu, že nový jednací bod se jen tak z ničeho nic neobjeví na informační tabuli, ale někdo jej musí uvést a říct, takže i znění bodu patří do projevu.

Když má program celý jeden projev v „surovém“ stavu, tak je potřeba ho upravit. Neopracovaný projev může obsahovat i něco, co připomíná scénické poznámky z divadelního světa. Jedná se vlastně o část textu, kterou nikdo

```
<p align="justify">Dále budeme hlasovat o návrhu pana
↪  poslance Luzara, který navrhuje zařazení nového bodu, a
↪  tím je Pozice vlády k nucenému prodeji Mittal Steel
↪  Ostrava. </p>
<p align="justify">Já zahajuji hlasování a ptám se, kdo je
↪  pro zařazení tohoto nového bodu. Kdo je proti? </p>
<p align="justify">Hlasování číslo 30, přihlášeno je 189,
↪  pro 98, proti 3. Tento návrh byl přijat. </p>
```

```
<P ALIGN="JUSTIFY"> V˜tuto chvíli oznamuji, že pan poslanec
↪  Maixner má náhradní kartu č. 6 a pan poslanec Gongol
↪  náhradní kartu č. 2.</P>
<P ALIGN="JUSTIFY"> Pro dnešní jednání jsou omluveni poslanci
↪  Jaroslav Maňásek a Michael Kuneš z˜důvodu nemoci. Z˜vlády
↪  potom pánové ministři Jan Ruml, Jiří Gruša a Karel
↪  Kühnl.</P>
```

```
<p>Ptám se, kdo tento návrh podporuje? Kdo Je proti?</p>

<p>Tento návrh nebyl přijat - pro bylo 92, proti 45 poslanců.
↪  Protože nehlasovalo 47,
nebyl tento návrh přijat.</p>

<p>Konstatuji, že jsme tím projednali bod č. 9. Přistoupíme
↪  k˜projednávání bodu
č. 10, kterým je</p>
```

Obrázek 3.7: Příklady HTML tagů s projevy

neřekl, ale zapisovatel se tím snažil zachytit, co se dělo v sále. Může to být třeba reakce na výzvu, aby poslanci povstali – *„Já vás prosím, abyste povstali. (Děje se.)“*. Někdy je situace jiná a v závorce je zapsána reakce někoho jiného, kdo ale mluví ze svého místa a reakce v závorce je důležitá pro kontext projevu řečníka u pultíku – *„Pane místopředsedo, velmi děkuji za slovo. Vážené kolegyně, vážení kolegové, vážení přítomní členové vlády, já se pokusím být zajímavější, než telefon paní ministryně Schillerové, ale to bych musel udělat salto dozadu. (Ministryně Schillerová z vládní lavice: Já pracuji.) Já také pracuji, ale měla byste mě u toho poslouchat.“*. V tomto případě by se možná nabízelo vytvořit z textu v závorkách ještě druhý projev. Bohužel ale „projev v závorkách“ nemá žádný pevně daný styl a podoba závorky může být rozmanitá, takže i s přihlédnutím k tomu, že se takový projev mimo pultík

neděje často a není moc dlouhý, se žádný takový projev nevytváří. Část mezi závorkami se tedy z projevu vždy odstraní, protože je tam něco, co řečník neříkal. Dále v projevu mohou být takové drobnosti jako zbytečně se opakující mezery nebo například přebytečná dvojtečka na začátku projevu. Tyto neduhy program také opravuje.

V rámci načítání projevu se rovnou pracuje i se samotnými slovy, z kterých se projev skládá. První program analyzuje slova pro entitu *Slova* a poté i pro entitu Zminka.

Vytvoření entit *Slovo* probíhá tak, že program vezme projev a pomocí nástroje Morphodita z něj dostane seznam lemmat s tagy. Vnitřně se nejdříve provede tokenizace, takže nástroj má seznam tokenů a z nich poté získá seznam lemmat s tagy. Nástroj označuje jako samostatné lemma i různá interpunkční znaménka a podobné, pro tuto práci nepotřebné, věci. Tato nepotřebná lemmata vyfiltrujeme díky informaci v tagu, protože tag nám říká i informaci o tom, jestli se jedné o nějaké interpunkční znaménko nebo něco podobného. Už při načítání dvojic lemmat a tagů provádí program určitou optimalizace pro budoucí snadnější zacházení. Jestliže se v jednom projevu objevuje jedno slovo v úplně stejném tvaru (je tedy stejné lemma i tag), tak se nevytváří nový záznam, ale jen se přičte 1 k počtu výskytů u už existujícího záznamu. To podporuje i entita Slovo, která má počet výskytů jako atribut. Projev je tedy převeden přes lemmata s tagy a počtem výskytů na slova. Ke slovům se přiřazuje ještě sentiment 2.4.1. Pro každé slovo se kontroluje, jestli se vyskytuje ve slovníku pozitivních a negativních slov. Pokud se vyskytuje, tak se slovu přiřadí sentiment – značení: 1 pro pozitivní a -1 pro negativní. Jestliže slovo ve slovníku není, tak nemá žádný sentiment –značení: 0 pro slova bez sentimentu. Nakonec jsou entity Slov uloženy do Databáze. Projev je po zpracování jednotlivých slov ještě obohacen o počty pozitivních a negativních slov, celkový počet slov a sentiment celého projevu, který se počítá jako aritmetický průměr ze sentimentů všech slov projevu. Příklady Slov zobrazuje Obrázek 3.8. Na tomto obrázku lze vidět i to, že některá slova mají za sebou i dodatečné informace. Může zde být například krátké popsání různých významů slov. V databázi jsou uchována slova i s dodatečnými informacemi pro případné budoucí využití. Při vizualizaci slov v rámci tohoto projektu, například ve slovních mracích, se tyto dodatečné informace odstraňují.

Zmínky politiků navzájem jsou analyzovány v momentu, kdy má program k dispozici seznam entit Slovo z jednoho projevu. Program postupně prochází seznam všech slov jednoho projevu. Slovo, které by mohlo být zmínkou jiného politika, musí začínat velkým písmenem, tím se daná množina slov zásadně redukuje. Poté program prochází seznam poslanců volebního období, v kterém byl projev respektive slovo proneseno a hledá přesnou shodu mezi příjmením poslance a slovem. Je možné, že nenajde žádnou shodu, slovem tedy může být jméno člověka, který není poslanec, nebo třeba jméno obce. V případě nalezení shody může dojít k nalezení ne jen jednoho poslance. Poslanci mohou mít stejná příjmení (popsáno v části 2.2.1) a nelze pak tedy rozhodnout, koho se

	id slovo	id projev	slovo	tag	pocet vyskytu	sentiment
142	142	2	prvý	AAIS3----1A----	1	0
143	143	2	se_^(zvr._zájmeno/č...	P7-X4----------	1	0
144	144	2	pan_^(oslovení)	NNMS1-----A----	1	0
145	145	2	sněmovna	NNFS2-----A----	1	0
146	146	3	občanský	AAIS1----1A----	1	0
147	147	3	boj	NNIS7-----A----	1	-1
148	148	3	přetrvávat_:T_^(*4at)	VB-S---3P-AA---	1	0
149	149	3	spolehlivý	AAIS7----1A----	1	1
150	150	3	český	AAFS1----1A----	1	0
151	151	3	prvek	NNIS4-----A----	1	0
152	152	3	co-1	PQ--1----------	3	0
153	153	3	takřka	Db----------	1	0
154	154	3	Říman_;E	NNMP2-----A----	1	0
155	155	3	zdroj	NNIP2-----A----	1	0
156	156	3	člověk	NNMP1-----A---1	1	0
157	157	3	polovina	NNFS1-----A----	1	0
158	158	3	živel	NNIS4-----A----	1	0
159	159	3	zájem	NNIS2-----A----	1	1
160	160	3	zralost_^(*3ý)	NNFS4-----A----	1	1

