VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

ANALÝZA OBSAHU SOCIÁLNÍCH SÍTÍ TÝKAJÍCÍ SE ČESKÝCH MOBILNÍCH OPERÁTORŮ

ANALYSIS OF SOCIAL MEDIA CONTENT DISCUSSING CZECH MOBILE OPERATORS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **JAN PAVLŮ**
AUTHOR

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

BRNO 2017

Abstrakt

Tato práce se zabývá analýzou postojů u příspěvků ze sociálnch sítí týkajících se českých mobilních operátorů. Kromě analýzy postojů se zaměřuje na vizualizací stažených a analyzovaných dat. Analýza postojů je provedena za pomocí strojového učení s učitelem. Po stažení jsou příspěvky očištěny, lemmatizovány a převedeny na vektor příznaků. Pro klasifikaci se využívá *Stochastic Gradient Descent*. Analyzovaná data jsou zobrazena jak ve formě diagramů, tak ve tvaru seznamu příspěvků. Systém poskytuje i automatické přiřazení kategorií příspěvkům pomocí stejného principu. Při přiřazení postojů systém dosahuje úspěšnosti okolo 75%. Při přiřazení kategorií je sice vysoká přesnost (kolem 80%), ale nízká preciznost, navrátovost a F1 score(20% - 30%). Proto se automaticky neprovádí. Přínosem systému je, dokáže automaticky sbírat data z různých zdrojů, ta analyzovat a přehledně zobrazit. Také poskytuje prostředky, jak manuálně měnit přířazené hodnocení/kategori, což při občasném zásahu uživatele povede k postupnému zlepšování charakteristik systému.

Abstract

The main topic of this thesis is sentiment analysis of posts obtained from a social networks. The posts are about czech mobile network operators. The essential part of implemented system is also data visualization. The sentiment analysis is done using machine learning techniques. Downloaded posts are cleaned, lemmatized and transformed to feature vectors. *Stochastic Gradient Descent* algorithm is used for classification. Analyzed data are visualized in charts and as the list of posts. The system provides tools for text categorization. The accuracy, precision, recall and F1 score of sentiment analysis is about 75%. The accuracy of post categorization is high (about 80%), but precision, recall and F1 score are low (about 30%). This is the reason why post categorization isn't automatically done. The benefit of the system it that it automatically collects data from different sources, analysis them and displays them. It also provides tools for manual change of sentiment/categories which can lead to better system characteristics with some help of users.

Klíčová slova

sentiment, analýza sentimentu, analýza postojů, mobilní operátor, dolování dat, analýza textu, aspekty, strojové učení, klasifikace dokumentů

Keywords

sentiment, sentiment analysis, data minign, text mining, mobile network operators, machine learning, document classification

Citace

PAVLŮ, Jan. *Analýza obsahu sociálních sítí týkající se českých mobilních operátorů*. Brno, 2017. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Doc. RNDr. Pavel Smrž, Ph.D.

Analýza obsahu sociálních sítí týkající se českých mobilních operátorů

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana docenta Smrže. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Jan Pavlů
15. května 2018

Poděkování

Rád bych poděkoval svému vedoucímu za poskytnutí odborné pomoci při tvorbě této práce. Dále bych chtěl poděkovat všem uživatelům, kteří se zúčastnili testování systému. Obzvláště Tomáši Bártů, který odhalil chybu se špatným indexováním měsíců na stránce s příspěvky a Dominiku Roháčkovi, jenž mě donutil v produkční verzi vypnout debugovací režim. Také bych chtěl poděkovat své rodině, která mě během období, kdy jsem bakalářskou práci psal, neustále obtěžovala dotazy typu: „Kdy už to budeš mít hotový?“

Obsah

Kapitola 1

Úvod

V dnešní době, kdy má přístup k internetu velké množství obyvatelstva této planety, jsou internetové portály cenným zdrojem informací. V posledních letech zažívají velký rozmach sociální sítě, kde každodenně dochází k přenosu obrovského objemu dat. Mezi nejznámější sociální sítě patří například Facebook, Twitter a Instagram. Zaměřením této práce je získat příspěvky ze sociálních sítí týkající se mobilních operátorů, analyzovat je a následně zobrazit, jaký mají lidé postoj vůči operátorům a službám, které poskytují.

Postoj je pohled, názor, pocit, nebo stanovisko osoby na konkrétní produkt, událost, nebo službu[3]. Obvykle se jedná o subjektivní názor. Ve většině případů se postoj dělí na kladný, záporný a případně neutrální.

Analýza postojů je obor, který se zabývá posouzením lidských názorů a pocitů vyjádřených formou psaného textu [3], [1]. Tento obor je v současnosti aktivně zkoumán, jelikož díky velkému množství blogů, stránek s recenzemi a sociálních sítí je volně dostupný obrovský objem textu, který nese určitý postoj. Jeho zkoumání je důležité z toho důvodu, že postoj má vysoký vliv na lidské chování. Jednotlivé volby, které lidé za svůj život provedou jsou jím často ovlivněny. A to nejenom vlastním. Jak ukazuje tato studie [22], tak většina uživatelů internetu před výběrem produktu zjišťuje názory jiných lidí. To je jedním z důvodů, proč je tento obor důležitý jak pro usnadnění rozhodování např. o koupi určitého výrobku, tak i pro odhalení jeho slabých, či naopak silných stránek. Na základě tohoto odhalení může firma, která analýzu provede učinit rozhodnutí, jež povedou k vyšší, nebo naopak nižší spokojenosti zákazníků. Dalším využitím může být např. systém pro porovnání dvou produktů a jejich vlastností [15], nebo systém pro shrnutí recenzí a názorů zákazníků [27].

Získání výsledného postoje z textu je vzhledem k nejednoznačnosti a komplexnosti *přirozeného jazyka* komplikovaná úloha. Přirozeným jazykem se rozumí jazyk, jenž byl vytvořen a rozvíjen lidmi skrz přirozené používání a komunikaci. Nebyl tudíž vytvořen uměle, jako například programovací jazyky[24]. Na rozdíl od zmiňovaných programovacích jazyků, nebo strukturovaných dat uložených v databázi, je práce s nestrukturovaným textem pro počítač mnohem složitější. Ani v současnosti nejsou programy schopny stoprocentně rozumět ať už syntaxi, nebo sémantice přirozeného jazyka. Díky rozmachu umělé inteligence, strojového učení a práci vědeckých pracovníků však dochází neustále k pokroku v této oblasti.

Tato bakalářská práce se nejprve věnuje teoretickým základům nutným pro návrh a implementaci systému. Postupně prochází jednotlivé kroky, které bylo nutné při tvorbě aplikace podniknout. Ve zkratce se jedná o předzpracování textu, získání příznaků, kategorizaci textu, přiřazení postoje a vizualizaci výsledků. Poslední kapitola se věnuje vyhodnocení přesnosti systému a obsahuje návrhy, jak jej zdokonalit.