Obrázek 3.8: Příklad zástupců entity Slovo

zmínka týká. Je tedy přiřazena ke všem, které program nalezl. Vytvořené zmínky jsou poté nahrány do databáze. Shodná příjmení u více poslanců nejsou jediný problém, který se v této doméně vyskytuje. Existují lidé, kteří mají víceslovná příjmení. Potom nelze obecně říct, jestli mají více příjmení, ale používají jen jedno a v tom případě které. Různí poslanci je taky mohou oslovovat libovolnou podmnožinou slov, která tvoří jejich příjmení. Řešením by v tomto případě bylo procházení slov v pořadí, v kterém byla pronesena. Program by se poté nedíval na jedno slovo izolovaně, ale zaměřil by se na takové „okénko“, kde by bylo několik slov (počet by měl být roven počtu slov v nejvíceslovném příjmení) a program by s touto informací dokázal určit, jestli byl poslanec zmíněn. Tento postup by eliminoval i to, aby se zmínění poslance s vícejmenným příjmením nepočítalo víckrát za jednu reálnou zmínku. Toto řešení by bylo funkční, ale v práci není implementováno, protože přednost dostaly jiné funkce. K tomuto kroku bylo přistoupeno i z toho důvodu, že vícejmenná příjmení se vyskytují v historii Poslanecké sněmovny Parlamentu ČR jen u 12 poslanců z 1027, tedy u 1,17% z celkového počtu poslanců.

3.4.7 Zpracování statistik

Program je v tuto chvíli v momentu, kdy jsou načtená v databázi všechna data ze souborů. Došlo už i k nějakému zpracování jako jsou slova nebo zmínky. Program se ale ještě zaměří na zpracování dat vzhledem k jednotlivým po-

slancům. Program má seznam poslanců ve zpracovávaném volebním období. Tento seznam prochází a postupně vytváří statistiky, konkrétně jde o statistiky vyjádřené následujícími entitami z databáze: *Poslanec_statistiky*, *Top_slova*, *Poslanec_statistiky_zminky*, *Poslanec_statistiky_mesice* – viz Obrázek 3.2.

3.4.7.1 Statistiky poslance

Entita *Poslanec_statistiky* uchovává souhrnné statistiky o poslanci za volební období. Konkrétně je to sentiment a počet slov. Sentiment se počítá jako aritmetický průměr ze všech slov všech projevů. Zde je ale optimalizace v podobě toho, že entita Projev si pamatuje počet pozitivních i negativních slov, tudíž není nutné chodit až na úroveň entity Slovo. Počet slov se získá podobně. Každá entita Projev má uložený i počet slov projevu, takže zde stačí sečíst tyto hodnoty u všech entit Projev.

3.4.7.2 Top slova

K poslanci se váže mnoho projevů a slov. Toto množství nedokáže člověk jednoduše zpracovat. V ideálním případě by tedy existovalo zjednodušení, které by poslance popsalo v několika slovech, z kterých by byly patrné jeho specializace nebo názory. Tento účel by skvěle plnila klíčová slova. Vytvoření klíčových slov ke každému poslanci program neumí, ale nabízí podobnou funkci. Ve zkratce program umí vybrat nejpoužívanější slova každého poslance a tento seznam dále očistí o stopslova. Vznikne tedy seznam slov, která rozhodně nelze označit jako klíčová slova, ale do určité míry plní stejný účel. Jde tedy o určitý druh redukce dimenzionality.

Konkrétní postup, jak program dosáhne výše popsaného seznamu je následující. Na začátku dostane program jednu entitu *Poslanec*, u které postupně projde všechny entity *Slovo*. U všech slov si pamatuje počet výskytů, kde nezáleží na různém tagu. Když má program zpracovaná všechna slova, seřadí slova podle počtu výskytů a vybere prvních 50 slov. Počet 50 slov byl zvolen z důvodu, že se v tomto počtu ještě lze vyznat a zároveň těch slov je dostatečný počet, aby to o poslanci řeklo některé informace. Aby slovo mohlo být v tomto výběru, tak se kromě pořadí v seřazeném seznamu hledí ještě na to, jestli se slovo nevyskytuje v seznamu stopslov.

Seznam stopslov byl vytvořen přímo pro tento projekt a obsahuje přes 400 slov. Základ tvoří slova slovních druhů jako jsou spojky, předložky a podobně. Dále byla přidána slova, která jsou často používaná, ale nemají žádný vlastní význam, například: argument, cena, číslo, čtení, doporučit a tak podobně. Tato slova by v jiných doménách nemusela být brána za stopslova, ale v této doméně tomu tak je. Zajímavým příkladem jsou slova z projevů poslanců, kteří zrovna předsedají schůzi. Zde se opakují pořád dokola stejná slova, ale významově většinou neříkají mnoho.

3.4.7.3 Zmínky poslanců

Informace o tom, jak se poslanci nazvájem zmiňují v jednotlivých projevech je zajímavá, ale pro získání většího kontextu je vhodné mít pohled z většího odstupu. Entita *Poslanec_statistiky_zminky* zajišťuje právě tuto funkcionalitu. Pro získání entit tohoto typu je potřeba přes etity Projev daného poslance projít všechny entity Zminka a postupně je nasčítat pro každého poslance zvlášť, vytvořit z nich entity a ty uložit do databáze.

3.4.7.4 Měsíční poslanecké statistiky

Rozdíl v „měřítku“ pohledu celkových statistik poslance a jednotlivých projevů je dost velký, proto by bylo vhodné mit k dispozici nějakou střední cestu. Tu poskytuje entita *Poslanec_statistiky_mesice*. Tato entita uchovává za každý měsíc statistiky o sentimentu a všech třech typech počtů slov (celková, pozitivní a negativní). Před vytvořením této entity se uvažovalo ještě nad shlukováním statistik po jednotlivých schůzích. Na některé pohledy by to bylo asi vhodnější, ale utrpěla by srozumitelnost. Pojem měsíc je jasný každému a má přesně definovaný začátek i konec. Zato jedna schůze se může táhnout v extrémním případě celým volebním obdobím. Vyhotovení těchto statistik probíhá podobně jako u celkových statistik poslance. Zde je však navíc podmínka na projevy. Program prochází všechny projevy poslance a rozděluje je na „hromádky“ podle data jednacího bodu, který je navázán na projev. Z data nás zajímá pouze měsíc a rok. Když jsou všechny projevy rozděleny, tak program každou „hromádku“ zredukuje do jedné entity *Poslanec_statistiky_mesice* a všechny tyto entity následně nahraje do databáze.