Systém, který byl v rámci této práce navržený a následně implementovaný má za úkol nasbírat data, analyzovat je a v neposlední řadě zobrazit takovým způsobem, aby byla výsledná vizualizace jasná a uživatel lehce rozpoznal, jaké je obecné mínění o daném operátorovi a službách jím poskytovaných. Zároveň by měl systém poskytnout i nástroje umožňující operátorovi sledovat, jaká je zpětná vazba zákazníků na jeho služby. Na základě toho by mohl reagovat a svou nabídku upravit.

Právě sběr dat z různých zdrojů, jejich automatická analýza a následná vizualizace je hlavní výhodou oproti existujícím systémům. Pro české operátory není veřejně dostupná žádná stránka, jež by tyto operace prováděla. Aplikace také umožňuje přiřazovat příspěvkům kategorii. Z těchto důvodů je prospěšná jak pro uživatele, jenž se rozhlíží po novém operátorovi, tak pro poskytovatele mobilních služeb.

Uživatel hledající operátora se nejdříve může podívat na stránku s diagramy, kde vidí, jak si vůči sobě operátoři stojí. Také si může prohlédnout, jak jsou hodnoceny jednotlivé služby jimi poskytované. Pro detailnější přehled lze přejít na stránku s příspěvky, jež umožňuje filtrování a nachází se na ní názory z různých zdrojů (z Twitteru, Tarifomatu a do prosince 2017 i z Facebooku). Tyto informace by mu měli ulehčit výběr poskytovatele mobilních služeb.

Zároveň systém poskytuje prostředky pro mobilní operátory, díky nimž mohou sledovat, jak si stojí na trhu v oblíbenosti. V současné době zaměstnanci poskytovatelů mobilních služeb musí procházet jednotlivé sociální sítě odděleně a sledovat na nich příspěvky. Tato aplikace poskytuje zobrazení příspěvků z různých stránek na jednom místě a umožňuje i textové vyhledávání. To dovoluje například sledovat, jak zákaznící reagují na nově spuštěnou službu, ať už u dané společnosti, nebo u konkurence. Zároveň automaticky přiřazuje příspěvkům hodnocení, takže zaměstnanci mohou na první pohled vidět, jestli je příspěvek pozitivní, negativní, nebo neutrální.

Kapitola 2

Teoretický rozbor

Tato kapitola se zaměřuje na teoretické základy a vysvětluje pojmy, které budou později v této práci používány. Také se zde nachází podrobnější popis postoje a jednotlivých kategorií, do nichž se dělí. Dále se zde nalézá rozdělení analýzy postojů dle technik jaké se využívají a rozsahu textu, k němuž se postoj vztahuje. V rámci tohoto rozboru jsou v této kapitole často citovány práce využívající rozebírané techniky.

Větší prostor je věnován aspektově orientované analýze postojů, jelikož ta umožňuje nejenom klasifikovat dokument dle polarity, ale i nalézt jednotlivé aspekty a jim přiřadit hodnocení. Uživatel má tudíž podrobnější přehled o produktu, nebo službě, jež ho zajímá.

Samostatná podkapitola je také vyhrazena analýze postojů na základě dat ze sociálních sítí a to z důvodu, že úzce souvisí s touto bakalářkou prací. Poslední část je vyhrazena překážkám, které mohou při analýze postojů vyvstat.

2.1 Postoj a jeho rozdělení

Jak již bylo řečeno v úvodu, tak postoj je názor, nebo pohled osoby na konkrétní produkt, akci, nebo službu. Lze říci, že je úzce spjatý s emocemi a sám emoce nese. Ty mohou být pozitivní, negativní, nebo neutrální. Toto rozdělení bude dále označováno jako *polarita*. V [24] je postoj definován jako trojice (y, o, i), kde *y* označuje typ postoje, *o* jeho orientaci a *i* intenzitu. Možné typy postoje popsané ve stejném zdroji jsou:

- racionální postoj
- emoční postoj

Racionální postoj vychází z racionálního uvažování. Neměl by tudíž být zastřen emocemi a měl by být založen na logickém uvažování konkrétního jedince [24]. Příkladem tohoto typu postoje je například věta „Ceny tohoto operátora jsou vyšší než u konkurence.“

Emoční postoj vychází z emočního vztahu ke konkrétní entitě a týká se psychologického stavu lidské mysli.[24] Je silnější, než racionální postoj a je snadněji odhalitelný a obvykle i důležitější. Věty obsahující tento druh postoje mohou často obsahovat superlativy (např. nejlepší, nejdražší), zájmena, příslovce, slovesa vyjadřující vztah osoby ke konkrétní věci (např. miluji, nesnáším) a podstatná jména, která jsou emočně zabarvená (např. krám). Příklad věty nesoucí emoční postoj je: „ Tento operátor je nejhorší!“

Orientace postoje je synonymem pro polaritu postoje. Tudíž může být buď *pozitivní*, *negativní*, nebo *neutrální*.

Intenzita postoje záleží na skladbě věty a slovech, která jsou ve větě použita. Příkladem mohou být slova *slušný* a *úžasný*. V tomto případě má druhé slovo pocitově vyšší intenzitu, než slovo první. Intenzita postoje se dále může měnit za pomocí dalších slov, kterými jsou v českém jazyce obvykle příslovce. K těmto slovům patří *velmi, zdaleka, trochu docela, opravdu* a další.

2.2 Analýza postojů

Definice analýzy postojů udává, že se jedná o úlohu patřící do zpracování přirozeného jazyka a dolování textu, jejímž cílem je extrakce, klasifikace a sumarizace postojů a emocí vyjádřených v textu [3].

Základním úkolem analýzy postojů je tudíž rozpoznání emocí a rozpoznání polarity (orientace) textu [6]. Tato práce se zaměřuje na druhý jmenovaný úkol. Tyto dva úkoly spolu úzce souvisí a rozpoznání emocí může být rozšířením v systému pro rozpoznání polarity. Pod analýzu postojů spadá i zjištění, zda je zkoumaný text objektivní, nebo subjektivní a odhalení, k čemu se konkrétní text vztahuje. Úkol, který se zabývá zjištěním *cíle postoje* se nazývá *extrakce aspektů*. Cílem postoje se v tomto smyslu rozumí hodnocený atribut nebo vlastnost.

Metody pro analýzu postojů lze dle [8] rozdělit do několika kategorií. Jedná se o *strojové učení*, *metody založené na slovníku*, *statistické metody* a *přístupy založené na využití pravidel*.

2.2.1 Techniky založené na strojovém učení

Při použití technik založených na strojovém učení je nutností mít korpus s anotovanými daty. Vyhodnocování postojů pak probíhá v závislosti na těchto datech. Tento způsob se označuje jako *strojové učení s učitelem*. Výhodou tohoto postupu je, že systém se může neustále zdokonalovat a s velikostí datasetu se pravděpodobně bude zvyšovat i přesnost výsledků.

Lze využít i techniky při nichž nejsou potřeba manuálně anotovaná data. Ty jsou označeny jako *strojové učení bez učitele*. V tomto případě je ale nutná alespoň lexikální databáze, či slovník nesoucí slova a k nim přiřazené hodnocení (pozitivní nebo negativní). Tyto zdroje pak slouží jako základ, na němž lze strojové učení postavit.