3.4.8 Výstup programu

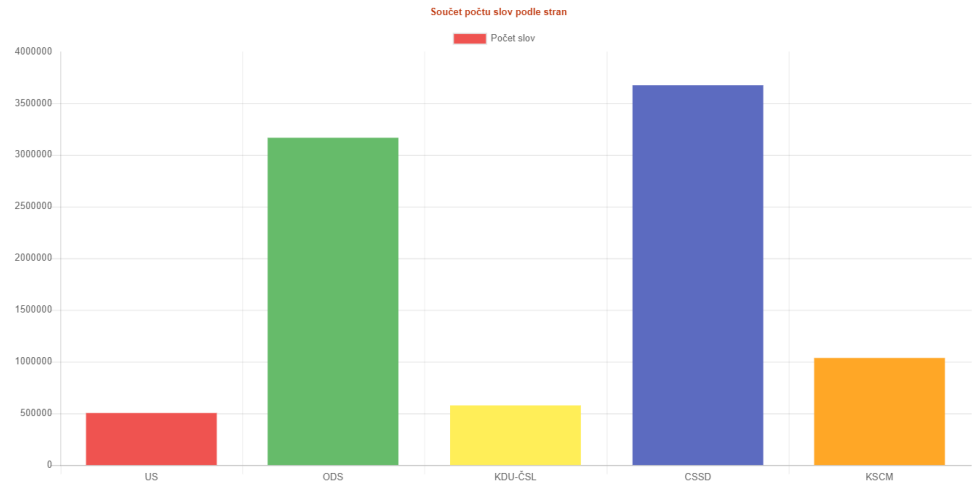
Výstupem tohoto programu jsou entity nahrané v databázi, která se používá pro přenos informaci do prezentačního projektu. Databáze je podrobně popsána v samostatné sekci 3.2.

3.5 Prezentace analyzovaných dat (Třetí projekt)

Analyzovaná a zpracovaná data uložená v databázi je potřeba prezentovat uživateli. Prezentaci těchto dat zajišťuje tento projekt. Projekt využívá frameworku Vaadin v kombinaci s knihovnami Chart.js a Kumo.

3.5.1 Popis běhu programu

V horní části webové stránky jsou umístěna tlačítka, která představují přepínač karet (jakýsi rozcestník). Každé tlačítko po kliknutí zobrazí na stránce jinou kartu. Karty lze rozdělit do tří kategorií. Do první lze zařadit karty,



Obrázek 3.9: Příklad sloupcového grafu

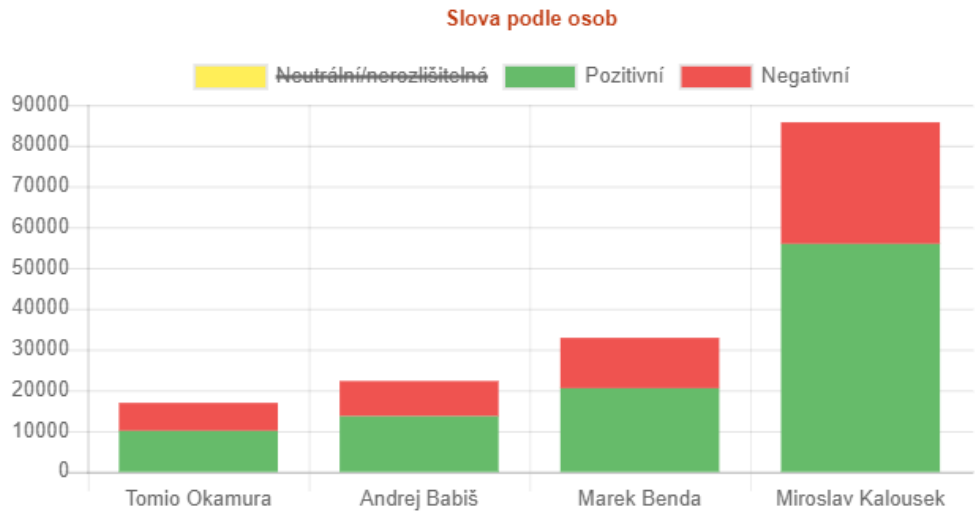
které se zaměřují na jednotlivé entity – poslance, osoby a strany. Druhá kategorie zahrnuje karty srovnávací, zde jde tedy již o interakci a kontext mezi více entitami, srovnání se zaměřují na: osoby, strany a období. Samostatnou kategorií je poslední karta a tou je karta s informacemi. Tyto informace popisují ostatní karty a pomohou uživateli v porozumění toho, jak tato webová stránka funguje. Při spuštění se zobrazí jako první karta Poslanec, která je stejně jako ostatní karty popsána podrobněji dále.

3.5.2 Vizualizační prvky

V této sekci jsou popsány prvky, které prezentují jednotlivé informace. Z těchto prvků se skládají výše popsané karty. Na prezentaci čistě textových údajů jsou využity nástroje, které poskytuje přímo framework Vaadin. Jde o různé formulářové prvky jako jsou – textová pole, tabulky, rolovací seznamy,... Tyto prvky pokryjí pouze základní vizualizaci, která by samotná byla nezajímavá a neupoutala by uživatele.

3.5.2.1 Grafy

Program využívá grafy z knihovny Chart.js. Konkrétně jde o dva typy grafů – sloupcové a skládané sloupcové. Oba typy jsou vždy orientovány vertikálně. Skládané sloupcové grafy se liší od prostých sloupcových tím, že jeden sloupec neznázorňuje jen jednu hodnotu, ale jde o součet více hodnot – výsledná výška sloupce je tedy *složena* z menších sloupců. Sloupce grafů mají standardní popisky a různé barvy pro rozpoznání jednotlivých sloupců, pokud je to potřeba. Příklady grafů poskytují Obrázky 3.9 a 3.10.



Obrázek 3.10: Příklad sloupcového skládaného grafu

3.5.2.2 Slovní mraky

Slovní mraky zprostředkovává knihovna Kumo. Slova mají náhodně přiřazené barvy. Barvy tedy nemají žádnou hlubší spojitost s konkrétním slovem. To stejné platí i pro natočení slov. Význam má ale rozdílná velikost slov. Velikost slova udává jak hodně je dané slovo zastoupeno ve vstupním souboru slov. Uživatel nepozná ze samotného mraku, kolikrát je které slovo použito. Lze ale porovnávat velikost slov mezi sebou a z toho zjistit, která slova jsou častěji zastoupena a která ne. Příklad slovního mraku zobrazuje Obrázek 3.11.

3.5.3 Karta Poslanec

Tento pohled se soustředí na jednu entitu a tou je *Poslanec*. První je potřeba vybrat určitého poslance. K tomu jsou určeny 3 rozevírací seznamy. První obsahuje výběr volebního období, druhý zobrazuje seznam stran, za které byli poslanci zvoleni a třetí umožňuje vybrat konkrétního poslance. Poslance popisuje profil, kde jsou zobrazeny údaje jako jméno, kandidátka, na které byl zvolen, email,... Druhou částí pohledu na poslance jsou statistiky, konkrétně: nejpoužívanější slova, zmínky, měsíční grafy a projevy. Příklad karty je zachycen na Obrázcích 3.12 (profil a nejčastější slova), 3.13 (zmínky) a 3.14 (měsíční grafy).

Nejpoužívanější slova – Na prezentaci zaznamenaných nejpoužívanějších slov používá program dva nástroje – slovní mrak a tabulku se slovy.

Obrázek 3.11: Příklad slovního mraku

Samotná slova program získá z entity *Top_slova* , která je přes *Poslanec_statistiky* navázána na *Poslance*. Předchozí program, který data analyzoval a zpracoval, připravil pro každého poslance maximálně 50 nejpoužívanějších slov. Těchto 50 slov vloží program do slovního mraku, kde se slova zobrazí různě velká podle četnosti. Tabulka zobrazuje všech 50 slov i s přesnými údaji o četnosti a ještě přidává u každého slova podíl tohoto slova ku všem slovům, které poslanec vyřkl. Počet všech vyřčených slov je známý a lehce dosažitelný z entity *Poslanec_statistiky*.