Nedostatkem systémů založených na strojovém učení je, že jsou sémantický slabé. Obvykle je text reprezentován ve formě *bag of words*, což znamená, se berou v potaz četnosti jednotlivých slov, ale ztrácí se informace o vztazích mezi nimi. Mezi techniky, které se pro strojové učení využívají patří *Naive Bayes algoritmus*, *Maximum Entropy* a *Support Vector Machines (SVM)*.

Další nevýhodou je, že při užití manuálně anotovaných dat je výsledná přesnost ovlivněna tím, na jakou doménu dat je klasifikátor použit. Pokud je použit pro data, jež jsou stejného typu jako ta, z nichž se systém učil, tak přesnost obvykle překonává ostatní metody. Ale v případě, kdy chceme klasifikátor použít pro odlišná data, tak přesnost klesá. V doméně, pro níž byl klasifikátor natrénován se přesnost většinou pohybuje mezi 70 - 90%.

Ukázkou metody založené na strojovém učení je *Opinion Digger* [17], který využívá strojového učení bez učitele. Tím pádem odpadá nutnost manuální anotace dat. Na druhou stranu využívá lexikální databáze *WordNet*, což omezuje použití na angličtinu a jazyky, pro které existuje podobná databáze.

2.2.2 Metody založené na slovníku

Tyto metody fungují tak, že celkový postoj se určuje na základě *sémantické orientace* jednotlivých slov. Sémantickou orientací se rozumí míra subjektivity a názoru v textu [8].

Tento přístup je možné kombinovat spolu se strojovým učením, jako to udělal například Adrius Mudinas a spol. s jejich systémem *pSenti* [19]. Výhodou kombinovaného přístupu je, že poskytuje vyšší přesnost, než modely založené pouze na slovníku a oproti technikám založených čistě na strojovém učení poskytuje čitelnější a strukturovanější výsledek díky využití aspektů. Výsledné porovnání také ukázalo, že při použití kombinovaného přístupu je výsledný klasifikátor nezávislejší na doméně na níž je použit, než u technik založených čistě na strojovém učení.

2.2.3 Statistické modely

Statistické modely reprezentují dokumenty jako kombinaci aspektů a hodnocení. Tyto modely jsou vhodné pro použití analýzy postojů u recenzí, jelikož v jedné recenzi se obvykle vyskytuje mnoho různých aspektů a na každý z nich může mít uživatel jiný názor. Příkladem nechť je věta „Pokrytí signálem je dobré, ale internet neustále padá“, kde je prvním aspektem signál (případně pokrytí signálem) a uživatel k němu vyjadřuje pozitivní postoj a naopak k druhému aspektu, což je internet, vyjadřuje postoj negativní. Využívá se toho, že aspekt a slova, která obsahují vůči němu postoj, jsou obvykle velmi blízko sebe - tzv. *shlukování*.

Tuto techniku použili například Samaneh Moghaddam a Martin Ester [18]. Ve výsledku dosáhli 83% při extrakci příznaků a 73% při ohodnocování jednotlivých aspektů.

2.2.4 Přístupy založené na použití pravidel

Tyto metody získávají z textu slova, která jsou nositeli určitého postoje a v závislosti na počtu negativních a pozitivních slov určí výslednou polaritu dokumentu. Při zpracování se využívají slovníky v nichž se nachází slova nesoucí určitý postoj, dále také pravidla pro změnu výsledné polarity při nalezení negace ve větě, pravidla pro superlativy, příslovce, jež umocňují postoj, emotikony atd.

Příkladem práce, kterou lze zařadit do této sekce je [14]. Jejich systém dosahuje přesnosti 91% na úrovni recenze a 81% na úrovni vět. Bohužel v díle chybí porovnání s jinými metodami nad stejným datasetem.

2.2.5 Další způsoby dělení

Dalším způsobem, jak je možné analýzu postojů dělit, je podle úrovně rozsahu textu, k němuž se postoj vztahuje. Takto lze rozlišit *analýzu na úrovní dokumentu*, *analýzu na úrovni vět* a *analýzu na úrovni slov/aspektů*[8]. Volba, jakou z metod použít je závislá na druhu klasifikovaného textu a úrovně detailů, které si uživatel přeje zjistit. O aspektově orientované analýze bude více řečeno v následující kapitole, jelikož je oproti zmíněným typům složitější.

2.3 Aspektově orientovaná analýza postojů

Aspektově orientovaná analýza postojů se nezabývá pouze určením polarity, ale také zjištěním jakého atributu entity se postoj týká. Například ve větě: „Už několik dní je v této

lokalitě špatný signál.“ je entitou operátor, kterého se příspěvek týká, ale konkrétním aspektem může být *signál*, *výpadek signálu*, nebo *výpadek signálu v lokalitě*. Tento příklad ukazuje, že určení aspektu není triviální záležitost a obvykle existuje více možností, jak aspekt zvolit. Při aspektově orientované analýze je jedním z pod-úkolů zjistit výrazy nesoucí aspekt. Ty se dělí na dvě skupiny - výrazy s explicitním aspektem a výrazy s implicitním aspektem [6].

Výrazy s explicitním aspektem jsou podstatná jména, nebo fráze z nich složené. Příkladem může být věta: „Rychlost internetu je velice nízká.“, kde je daným výrazem rychlost internetu.

Výrazy s implicitním aspektem typicky nejsou podstatná jména, nebo fráze z nich tvořené, ale jde o výrazy, které určitý aspekt evokují. Na ukázku je zde uvedena věta: „V Polsku za stejný tarif dají 250 Kč, kdežto my platíme 1200Kč!“ Zde je aspektem cena tarifu, ačkoliv slovo cena není v textu nikde uvedeno.

Pro aspektově orientovanou analýzu lze využít jak metody pro strojové učení s učitelem, tak i bez něj [25]. V citovaném díle je zvolen přístup s učením bez učitele a to z toho důvodu, že pro daný jazyk není v současnosti dostatečné množství anotovaných dat. Z toho vyplývá, že množství anotovaných dat může být jedním ze základních kritérií pro výběr metody pro klasifikaci textu.

2.3.1 Získání aspektů

V současnosti převládají dva postupy, jak získat z dokumentu aspekty [15] [25]. První z nich je založený na získání jmenných frází z dokumentu. To jsou ty fráze, v nichž je hlavou podstatné jméno, nebo substantivní zájmeno [13]. V případě extrakce aspektů jsou důležité obzvláště ty, kde je hlavou podstatné jméno.