Zmínky – Na zmínky poslanců mezi sebou pohlíží program třemi pohledy. V prvním se zaměřuje na to, které poslance vybraný poslanec zmiňoval nejčastěji. Tento pohled je prezentován tabulkou se jmény a počty zmínek daného poslance. Druhý pohled je inverzí k prvnímu. Jedná se o podobnou tabulku, ale nyní zobrazuje, jací poslanci a kolikrát zmiňovali vy-

PROFIL:

Jméno: Ing. Marian Jurečka

Kandidátka: Křesťanská a demokratická unie - Československá strana lidová

Datum narození: 1981-03-15

Adresa kanceláře: \,\,\

Web poslance na psp.cz: ODKAZ

Email: \

Web: \

Facebook: \

STATISTIKY:

Počet slov: 93847 (průměrný počet slov poslance ve volebním období: 51904,405)

Celkový sentiment: 0,404 (průměrný sentiment poslance ve volebním období: 0,372)

Nejčastěji používaná slova poslance:

Slovo	Počet výskytů	Podíl z celé slovní zás...
český	365	0,389%
zemědělský	315	0,336%
zemědělství	313	0,334%
republika	295	0,314%
státní	259	0,276%
ministerstvo	257	0,274%
koruna	226	0,241%
program	198	0,211%
podpora	185	0,197%
zemědělec	176	0,188%
víno	171	0,182%
případ	161	0,172%

Obrázek 3.12: Příklad karty Poslanec – 1. část

Obrázek 3.13: Příklad karty Poslanec – 2. část

braného poslance. Třetí pohled ukazuje, jak vybraný poslanec zmiňoval celé strany. Tato data se sbírají znovu přes zmínky poslanců, ale program provede zobecnění na strany. Tento pohled reprezentují dva grafy. První z grafů ukazuje ve sloupcích počty zmínek za jednotlivé strany. Druhý graf ukazuje, kolikrát vybraný poslanec průměrně zmínil každého poslance z dané strany. Simuluje tedy částečně situaci, kdy by všechny strany měly stejný počet poslanců. Obecně totiž bývá typické, že poslanec častěji zmiňuje poslance té strany, která má ve Sněmovně více poslanců. Této simulace program docílí tím, že dělí absolutní počty zmínek počtem poslanců dané strany.

Měsíční grafy – K poslanci se vážou dva grafy s údaji po jednotlivých měsících. Jeden se zaměřuje na slova a druhý na sentiment. Zdrojem dat k oběma grafům jsou projevy vybraného poslance. Projevy jsou rozděleny podle měsíců, v kterých byly projednávány body, ke kterým projevy přísluší. Počtem slov za měsíc je poté součtem počtů slov všech projevů daného měsíce. Sentiment za měsíc je vypočítaný ze sentimentu

Obrázek 3.14: Příklad karty Poslanec – 3. část

všech slov všech projevů daného měsíce. Opět je zde použita optimalizace s tím, že počty negativních a pozitivních slov jsou uloženy u projevů a není tedy potřeba zacházet až na úroveň jednotlivých slov.

Projevy – Program zobrazuje i všechny projevy, které poslanec pronesl. Projevy jsou strukturovány po jednotlivých poslaneckých schůzích v rozbalovacích seznamech. U celých schůzí jsou zobrazeny následující informace – číselné označení, počet slov a sentiment všech projevů vybraného poslance v rámci konkrétní schůze. Každý projev má u sebe informace – název jednacího bodu, ke kterému se projev váže, počet slov a sentiment celého projevu.

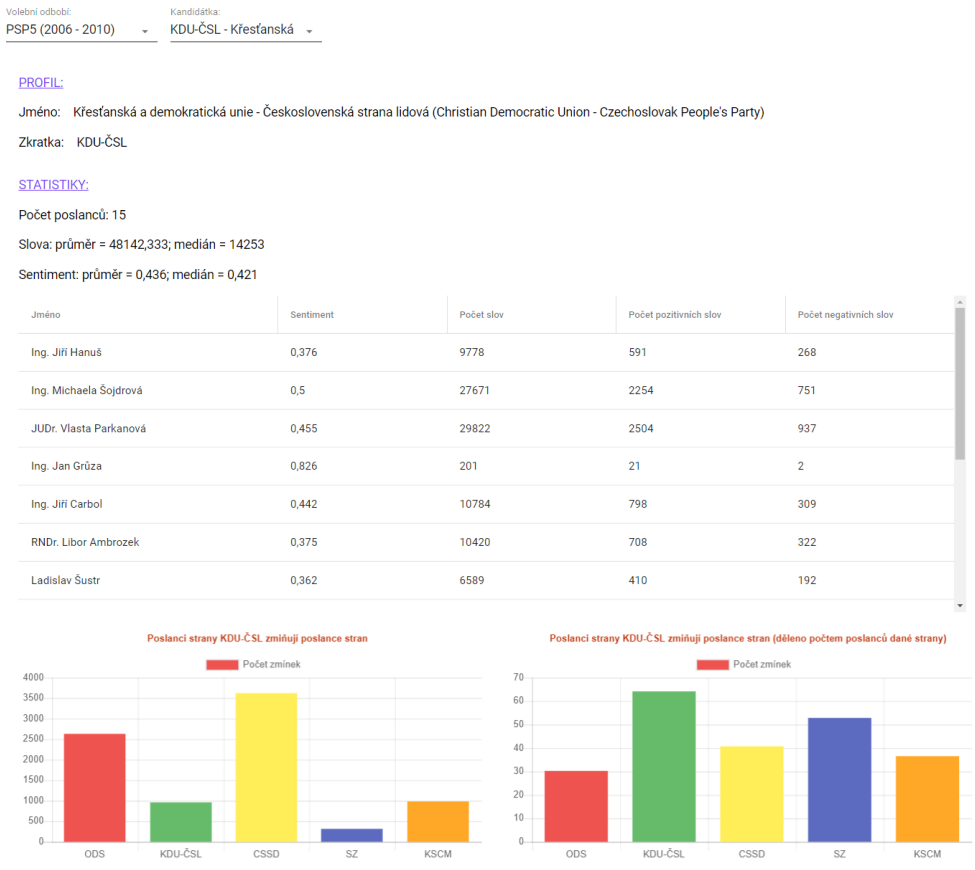
Obrázek 3.15: Příklad karty Osoba

3.5.4 Karta Osoba

Na této kartě si uživatel vybere osobu ze seznamu všech osob, které kdy měly poslanecký mandát. Původní seznam osob, který poskytuje Poslanecký sněmovna zahrnuje i osoby, které nikdy nebyly poslancem. Tyto osoby byly už dříve odstraněny, jak je popsáno podrobněji v části 3.4.1. Tento postup přináší z uživatelského hlediska zjednodušení výběru osoby a zároveň nevytváří zbytečně prázdné profily osob, které by neměly žádné statistiky, což by mohlo působit zmateně. Po výběru ze seznamu se uživateli zobrazí profil osoby, který je stejný jako v případě poslance, přesněji se použijí informace z nejnovějšího poslaneckého mandátu vybrané osoby. Příklad této karty dokresluje Obrázek 3.15.

Další část tvoří dva grafy. Oba jsou prosté sloupcové a u obou představuje jeden sloupec jedno období. První graf vyjadřuje sentiment poslance za období a druhý počet slov za období. U osob s více poslaneckými mandáty lze tedy vidět vývoj v delším časovém intervalu.

Poslední část je rolovací seznam, v kterém lze vybrat jedno z období, kdy vybraná osoba byla poslancem. Po výběru se pod seznamem objeví statistiky poslance jako jsou na kartě Poslanec – nejpoužívanější slova, zmínky, měsíční grafy a projevy. Původní myšlenkou bylo načítat všechny poslanecké mandáty vybrané osoby rovnou. Existují ale dva hlavní důvody, proč se poslanecký



Obrázek 3.16: Příklad karty Strana

mandát u konkrétního člověka vybírá jednotlivě. Za prvé při načtení statistik všech poslaneckých mandátů pod sebe docházelo k dlouhému načítání, což by kazilo uživatelský zážitek. Druhým důvodem je to, že u některých osob s více poslaneckými mandáty by stránka byla moc dlouhá, v čemž by se uživatel složitě orientoval.

3.5.5 Karta Strana

Tato karta popisuje poslance zvolené za jednu stranu v jednom období. Na začátku uživatel vybere pomocí dvou rozevíracích seznamů volební období a stranu. Po vybrání strany se zobrazí její statistiky. Nahoře je krátký profil – jméno a zkratka. Pod profilem jsou souhrnné statistiky pro celou stranu – průměry a mediány počtu slov a sentimentu. Tuto kartu si lze lépe představit pomocí Obrázku 3.16.

Následuje tabulka, kde jeden řádek představuje jednoho poslance. Tabulka

má dále čtyři sloupce se statistikami – sentiment, počet slov, počet negativních a počet pozitivních slov. Tabulku lze řadit podle kteréhokoliv sloupce a je možno jednoduše zjišťovat, který poslanec pronesl nejméně slov nebo kdo má nejvyšší sentiment.

Pod tabulkou jsou dále dva grafy. V obou případech jde o prosté sloupcové grafy, kde jeden sloupec představuje jednu stranu. Grafy vyjadřují, jak poslanci vybrané strany zmiňují poslance jiné strany. První zobrazuje absolutní čísla, druhý hodnotu u každé strany vydělí jejím počtem členů ve sněmovně. Sloupec pak tedy říká informaci, kolikrát průměrně byl zmíněn každý poslanec dané strany.

3.5.6 Srovnávací karty

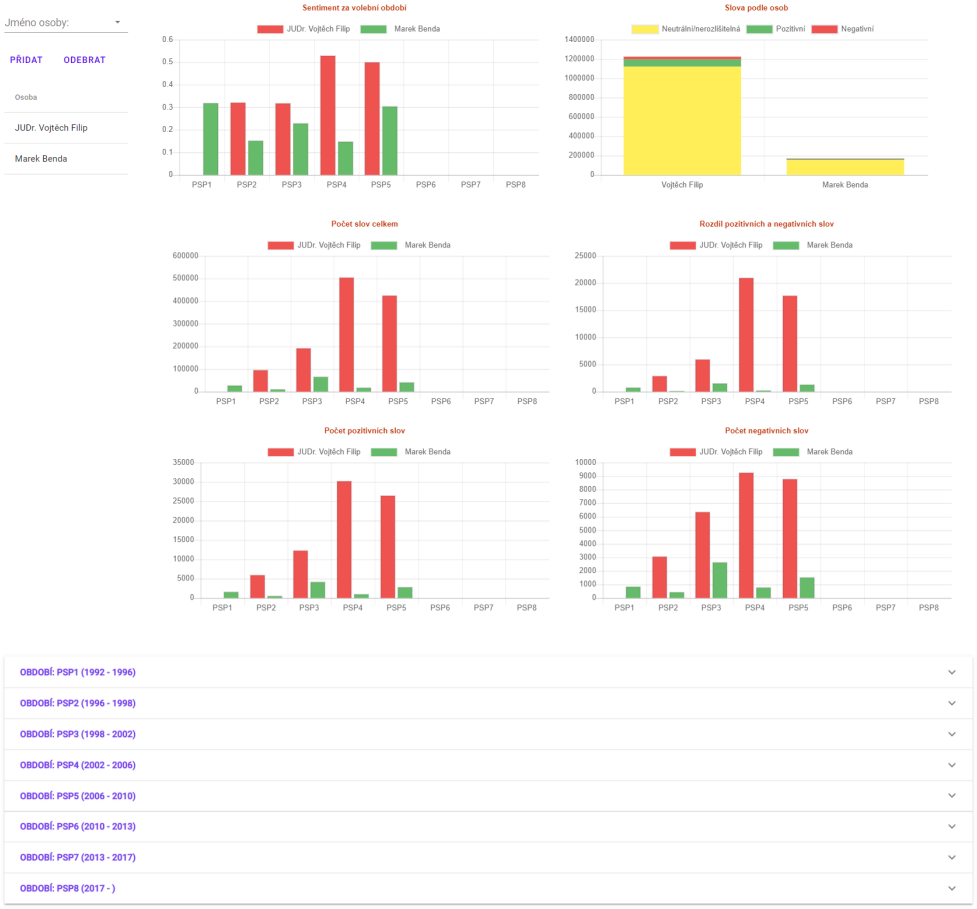
Následující sekce se věnují kartám, které porovnávají více zástupců stejné entity.

3.5.6.1 Karta Srovnání Osob

Na této kartě lze srovnávat jednotlivé zástupce entity Osoba mezi sebou. Uživatel si na začátku vybere množinu osob, které chce porovnávat. Už po přidání první osoby se na stránce objeví množství grafů a dalších statistik. Srovnání pro lepší orientaci ukazuje Obrázek 3.17.

Na začátku stránky popisuje srovnání osob šest grafů. U pěti z nich jde o prosté sloupcové grafy, kde jeden sloupech vždy představuje jedno volební období. Grafy se zaměřují na sentiment, počet slov a také na počty negativních a pozitivních slov. Šestý graf se zaměřuje na počet slov. Jedná se o graf skládaný sloupcový, kde jeden sloupec představuje jednu osobu. Každý sloupec se skládá ze tří částí. Jde o počty pozitivních, negativních a neutrálních slov. Velikost celého složeného sloupce tedy vypovídá o celkovém počtu slov dané osoby. V tomto grafu lze vidět i to, že slova bez sentimentu mají většinové zastoupení v projevech asi všech osob (odpozorováno). To je dáno tím, že studovaná doména (Poslanecká sněmovna) nabízí projevy s málo výrazným pozitivním nebo negativním sentimentem a dalším důvodem může být i velikost slovníku s pozitivními a negativními slovy.

Pod šesticí grafů je ještě oblast, která se zaměřuje na jednotlivá období. Zobrazují se jen ta období, v kterých byla alespoň jedna z vybraných osob poslancem. Pro každé období jsou dostupné dva grafy. Oba jsou sloupcové, kde jeden sloupec odpovídá jednomu měsíci. Jeden graf se zaměřuje na počet slov, druhý na sentiment. V grafech se zobrazí vždy všichni poslanci daného období, kteří mají vazbu na některou z vybraných osob. Pomocí grafů lze tedy osoby porovnávat podrobněji. Toto srovnání využívá stejné grafy, které byly popsány už v popisu karty Poslanec.



Obrázek 3.17: Příklad karty Srovnání osob

3.5.6.2 Karty Srovnání stran a Srovnání období

Tyto dvě karty mají stejnou strukturu obsahu a tou je šest grafů. U karet se ale liší pohled na doménu. Srovnání stran se zabývá stranami – je zde tedy potřeba na začátku vybrat období, respektive množinu stran. Srovnání období srovnává všechna volební období Poslanecké sněmovny. Jeden sloupec grafu vždy vyjadřuje období nebo stranu. První trojice grafů se zaměřuje na slova. První graf ukazuje celkové počty slov, druhý průměry počtu slov na jednoho poslance a třetí medián počtu slov na jednoho poslance. Druhá trojice grafů vizualizuje informace o sentimentu. První graf ukazuje celkový sentiment – sentiment spočítaný ze všech projevů. Druhý graf představuje průměr sentimentu na jednoho poslance a třetí medián sentimentu. Strukturu obou výše popsaných karet si lze lépe představit s pomocí Obrázku 3.18.