Druhou metodou je použití statistického přístupu, jenž využívá toho, že některá slova se často vyskytují spolu a tak tvoří slovní spojení [16]. V obou případech je důležitou součástí procesu přiřazení slovních druhů k jednotlivým slovům v dokumentu. Tento proces bývá v anglické literatuře označován jako „POS tagging“. V případě, kdy jsou slovům v dokumentu přiřazeny slovní druhy, je možné vyřadit ta slova, jejichž slovní druh nemá význam pro určení aspektu a zanechat pouze ta rozhodující. Těmi bývají hlavně podstatná jména. Tohoto přístupu využívá například systém *Sumview*. [27]

Nevýhodou jmenných frází je, že často získají příliš mnoho spojení. Mezi nimi se pak mohou vyskytovat i taková, jež ve skutečnosti nedávají žádný smysl. Naopak použití statistických metod často nezíská všechny použité fráze. Vynechá zejména ty, které se vyskytují jen vzácně, nebo jsou tvořeny jen jedním slovem [15]. Ukázkou přístupu, kdy se aspekt přiřazuje na základě slovníku, je například práce Nadeem Akhtara, Nashez Zubaira, Abhishek Kumara a Tameema Ahmada z roku 2017 [2].

Po získání aspektů lze využít techniky pro „očištění aspektů“. Ty je vhodné použít zejména v případě, kdy byl pro generování aspektů vytvořen automatický systém. Může totiž dojít ke stavu, kdy bude vytvořeno příliš mnoho aspektů, některé z nich budou redundantní atd. Techniky pro čištění aspektů jsou následující: odstranění těch aspektů, jejichž výskyt je nízký, shlukování podobných aspektů do skupin a oříznutí výsledného seznamu aspektů. Tyto metody byly použity v systému *pSenti*, který dosahuje v klasifikaci velmi dobrých výsledků [19]. Metodami pro výběr selekcí jsou *Chi-square*, *Point-wise Mutual Information* a *Latent Semantic Indexing*[1].

Dalším způsobem, jak se zbavit aspektů je *redukce aspektů*. Redukce aspektů je transformační metoda aplikující na data takové transformace, aby je bylo možné zobrazit v prostoru s menším počtem dimenzí [1].

2.4 Analýza postojů na základě dat ze sociálních sítí

Vzhledem k zadání této bakalářské práce se tato sekce věnuje popisu současného stavu na poli analýzy postojů z dat získaných ze sociálních sítí. Sociální sítě jsou výbornými zdroji dat pro analýzu postojů a to především díky oblibě, jakou mezi lidmi mají. Jen na Facebooku je dle statistického portálu statista.com přes 2 miliardy aktivních uživatelů. [26]

Vhodnými sociálními sítěmi pro sběr dat jsou například Twitter, Facebook[1] a Google Plus.

Při zpracování dat ze sociálních sítí je prvním krokem získání oněch dat. To lze většinou pomocí oficiálních nástrojů od provozovatelů dané sítě. Je však nutné podotknout, že společně se zaváděním GDPR se zpřísňuje ochrana osobních údajů uživatelů daných sítí, což vede k tomu, že je legální cestou nemožné získat detailní data o uživateli [2].

2.4.1 Způsoby využití dat ze sociálních sítí

Jedním ze způsobů, jak byla zatím analýza postojů v této oblasti používána je měření deprese uživatelů [11]. Dalším využitím je odhalit potencionální zdroje online radikalizace [4], předpověď výsledků voleb a nepokojů v Egyptě [12], podle spokojenosti turistů s konkrétními destinacemi pak tato místa doporučovat [29] a monitorování názorů studentů na vzdělání [20].

Využití však zdaleka není omezené jen na popsané příklady. Lidé na sociálních sítích často vyjadřují svůj názor vůči konkrétní službě, či produktu ve formě veřejných příspěvků. Ty lze pak využít ke zjištění obecného názoru na tuto věc.

2.5 Potíže při zpracování jazyka a analýze postojů

Tato podkapitola se věnuje problémům a překážkám, které nastávají ať už konkrétně při analýze postojů, nebo při zpracování přirozeného jazyka.

Už ze své podstaty je analýza přirozeného jazyka netriviální problém, jelikož počítače nemají znalosti okolního světa, jež lidé běžně využívají při komunikaci. Při obvyklé komunikace se většinou počítá s tím, že je znám kontext, v rámci něhož byla věta vyřčena, či napsána. Například text „Nestojí to za to.“ je silně závislý na místě, kde se využije. Pokud je tato věta reakcí v diskuzi o tarifu, tak se vztahuje k jinému tématu, než když je použit v diskuzi o vysoké škole.

Nejednoznačnost přirozeného jazyka je problém, který zatím není v metodách pro zpracování jazyka stoprocentně vyřešen. Jedná se o pochopitelnou věc, jelikož i pro člověka může být občas pochopení věty složité. Uvedu zde větu: „Děti našly houby.“ Bez znalostí o okolním světě není z věty zřejmé, kdo koho našel. Další nejasností je, zda děti opravdu našly houby, nebo nenašly nic.

Dalším z úkolů, který ztěžuje zpracování přirozeného jazyka a blíže se dotýká analýzy postojů, je *rozpoznání ironie a sarkasmu*[30]. Ten totiž může kompletně změnit polaritu příspěvku. Příkladem budiž věta: „To se vám zas povedlo. S každou změnou jsou vaše služby „lepší“ a „lepší“.“ Přestože jsou ve větě 3 výrazy nesoucí kladný postoj - *povedlo* a dvakrát *lepší*, celkový postoj je díky využití sarkasmu negativní.

[1] Od 26. 3. 2018 došlo k omezení možností pro získání dat z Facebooku kvůli ochraně osobních údajů uživatelů. Toto mimo jiné způsobilo, že přes GraphAPI už nejsou volně dostupné informace týkající se uživatelů Facebooku.

[2] Tento problém se týká především Facebooku po kauze s únikem osobních dat a jejich využíváním firmou Cambridge Analytica

Mezi další překážky, které stěžují práci, patří *rétorické otázky*. Přestože nemusí nést výrazy, dle nichž by se zdálo, že nějaký postoj nesou, opak je pravdou. Věta „Proč se já vždy nechám přemluvit k prodloužení smlouvy?“ neobsahuje slova nesoucí postoj, přesto však vyznívá negativně.

Výčet zde jmenovaných problémů rozhodně není kompletní, ale to také není smyslem této práce. Zde ukázané překážky a příklady jsou jmenované pro to, že se vyskytovali v příspěvcích, které se na sociálních sítích vyskytují běžně. Některé další záležitosti, s nimiž je nutné se při analýze postojů vypořádat, jsou jmenovány v další kapitole. Jedná se především o věci vztahující se k úpravě textu příspěvků. Jmenovitě lemmatizace, odstranění stop slov, odstranění diakritiky a další.

Kapitola 3

Struktura systému

Obsahem této kapitoly je objasnit strukturu aplikace a systematicky popsat jednotlivé akce, které aplikace vykonává. Cílem této kapitoly není vylíčit samotnou implementaci systému, ale popsat jednotlivé součásti systému a technologie, na nichž aplikace staví.

Zahájena je popisem stahování dat a tvorbou korpusu, jejich předzpracováním a anotováním, extrakcí příznaků, indexací a vizualizací výsledných dat. Také popisuje metody používané pro ohodnocení kvality klasifikátoru. V této části se nepopisuje analýza postojů, jelikož té byla věnována téměř celá předchozí kapitola.