Obrázek 3.18: Příklad karty Srovnání stran

3.5.7 Karta Info

Tato karta slouží z větší části jako nápověda. Jsou v ní popisy ostatních karet, aby se mohl uživatel poučit v tom, jak program používat. Kromě popisů je zde ještě přepínač tmavého a světlého tématu.

3.6 Shrnutí a diskuze

3.6.1 Testování

Testování všech tří programů proběhlo dvojím způsobem. V každém programu jsou unit testy z knihovny JUnit, které testují činnost jednotlivých funkcí obsažených v programech. Většinou se jedná o funkce, které se v programech používají opakovaně a jejich chybovost by mohla způsobit největší problémy.

Testování práce s databází a s webovou částí se testuje automatizovanými testy složitě. Vzhledem k tomu, že je dostupné velké množství dat, pro které je celý systém navrhován, testovalo se dále pomocí nahrání těchto reálných dat do programů. Poté proběhl průchod každou kartou zvlášť. Na každé kartě se simulovala předpokládaná uživatelská aktivita. Při všech nahraných datech byl zaznamenám problém s delším načítáním obsahu některých karet (hlavně karta Poslanec). Tento problém je pravděpodobně způsoben velkým počtem projevů v databázi. Tato chyba neovlivňuje výsledky nástroje, ale omezuje uživatelský komfort, proto by měla být odstraněna před případným nasazením a zveřejněním pro širší veřejnost.

Vizuální i logická část webové aplikace byla podrobena také širší skupině lidí (rodina, přátelé,...), kteří také pomohli odhalit chyby nebo přidali návrhy na zlepšení, za což jim i zde děkuji. Nejčastěji šlo o chyby vizuální, například: velikosti textů, nelogičnosti členění prvků nebo chybějících popisků, jejichž absence narušovala přehlednost stránky. Návrhů na zlepšení bylo mnoho a většinou šlo o nápady, které již byly v nějaké fázi zpracování a šlo tedy o změnu nebo rozšíření daného nápadu. Jako konkrétní příklad lze uvést zařazení grafů s mediány různých souborů dat. Grafy s mediány přináší ve spolupráci s grafy s průměry nové možnosti porovnávání osob, stran,... Tito „testeři“ neměli žádný pevně daný testovací scénář, ale spíše simulovali běžného uživatele, takže si aplikaci zkoušeli podle sebe.

3.6.2 Identifikované problémy domény a jejich řešení

Zde jsou popsány problémy, které se vyskytovaly ve více částech programu a jsou tedy popsány v této samostatné části.

Poslanci bez kandidátky – Někteří *Poslanci* nemají vazbu na žádnou kandidátku (*Orgán*). To je z principu nemožné, protože v ČR jsou a vždy byly volby do Poslanecké sněmovny soutěží politických stran a každý

budoucí poslanec musí být zvolen za některou z nich. Problém byl objeven v rámci prvního volebního období, kde žádný *Poslanec* nemá vazbu na kandidátku. Některé funkce programu ale předpokládají příslušnost *Poslance* ke kandidátce a je tedy potřeba, aby každý *Poslanec* nějakou kandidátku měl. Z počátku byl tento problém řešen až na úrovni prezentačního programu, protože jde o problém v rámci prezentace dat. Problém byl řešen tak, že se *Poslanci* bez kandidátky přiřadila vymyšlená kandidátka s názvem „Strana neurčena". Bohužel počet míst, kde by se musela kandidátka takto uměle přidávat bylo mnoho a došlo se k tomu, že se těmto *Poslancům* přidá vymyšlená kandidátka už v programu, který zpracovává a analyzuje data. To s sebou přinese výhodu, že v rámci zpracování dat je úprava snadná a v prezentačním programu už nejsou potřebné žádné úpravy.

Počet členů strany – Každé straně zvolené do Poslanecké sněmovny připadne po volbách určitý počet mandátů. Součet mandátů všech stran musí být roven 200 (počet poslanců). Program ale nezná tento počet mandátů pro každou stranu. Program zná počet lidí, kteří za danou stranu do Poslanecké sněmovny v určitém období nastoupili. Někdy se stává, že některý poslanec svůj mandát nedokončí a je nahrazen někým jiným ze stejné strany (resp. kandidátky). Strana má tedy pořád stejný počet mandátů, ale počet poslanců se může zvyšovat. Některé funkce programu pracují právě s počtem poslanců, kteří za určitou stranu do Sněmovny nastoupili. Správné řešení by ale bylo použít počet mandátů.

Počet mandátů je v každém období 200, počet poslanců je ale průměrně 216. Změn tedy není zanedbatelně, ale typicky bývají změny rozprostřeny mezi všechny strany a vliv používání počtu poslanců místo počtu mandátů tedy není tak velký. Do budoucna by pro zpřesnění bylo lepší použít počty mandátů. Zde by ale bylo zapotřebí připojit nový zdroj dat.

Poslanci mohou měnit strany – Poslanec XY byl zvolen za stranu A. Po libovolně dlouhé době poslanec XY stranu (resp. poslanecký klub) opustí a může se z něj stát nezařazený poslanec bez příslušnosti k nějakému klubu nebo může přejít k jinému klubu. Tyto „přestupy" program nedetekuje a každý poslanec má v rámci jednoho období vždy pevně přiřazenou stranu, na jejíž kandidátce byl zvolen. Poslancovy projevy se počítají stále do projevů strany A, i když poslanec je velkou část volebního období v poslaneckém klubu strany B nebo je nezařazený.

Nyní záleží na tom, jestli je tento postup chybný. Volič volí politickou stranu, která je zastoupena poslanci. Volič nepromlouvá do změn poslanců mezi kluby. Z pohledu voliče je tedy zajímavá ta množina poslanců, kterou volil. Dává tedy smysl, když může zjistit informace o této skupině a ne o skupině, která by se v průběhu času měnila. Na druhou stranu poslanci reflektují často názory svého poslaneckého klubu

a chování poslance po „přestupu“ se může diametrálně lišit, takže tím by dávalo smysl reflektovat změny v příslušností poslancům ke klubům. Oba postoje mají svá pro i proti. V rámci této práce byl ale použit první přístup, protože převážily důvody pro jeho přijetí.

3.6.3 Možná budoucí rozšíření

V této sekci jsou popsány nápady, které se bohužel do rozsahu práce nevešly, ale jsou zajímavé a zasloužily by si podle názoru autora realizaci nebo alespoň podrobnější rozebrání.

Tabulka používaných slov a poslanců – Představa je taková, že by se jednalo o tabulku, kde by řádky označovaly jednotlivé poslance a sloupce slova. Obě tyto množiny (poslance a slova) by si mohl uživatel volně nastavit. V buňce tabulky, která má souřadnice určené konkrétním poslancem a slovem by byl počet výskytů slova v projevech poslance. Za úvahu stojí, jestli jako poslance brát jeden poslanecký mandát nebo se zaměřit na osoby a mít tedy všechny mandáty jedné osoby vždy pod touto osobou. Problémem by mohla být časová náročnost takového úkolu při zachování současného návrhu, protože by se u výřečných poslanců procházelo velké množství entit se slovy. Jedná se však zatím o domněnku, která není nijak prověřena. Použití této funkce by bylo vhodné pro porovnávání konkrétních poslanců. Bylo by možné například porovnávat pohled na zahraniční politiku a použít dvojici slov „západ“ a „východ“. Zajímavé by mohlo být i porovnání slov „evropský“ a „český“. Funkce by ale uměla porovnávat i více slov než pouhé dvojice.