3.1 Stahování dat a tvorba korpusu

Aby bylo možné vytvořit systém pro analýzu postojů za použití strojového učení s učitelem, je nutné mít získána a anotována potřebná data. Tato podkapitola se věnuje získání dat. V anglické literatuře je automatizované systematické získání obsahu webu často označováno jako *data scraping*. Automatizované stahování probíhá tak, že robot chovající se jako člověk prochází přes webové stránky a získává a zpracovává jejich obsah. Tento proces má několik cílů [10].

3.1.1 Komunikace se stránkou

Prvním z cílů je zprostředkovat komunikaci se stránkou z níž se budou získávat data. Obvykle se k tomuto účelu využívá komunikace přes bezstavový HTTP protokol. Nejčastěji se při komunikaci využívají HTTP metody *GET* a *POST*. Metoda *GET* se typicky používá k získání dat ze serveru, zatímco metoda *POST* k odeslání dat. Cílem této práce není detailní a kompletní rozbor HTTP metod, tudíž se zde nenaláze popis ostatních metod, ani podrobnější porovnání dvou zmíněných.

Jak Facebook, tak Twitter poskytují oficiální API pro získávání dat, tudíž určitým způsobem zapouzdřují manuální volání těchto metod, což vede k vyššímu stupni abstrakce a jednodušší práci pro programátory.

3.1.2 Zpracování HTML a extrakce obsahu

Po navázání komunikace je dalším krokem samotné zpracování a extrakce obsahu. V případě kdy zdroj dat poskytuje oficiální API, tak je obsah již získán ve formě JSON, nebo XML. V tomto případě stačí vyfiltrovat pouze významná data a nejsou potřeba provádět žádné další úpravy.

V případě, kdy není veřejné API k dispozici, je obsah obvykle získán ve formě HTML dokumentu. Ten je nutné očistit od HTML značek a získat z něj jen ta data, jež jsou pro systém relevantní. Je více možností, jak získat požadovaný obsah. Ke zbavení se HTML značek jsou hojně využívány regulární výrazy.

3.1.3 Budování výstupu

Po získání dat je dalším krokem jejich převod z nestrukturované podoby do strukturovaného tvaru vhodného pro jejich uložení. Jedním z nejpoužívanějších způsobů uložení dat je uložení do *relační databáze*. Dalšími variantami může být *NoSql databáze*, *objektová databáze*, či prosté uložení do souboru ve formátu *XML*, nebo *JSON*.

Takto stažená data tvoří *korpus*. Korpus se rozumí všechny stažené *dokumenty*. Dokumenty jsou v tomto případě příspěvky ze sociálních sítí. Jeden příspěvek se rovná jednomu dokumentu.

3.2 Uložení a formát dat

Data stažená z internetu je potřeba v jejich strukturované formě uložit. Jedním z rozhodnutí při ukládání dat je volba formátu. Velké množství dat, jež jsou přenášena mezi serverem a klientem je ve formátu JSON, nebo XML. Jedná se i serializovaná data, která je možné uložit přímo v této podobě, nebo jednoduše převést na relace. V současnosti je preferovaným způsobem JSON, jelikož je formát uložených dat jednodušší a čitelnější a zároveň se velmi podobá JavaScriptovému objektu.

Při ukládání dat je důležitým rozhodnutím vybrat informace, které budou ukládány. Tato práce vyvozuje údaje, které je potřeba získat z definice názoru v [6], kde se uvádí, že názor je definován jako čtveřice (e, a, s, h, t), kde *e* je *cílová entita*, *a* je *aspekt* na dané entitě, *s* je *postoj* hodnotitele na aspekt *a* u entity *e*, *h* je držitel daného názoru a *t* je datum a čas, kdy názor vznikl.

Význam a definice aspektu a postoje jsou již popsány výše. Entita je v rámci této bakalářské práce vždy mobilní operátor. Držitel daného názoru je uživatel, který přidal příspěvek na stránku operátora. Tím, že je uložen společně s příspěvkem je možné zjistit, co všechno daný uživatel publikoval, jaký je jeho obvyklý postoj ať už na konkrétní službu, nebo obecně. Obecným postojem se rozumí jakým způsobem obvykle hodnotí a jestli u něj převládají negativní, nebo pozitivní recenze.

Každý člověk má totiž odlišný způsob vnímání světa a to, co je pro jednoho pozitivní, může být pro jiného negativní. Příkladem budiž výsledky prezidentských voleb.

Tato práce se tvorbou uživatelských profilů nezabývá, jak z důvodu omezeného množství informací, jež je možno o uživatelích sociálních sítích získat, tak z důvodu časové náročnosti. Jedno z možných rozšíření by však mohlo být zavést tyto uživatelské profily a pak například hodnocení příspěvku upravovat v závislosti na tom, jaký uživatel příspěvek publikoval a jaký je obvykle jeho názor v porovnání s ostatními uživateli.

Posledním prvkem u dříve zmíněné definice názoru je datum a čas. Jeho význam tkví v tom, že umožňuje zjišťovat, jak se v průběhu doby mění postoj vůči konkrétnímu aspektu, nebo entitě. Tyto data pak mohou být základem pro vykreslení grafu, v němž je přehledně zobrazeno, jak se názor v průběhu doby mění. To umožňuje zákazníkovi vidět na první pohled přednosti jednoho produktu oproti druhém. Poskytovatel služeb/produktů má naopak možnost vidět vývoj spokojenosti zákazníků a v případě poklesu na danou situaci zareagovat.

3.3 Předzpracování dat

Předzpracování dat je klíčová část systému, která významně ovlivňuje výslednou efektivitu. Tato pasáž je prováděna před extrakcí příznaků a ovlivňuje tudíž jak přiřazení aspektu, tak přiřazení výsledného postoje. Příznaky se získají obvykle převodem textu do podoby vektoru.

Mezi metody pro předzpracování textu patří následující:

- Tokenizace
- Odstranění speciálních symbolů
- Odstranění stop slov
- Odstranění diakritiky
- Stematizace
- Lemmatizace
- Oprava chyb
- Označení slovních druhů

3.3.1 Tokenizace

Tokenizace je proces rozdělení textu do samostatných jednotek označovaných jako tokeny. Tokeny jsou nezávislé a minimální textové komponenty, které mají jasnou syntaxi a sémantiku [24]. Tokenizaci lze dělit na *tokenizaci na úrovni vět* a *tokenizaci na úrovni slov*. První případ rozděluje dokument na věty. V jazyce, jako je například čeština a angličtina je rozdělení jednodušší, protože k oddělení vět se používají jasně definovaná interpunkční znaménka. Problém však může nastat v případě použití zkratek ve větě. Věta u níž se tato obtíž vyskytuje je např. „Na nám. Karla IV se dnes konají trhy.“

V případě analýzy postojů však tokenizace na věty není příliš důležitá, tudíž nevadí, když dojde ke špatnému rozdělení u minimálního vzorku dat. Toto tvrzení neplatí v případě, kdy se dělá analýza postojů na úrovni vět. Pak je nutné věnovat rozdělení do vět větší pozornost. Běžně je však tokenizace na věty používána jen jako mezikrok mezi holým textem a tokenizací na slova.