Nejpoužívanější stopslova – Většinou je potřeba se stopslov zbavit. Mohly by se ale najít případy, kdy by mohla i statistika stopslov nést nějakou informaci. Kdyby byl u každého poslance seznam nejpoužívanějších stopslov, tak by to asi neříkalo nějaké objevné informace, ale zajímavé použití by mohlo být třeba v porovnání slov „já“ a „my“. Z toho by mohlo být v určitém zjednodušení vidět, jestli je daný poslanec spíše týmový hráč nebo se projevuje víc sám za sebe.

Statistiky období – V programu lze porovnávat všechna volební období mezi sebou – karta Srovnání období. Program by ale v budoucnu ještě mohl zobrazovat mnoho statistik pro jednotlivá volební období. Například by se mohly vyhodnocovat různé žebříčky s poslanci s nejvíce slovy, nejméně slovy, nejvyšším sentimentem a podobně.

Počty projevů – Program se zaměřuje hlavně na počet slov a sentiment. Jsou zde ale i další údaje, které by bylo možné porovnávat. Jedním z nich by mohl být počet projevů. Projevy poslanců mají různou délku a mohlo by být zajímavé třeba to, kolik slov použije poslanec průměrně v jednom projevu nebo jednoduše kolik projevů poslanec pronesl.

Tituly poslanců – Další částí dat, kterou se program nezaobírá, jsou tituly poslanců. Myslím, že by mohlo být zajímavé pozorovat, jaké tituly, a částečné tedy i obory, v Poslanecké sněmovně mají vyšší nebo nižší zastoupení. Roste nebo klesá počet lidí s vysokoškolským titulem? Zvyšuje se například počet vystudovaných lékařů nebo právníků? Tyto informace by mohly být z aktuálního zpracování docela snadno získatelné.

3.6.4 Diskuze nad možným výkladem výstupů

Zde je sepsáno několik postřehů, které mají dokreslit, jak by šlo informace získané díky této práci využít a vykládat je. Nejsem odborníkem v oblasti sociologie ani politologie, takže se jedná jen o osobní domněnky. Všechny domnělé výklady jsou apolitické. Politické výklady by ukázaly jistě ještě širší využití, bylo by možné třeba konfrontovat výroky konkrétních politiků s výstupy programu.

Vývoj sentimentu za období – Z grafu sentimentu v jednotlivých obdobích na kartě Srovnání období lze vidět, že sentiment se nemění nijak dramaticky, ale i tyto relativně malé rozdíly by bylo možné připsat různým náladám v Poslanecké sněmovně, resp. v celé společnosti.

Vývoj počtu slov za období – Zde je to podobné jako u předchozího příkladu. U slov jsou ale rozdíly mezi obdobími relativně vetší. Například v posledních několika obdobích lze vidět docela znatelný vzestup, ne vždy ale tato statistika rostla. Pozor je také nutné dát u období, která skončila předčasně a nelze je tedy plnohodnotně porovnávat s těmi ostatními.

Předsedající schůzí mají vyšší sentiment – Řízení sněmovny vždy vede předsedající. V této funkci se střídají předseda a místopředsedové Poslanecké sněmovny. Tito poslanci mají typicky vyšší sentiment, než kolik je průměrný sentiment všech poslanců. Jedná se jen o domněnku, ale mohlo by to být způsobené tím, že poslanci z předsednictva Sněmovny často používají krátké věty k tomu, aby ostatní poslance uvedli k řečnickému pultu. Při těchto uváděních používají předsedající slova s kladným sentimentem. Příkladem může být tento projev, kterou předsedající použil: „Děkuji, pane místopředsedo. S přednostním právem je přihlášen pan předseda Bartošek. Prosím, pane předsedo, máte slovo.“. Už podle prvního pohledu na projev lze poznat, že celkové vyznění projevu má kladný sentiment. Podobných projevů pronese každý předsedající mnoho, a proto to jeho celkový sentiment ovlivňuje.

Vládní strany mají vyšší sentiment – Při pohledu na historické složení vlád České republiky lze vidět to, že strany, které tyto vlády tvořily, měly typicky vyšší sentiment než strany, které byly v opozici. To intuitivně dává smysl, protože vláda typicky chválí to, co udělala, a prezentuje v dobrém věci, které plánuje. Opozice to na druhou stranu často kritizuje a hledá chyby, což může nést projevy s relativně negativnějším sentimentem.

Závěr

Hlavním cílem práce bylo vypracovat nástroj na analýzu poslaneckých projevů. Pro dosažení tohoto cíle i potřebných dílčích cílů byly vytvořeny 3 Java projekty.

První projekt se zabývá stažením dat o poslancích, která jsou následně potřeba pro splnění dalších cílů. V této části byly využity výstupy z teoretické části, konkrétně z části o postupech extrakce dat z webu.

Druhý projekt má na starosti zpracování stažených dat. Pro tento účel byly použity poznatky z teoretické části, konkrétně z části o extrakci dat z textu. Dále byla na zpracovaných datech provedena analýza, která čerpala opět z výstupů teoretické části, kde se ukázal jako vhodný nástroj na analýzu textu nástroj MorphoDiTa.

Třetí projekt má v popisu práce zobrazovat zpracované a zanalyzované informace v podobě, která je vhodná a vizuálně příjemná pro poučeného uživatele. K tomuto účelu byl použit framework Vaadin, který byl podpořen knihovnou pro vytváření grafů. Jde o knihovnu Chart.js, která je upravena pro použití v jazyce Java. Dále vizualizaci zprostředkovala knihovna Kumo, která umí vytvářet slovní mraky.

V budoucnosti by bylo možné nástroj rozšířit o data ze Senátu Parlamentu ČR nebo by bylo možné čerpat ze zahraničních zdrojů a srovnat do určité míry různé státní parlamenty mezi sebou. Nabízí se také vylepšení aktuálně používaných metod, které nástroj využívá, pro dosažení přesnějších výstupů. Cestou dalšího rozvoje by mohla být i kombinace s jinými zdroji, které by zasadily aktuální výstupy do širšího kontextu.

Literatura

[1] Ústava ČR. *Parlament České republiky [online]*, prosinec 1992, [cit. 2020-04-08]. Dostupné z: `https://www.psp.cz/docs/laws/constitution.html`

[2] Poslanci a orgány. *Parlament České republiky [online]*, duben 2020, [cit. 2020-04-08]. Dostupné z: `https://www.psp.cz/sqw/hp.sqw?k=182`

[3] Jednací řád Poslanecké sněmovny ČR. *Parlament České republiky [online]*, 1995, [cit. 2020-04-08]. Dostupné z: `https://www.psp.cz/docs/laws/1995/90_index.html`

[4] Digitální knihovna Poslanecké sněmovny ČR. *Parlament České republiky [online]*, [cit. 2020-04-08]. Dostupné z: `https://www.psp.cz/eknih/index.htm`

[5] Stenoprotokoly. *Parlament České republiky [online]*, [cit. 2020-04-08]. Dostupné z: `https://www.psp.cz/sqw/hp.sqw?k=1352`

[6] Projekt Digitální knihovna Poslanecké sněmovny ČR. *Parlament České republiky [online]*, 2016, [cit. 2020-04-08]. Dostupné z: `https://www.psp.cz/sqw/hp.sqw?k=2032`

[7] Poslanci a osoby. *Parlament České republiky [online]*, [cit. 2020-04-08]. Dostupné z: `https://www.psp.cz/sqw/hp.sqw?k=1301`