Pro tokenizaci na úrovní slov se k rozdělení obvykle používají bílé znaky. Tento proces je velmi důležitý, jelikož další operaci, jako je například lemmatizace a stematizace se provádějí právě nad slovy.

3.3.2 Normalizace textu

Normalizace textu je proce skládající se ze série kroků, která slouží ke standardizaci textových jednotek tak, aby mohla být předána analytickým systémům a systémům pro zpracování textu. Normalizace výrazně ovlivňuje výslednou přesnost systému [24].

Odstranění speciálních symbolů

Speciálními symboly se v tomto kontextu rozumí hlavně interpunkční znaménka. Tento úkon se provádí z důvodu, že pro sémantickou analýzu nemají přílišný význam. Výjimkou je znaménko **!**, jež bývá často využíváno pro zesílení názoru.

V případě příspěvků ze sociálních sítí se v psaném textu také často nachází emotikony. Otázkou je, zda je lepší je odstranit, nebo zanechat. Na první pohled se může zdát, že by jednoznačně měly tyto symboly při analýze postojů pomoci. Problém však je, že jsou často využívána ironicky. Ideálním řešením by pravděpodobně bylo nenastavovat postoj podle emotikon, ale mít pro ně speciální pravidla, pomocí nichž by došlo k ovlivnění přiřazeného hodnocení dokumentu až po přiřazení hodnocení na základě textu. V rámci této práce se emotikony neberou v potaz. Mohlo by se jednat o rozšíření systému.

Konverze velikosti písmen

Konverze velikosti písmen umožňuje snazší práci v dalším zpracování textu. Nestane se například, že je některé slovo počítáno jako dvě rozdílná jen pro to, že jednou je napsáno s prvním písmenem velkým a podruhé ne.

Odstranění diakritiky

Tento krok se netýká všech jazyků, ale v češtině je nezbytný z důvodu, že ne všichni uživatelé sociálních sítí používají při psaní příspěvků diakritiku. Tento prvek může následně ztížit lemmatizaci, jelikož poté není jasné, zda lemma od slova *sane* jsou sáně, nebo saň. I za tuto cenu je však tento krok normalizace potřebný.

Opakem tohoto přístupu by bylo přidání diakritiku, ale opět nastává problém s nejednoznačností.

Dalším řešením je využit dva rozdílné lemmatizátory. Jeden, co pracuje s diakritikou a druhý, jenž pracuje bez ní. Toto řešení neodstraní problém z nejednoznačností úplně, ale dokáže ho omezit. Nevýhodou je zvýšená paměťová náročnost systému a nutnost se pro každý dokument rozhodnout, jaký z lemmatizátorů použít.

Odstranění stop slov

Stop slova jsou slova, která nemají pro analýzu žádný význam, nebo je mizivý. Většinou se jedná o slova vyskytují se v textu nejčastěji. Příkladem jsou slova *a, i, protože, je* atd.

Pro češtinu lze najít na internetu několik seznamů stop slov, kde většina obsahuje stejná slova. O žádném z těchto seznamů však nejde tvrdit, že je kompletní, jelikož pro konkrétní doménu mohou být některá slova z těchto seznamů významné a jiná slova, jež nejsou důležitá, v nich naopak mohou chybět. Z tohoto důvodu je dobré upravit seznam stop slov v závislosti doméně, pro níž bude užit.

Odstranění stop slov má hlavně význam pro následnou extrakci příznaků. Tento krok redukuje problém, kdy by v dokumentu byla nejdůležitějšími slovy slova, jež nenesou žádný postoj, ale vyskytují se v textu nejčastěji.

Stematizace a lemmatizace

Stematizace má za cíl určit pro jednotlivá slova jejich *stem*. *Stem* lze popsat jako tvar slova, z nějž jsou odstraněny předpony a přípony. Ve skutečnosti nemusí být tento tvar existující slovo.

Lze tedy říci, že cílem stematizace je naleznout pro všechna příbuzná slova jeden společný tvar. Tento tvar se nemusí shodovat s morfologickým kořenem slova [7]. Toho se snaží dosáhnout až *lemmatizace*. Stematizaci lze využít pro jazyky, kde jsou slova tvořena na základě určité množiny pravidel. Mezi tyto jazyky čeština patří.

Pro stematizaci i lemmatizace lze využít několik různých algoritmů. Ty nejzákladnější jsou založeny na existencí slovníků, v nichž se nachází dvojice slov - kořen slova. V tomto slovníku se pak vyhledává hledaný výraz. Tento přístup má tu nevýhodu, že je nutné mít ve slovníku zapsány všechny možné tvary slov a k tomu jejich kořen. Tento způsob je běžně označován jako *Brute Force algoritmus*.

Další používané způsoby jsou *Suffix stripping algoritmy*, *stochastické algoritmy* a *hybridní algoritmy* [7].

Suffix stripping algoritmy využívají množinu pravidel, na jejímž základě se snaží získat základní tvar slova. Oproti Brute Force algoritmům nepotřebují slovník všech slov. Jejich problémem je, že si dost dobře nedokáží poradit s výjimkami, jež konkrétní jazyk může obsahovat.

Stochastické algoritmy určují kořen slova na základě pravděpodobnosti. Nejprve je nutné vytvořit pravděpodobnostní model zachycující vztahy mezi slovy a jejich kořeny. Takto vytvořeným modelem bývá soubor pravidel podobný tomu u *Suffix stripping algoritmů* [7]. Po vytvoření tohoto modelu se mu na vstup předá slovo, k němuž se hledá kořen a model na výstupu vyprodukuje nejpravděpodobnější výsledek.

Hybridní metody pak kombinují několik výše zmíněných přístupů.

Lemmatizace bývá přesnější, než stematizace. Na druhou stranu je však pro systém náročnější. Jak z hlediska času, tak z hlediska spotřeby zdrojů. Studie [3] ukazuje, že většina současných systémů pro analýzu postojů využívá lemmatizace právě z důvodu její přesnosti.

Oprava chyb

Oprava chyb je netriviální záležitost. Jejím smyslem je získat pravopisně správný tvar slova. To je důležité, protože zatímco slovo *dobrý* nese kladný postoj, tak slovo *dobrz* žádný nenese. Dalším problémem je, že uměle zvyšují velikost *lexikonu slov*. Termín lexikon využívám ve významu seznam unikátních slov.

Při odstraňování chyb často existuje několik různých možností, jak slovo opravit. Tudíž vybraná varianta nemusí být vždy správná. Jednou ze strategií jaké slovo vybrat je použít to slovo, jenž se v korpusu vyskytuje nejčastěji.

Dalším způsobem, jak se rozhodnout při výběru slova, jímž se má nahradit chybný výraz, je rozhodnutí na základě tvaru věty a jednotlivých slovních druhů. V tomto případě je nutné mít označené slova slovními druhy a soubor pravidel, které udávají jak se věty v daném jazyku staví. Tento způsob je však pro češtinu velice obtížný, jelikož český slovosled je značně volný a pružný.