[8] O serveru. *Hlídač státu [online]*, [cit. 2020-04-08]. Dostupné z: `https://www.hlidacstatu.cz/texty/o-serveru/`

[9] Databáze Stenozáznamy Poslanecké sněmovny Parlamentu ČR. *Hlídač státu [online]*, [cit. 2020-04-08]. Dostupné z: `https://www.hlidacstatu.cz/data/Index/stenozaznamy-psp`

[10] Scime, A.: *Web Mining: Applications and Techniques*. ITPro collection, Idea Group Pub., 2005, ISBN 9781591404163. Dostupné z: https://books.google.cz/books?id=iRi9AQAAQBAJ

[11] Zpracování přirozeného jazyka aneb NLP. *https://nlp.fi.muni.cz/ [online]*, [cit. 2020-04-08]. Dostupné z: https://nlp.fi.muni.cz/cs/ZpracovaniPrirozenehoJazyka

[12] Liu, B.: *Opinions, Sentiment, and Emotion in Text*. Sentiment Analysis: Mining Opinions, Sentiments, and Emotions, Cambridge University Press, 2015, ISBN 9781107017894. Dostupné z: https://books.google.cz/books?id=6IdsCQAAQBAJ

[13] Veselovská, K.; Bojar, O.: Czech SubLex 1.0. 2013, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. Dostupné z: http://hdl.handle.net/11858/00-097C-0000-0022-FF60-B

[14] Uveďte původ-Neužívejte dílo komerčně-Zachovejte licenci 3.0 Unported (CC BY-NC-SA 3.0). *https://creativecommons.org/*, [cit. 2020-04-08]. Dostupné z: https://creativecommons.org/licenses/by-nc-sa/3.0/deed.cs

[15] Kramer, O.: *Dimensionality Reduction with Unsupervised Nearest Neighbors*. Intelligent Systems Reference Library, Springer Berlin Heidelberg, 2013, ISBN 9783642386527. Dostupné z: https://books.google.cz/books?id=pU4_AAAAQBAJ

[16] Samudrala, S.: *Machine Intelligence: Demystifying Machine Learning, Neural Networks and Deep Learning*. Notion Press, 2019, ISBN 9781684660834. Dostupné z: https://books.google.cz/books?id=LC2DDwAAQBAJ

[17] TOKENIZACE. *https://www.czechency.org/*, [cit. 2020-04-08]. Dostupné z: https://www.czechency.org/slovnik/TOKENIZACE

[18] LEMMATIZACE. *https://www.czechency.org/*, [cit. 2020-04-08]. Dostupné z: https://www.czechency.org/slovnik/LEMMATIZACE

[19] Laplante, P.: *Encyclopedia of Computer Science and Technology*. CRC Press, 2017, ISBN 9781351645799. Dostupné z: https://books.google.cz/books?id=Dx86DwAAQBAJ

[20] keyword. *https://www.dictionary.com/*, [cit. 2020-04-08]. Dostupné z: https://www.dictionary.com/browse/keyword

[21] Java. *Oracle [online]*, [cit. 2020-04-08]. Dostupné z: https://www.oracle.com/java/technologies/

[22] Java – dnes při šálku dobré kávy. *Linuxexpres [online]*, duben 2007, [cit. 2020-04-08]. Dostupné z: https://www.linuxexpres.cz/praxe/java-dnes-pri-salku-dobre-kavy

[23] jsoup: Java HTML Parser. *jsoup.org [online]*, [cit. 2020-04-08]. Dostupné z: https://jsoup.org/

[24] Welcome to Apache Commons. *commons.apache.org*, [cit. 2020-04-08]. Dostupné z: https://commons.apache.org/index.html

[25] APACHE LICENSE, VERSION 2.0. *apache.org*, [cit. 2020-04-08]. Dostupné z: http://www.apache.org/licenses/LICENSE-2.0

[26] Různé licence a komentáře k nim. *https://www.gnu.org/*, [cit. 2020-04-08]. Dostupné z: https://www.gnu.org/licenses/license-list.cs.html

[27] What is free software? *https://www.gnu.org/*, [cit. 2020-04-08]. Dostupné z: https://www.gnu.org/philosophy/free-sw.html

[28] Commons IO. *commons.apache.org*, [cit. 2020-04-08]. Dostupné z: https://commons.apache.org/proper/commons-io/

[29] Commons Lang. *commons.apache.org*, [cit. 2020-04-08]. Dostupné z: https://commons.apache.org/proper/commons-lang/

[30] Miller, F.; Vandome, A.; McBrewster, J.: *Levenshtein Distance*. VDM Publishing, 2009, ISBN 9786130216900. Dostupné z: https://books.google.cz/books?id=TTzhQgAACAAJ

[31] MorphoDiTa. *http://lindat.mff.cuni.cz/ [online]*, [cit. 2020-04-08]. Dostupné z: http://lindat.mff.cuni.cz/services/morphodita/

[32] POSITIONAL TAGS. *http://ufal.mff.cuni.cz/ [online]*, [cit. 2020-04-08]. Dostupné z: http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/m-layer/html/ch02s02s01.html

[33] Mozilla Public License Version 2.0. *https://www.mozilla.org/ [online]*, [cit. 2020-04-08]. Dostupné z: https://www.mozilla.org/en-US/MPL/2.0/

[34] What Is SQLite? *sqlite.org [online]*, [cit. 2020-04-08]. Dostupné z: https://www.sqlite.org/index.html

[35] *https://vaadin.com/ [online]*, [cit. 2020-04-08]. Dostupné z: https://vaadin.com/

[36] chartjs. *https://vaadin.com/*, [cit. 2020-04-08]. Dostupné z: https://vaadin.com/directory/component/chartjs

[37] Chart.js. *https://www.chartjs.org/*, [cit. 2020-04-08]. Dostupné z: `https://www.chartjs.org/`

[38] Charts. *https://vaadin.com/*, [cit. 2020-04-08]. Dostupné z: `https://vaadin.com/components/vaadin-charts`

[39] Kumo - Java Word Cloud. *https://github.com/*, [cit. 2020-04-08]. Dostupné z: `https://github.com/kennycason/kumo`

[40] JUnit 5. *https://junit.org/*, [cit. 2020-04-08]. Dostupné z: `https://junit.org/junit5/`

[41] Unit testy v Javě a JUnit. *https://www.itnetwork.cz/*, [cit. 2020-04-08]. Dostupné z: `https://www.itnetwork.cz/java/testovani/java-unit-testy-v-junit`

[42] htm. *jakpsatweb.cz [online]*, [cit. 2020-04-08]. Dostupné z: `https://www.jakpsatweb.cz/enc/htm.html`

[43] .unl Přípona souboru. *https://soubory.info/*, [cit. 2020-04-08]. Dostupné z: `https://soubory.info/extension/unl`

[44] Co to je UNL soubor (ve vztahu k Ubyportu). *https://www.policie.cz/*, [cit. 2020-04-08]. Dostupné z: `https://www.policie.cz/clanek/unl-soubor.aspx`

Příloha **A**

Seznam použitých zkratek

HTML Hypertext Markup Language

IO Input Output

JIT Just In Time Compiler

JVM Java Virtual Machine

NLP Natural Language Processing)

Příloha **B**

Obsah přiloženého USB flash disku

- `readme.txt` ..... stručný popis obsahu USB flash disku
- `exe` ..... adresář se spustitelnou formou implementace
- `src`
  - `impl` ..... zdrojové kódy implementace
  - `thesis` ..... zdrojová forma práce ve formátu LaTeX
- `text` ..... text práce
  - `thesis.pdf` ..... text práce ve formátu PDF