Jednodušší variantou, než je oprava chyb, je odstranění nerozpoznaných slov. Tímto krokem se sice daný výraz ztratí, takže nebude mít vliv na výsledné hodnocení postoje, ale alespoň nedojde ke zvyšování velikosti lexikonu. Je však nutné se ujistit, že základna slov, na jejímž základě se kontroluje, zda je slovo existující, je dostatečně obsáhlá. Pokud by nebyla, docházelo by k odstraňování slov, která jsou existující. Dalším problémem jsou doménově specifické zkratky. Ty mohou být důležité ke zjištění aspektů, tudíž by nemělo dojít k jejich odstranění. Příkladem budiž *LTE, O2TV, 4G*,...

3.3.3 Označení slovních druhů

Označení slovních druhů je proces, jenž bývá v anglické literatuře označován jako *POS tagging*. Jeho cílem je přiřadit jednotlivým slovům jejich slovní druhy. Bývá častou součástí systémů pro analýzu postojů, kde se využívá jak pro získání aspektů (na základě jmenných frází), tak pro získání slov nesoucích postoj (těmito slovy bývají obvykle přídavná jména).

POS tagerry bývají často založené na Markovských modelech (*Markov models*) [5], či na modelu maximální entropie(*Maximum Entropy model*)[23].

3.4 Extrakce příznaků

Extrakce příznaků je jedním ze základních a velmi důležitých kroků při tvorbě systému pro analýzu postojů. Slouží k převodu dokumentu z textové reprezentace do formy vektoru. Důvodem je, že algoritmy pro klasifikaci dokumentů využívají matematické funkce, které pracují čísly, ne s řetězci.

Při extrakci příznaků je možné zvolit, jakým způsobem dojde k převodu řetězce na slovo. Základní reprezentací je *BOW model*.

3.4.1 BOW model

BOW model je jednoduchý a flexibilní přístup pro extrakci příznaků z textu. Tento model popisuje počet výskytů slov v dokumentu. Jeho omezením je, že se při použití ztrácí vztah mezi slovy. Například věta „Nemám rád tohoto operátora.“ bude převedena do tvaru „nemám“, „rád“, „tohoto“, „operátora“. Tím pádem se ztratí spojení mezi slovy „nemám“ a „rád“, což může vést k chybě při klasifikaci.

Tuto nepříjemnost je možné omezit zavedením n-gramů. To znamená, že se nebudou počítat četnosti použití jednotlivých slov, ale četnosti dvojice (bigramy), nebo n-tice slov (n-gramy). Možná překvapivě práce Panga a spol. ukazuje, že lepších výsledků se při klasifikaci dosahuje s použitím unigramů [21].

Opačné výsledky prezentuje práce Hang Cui a spol. [9], jejichž systém se zpřesňuje s využitím n-gramů vyšších stupňů. Hlavním rozdílem mezi těmito studiemi je velikost datasetu. Cui a spol. využili při své práci rozsáhlejší dataset, než Pang. Z jejich práce tedy vyplývá, že volba typu n-gramů při klasifikaci by měla být závislá na velikosti datasetu.

Dalším problémem, který se při BOW reprezentaci vyskytuje je, že slova objevující se v textu častěji mají vyšší váhu, i když je jejich přínos minimální. Jedná se např. o zájmena, spojky a slovesa.

Existuje několik postupů, jak se s tímto vypořádat. Jedním ze způsobů, jak omezit tento problém je již výše popsané odstranění stop slov. Za pomocí POS taggingu také lze přiřadit slovům slovní druhy a slova s určitým slovním druhem odstranit. Třetí způsob, který se snaží tento problém redukovat je zavedení principu, kdy váha slova nezávisí jenom na počtu jeho výskytů dokumentu, ale i na počtu výskytů daného slova v korpusu. Tato metoda je označována jako *TF-IDF*.

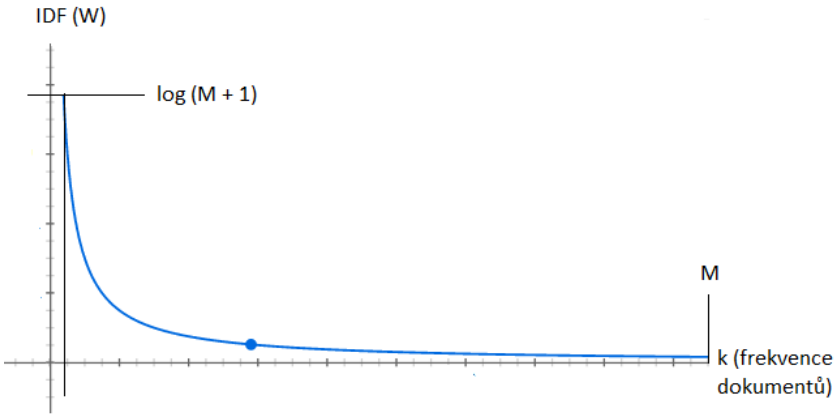
3.4.2 TF-IDF

TF v názvu metody označuje frekvenci termů(*term frequency*). Frekvence termů označuje, jak často se konkrétní slovo v dokumentu nachází. IDF označuje inverzní frekvenci termů (*inversion document frequency*).

Inverzní frekvence termů slouží k tomu, aby byly penalizovány ty výrazy, jež se v korpusu nachází nejčastěji. Pro výpočet inverzní frekvence termů lze využít následující rovnici:

$$IDF(W) = log[M + 1/k],$$

kde *IDF(W)* značí funkci inverzní frekvence termů pro slovo *W*, *M* značí celkový počet dokumentů v kolekci a *k* označuje celkový počet dokumentů v kolekci obsahujících slovo *W*. Následující graf zobrazuje charakteristiku této funkce 3.1.

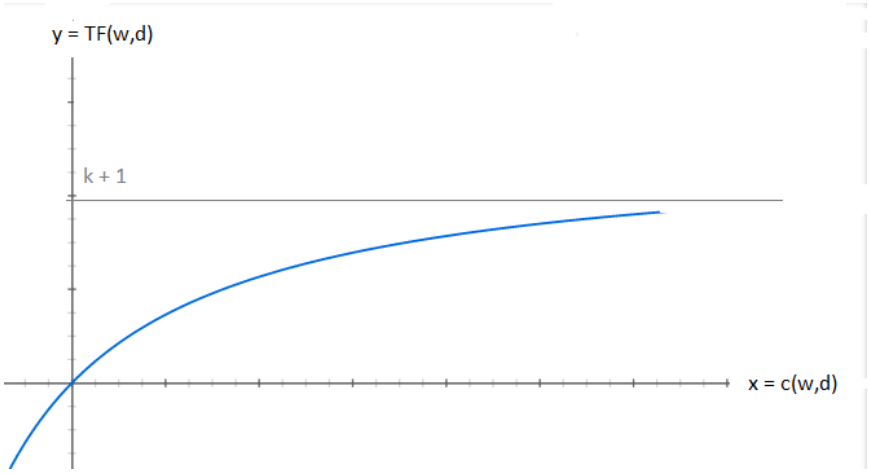


Obrázek 3.1: IDF funkce

Při použití této formule dojde k penalizaci nejběžnějších slov. Problém však nastane, když se slovo, které má vysokou hodnotu inverzní funkce nachází v jednom dokumentu mnohokrát. Aby došlo k omezení jeho vlivu, tak lze využít následující funkci:

$$TF(w,d) = \frac{(k+1)c(w,d)}{c(w,d)+k},$$

kde TF(w,d) označuje funkci pro frekvenci termů pro slovo w v dokumentu d, k je celkový počet dokumentů v kolekci obsahující slovo w a $c(w, d)$ označuje počet slov w v dokumentu d. Charakteristiky této funkce zobrazuje graf 3.2.



Obrázek 3.2: IDF funkce

3.5 Indexace dat

Indexace dat má za úkol vytvořit pro kolekci dokumentů takové datové struktury, které umožňují rychlé vyhledávání. Databázový systém obvykle takovou strukturu obsahuje a ta umožňuje vyhledávat data bez toho, aniž by musely být procházeny všechny řádky konkrétní tabulky, což může znatelně zvýšit rychlost a efektivitu. Pro indexaci dat se využívají např. následující techniky [28]:

- Sufixové stromy
- Invertovaný index
- Ngram index

Tato práce nemá za cíl popsat a rozebrat všechny indexační techniky, takže je zde popsán pouze *invertovaný index*, jenž patří k těm nejpoužívanějším.

3.5.1 Invertovaný index

Invertovaný index se používá k mapování slov na jejích pozici v dokumentu. Jednoduše si ho lze představit jako rejstřík v knize, kde je k jednotlivým pojmům přiřazena informace o pozici (stránce), na níž se nachází.

Invertovaný index se skládá ze dvou hlavních částí: vyhledávací struktury, nebo slovníku obsahujícího všechny hledané hodnoty a invertovaného indexu pro každou z odlišných hodnot [31]. Pro ukládání invertovaného indexu lze využít datové struktury typu pole, B+ stromy a hashovací tabulky. Do invertovaného indexu lze také ukládat četnosti, kolikrát se hledaný výraz v dokumentu nachází.

Další důležitou charakteristikou invertovaných indexů je, že to jsou řazené sekvence identifikátorů jednotlivých záznamů. To umožňuje rychlé vyhledávání a případnou kompresi záznamů. Jednoduchý nákres invertovaného indexu je na obrázku 3.3.

3.6 Vizualizace dat

Vizualizace dat je proces zkoumání dat a informací a jejich převedení do grafické podoby. Typicky se skládá ze získání, zpracování a zobrazení dat. Z pohledu člověka následuje ještě proces vnímání, ukládání zobrazených dat do paměti a rozpoznání významu dat.

Vizualizace lze dělit na *interaktivní vizualizaci*, *prezentační vizualizaci* a *interaktivní storytelling*.

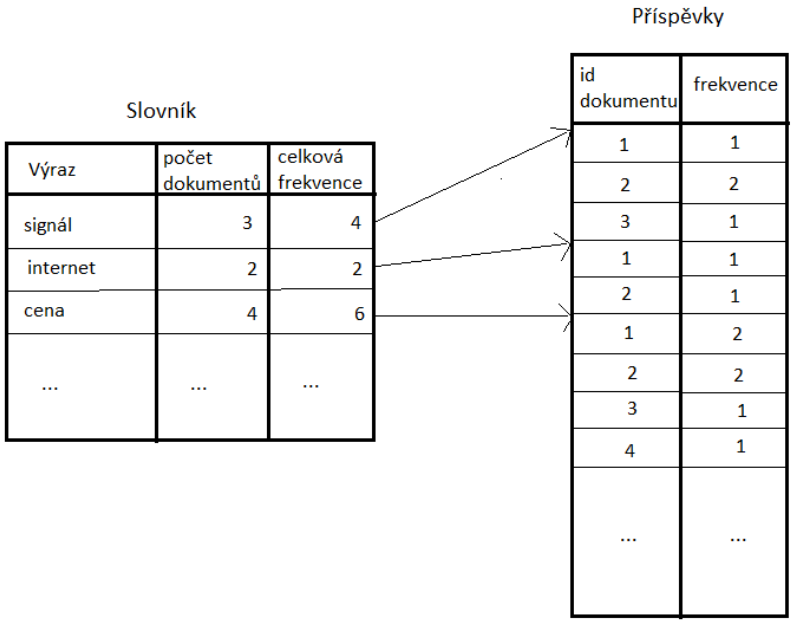
Interaktivní vizualizace umožňuje uživateli zadat vstup na jehož základě dojde k překreslení zobrazených dat. Je vhodná pro zkoumání závislostí mezi daty.

Prezentační vizualizace neumožňuje zadat uživatelský vstup. Často je zaměřeno na publikum a slouží pro převážně pro prezentaci výsledků.

Interaktivní storytelling je kompromisem mezi zmíněnými přístupy. Jedná se typicky o web stránku, kde je uživateli umožněno měnit data v určitém měřítku. Tento způsob vizualizace je pro systém implementovaný v rámci této bakalářské práce ideální.

3.7 Metriky pro vyhodnocení přesnosti systému

Důležitou částí při tvorbě systému je vyhodnocení úspěšnosti jeho klasifikace. Na základě této informace je pak možné se rozhodnout, zda by měl být klasifikátor použit, nahra-



Obrázek 3.3: Invertovaný index

zen, nebo alespoň upraven. Vyhodnocení úspěšnosti je založeno na tom, jak dobře systém odhaduje třídy (labels), které jsou přiřazeny jednotlivým dokumentům.

Mezi základní metriky patří [24]:

- Přesnost (Accuracy)
- Preciznost (Precision)
- Návratovost (Recall)
- F1 skóre (F1 score)

Tyto metriky se provádějí pro každou ze tříd, do kterých může dokument patřit. Pro jednotlivé třídy lze sestavit tzv. *confusion matrix* 3.1.

Tabulka 3.1: Confusion matrix

	P (předpovídané)	N (předpovídané)
P (aktuální)	True Positive (TP)	False Negative (FN)
N (aktuální)	False Positive (FP)	True Negative (TN)

Tato tabulka zobrazuje nejjednodušší variantu, kdy je na výběr mezi dvěma třídami. P značí pozitivní třídu, N značí negativní třídu. *True Positive* značí počet správných predikcí pro pozitivní třídu, *True Negative* značí počet správných odhadů pro negativní třídu, *False Negative* značí kolikrát klasifikátor špatně přiřadil dokumentu negativní třídu, i když měla být pozitivní a *False Positive* značí opak. Tzn. kolikrát byla dokumentu přiřazena pozitivní třída, přestože správně měla být negativní.

Přesnost značí kolik dokumentů bylo z celkového počtu přiřazeno správně. Rovnice je následující:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

Preciznost udává kolik odhadů je ve skutečnosti správných pouze vzhledem k pozitivní třídě. Další označení pro tuto metriku je *positive predicted value*[24].

$$Precision = \frac{TP}{TP + FP}$$

Návratovost udává, kolik instancí pozitivní třídy bylo správně zařazeno vůči celkovému počtu instancí pozitivní třídy.

$$Recall = \frac{TP}{TP + FN}$$

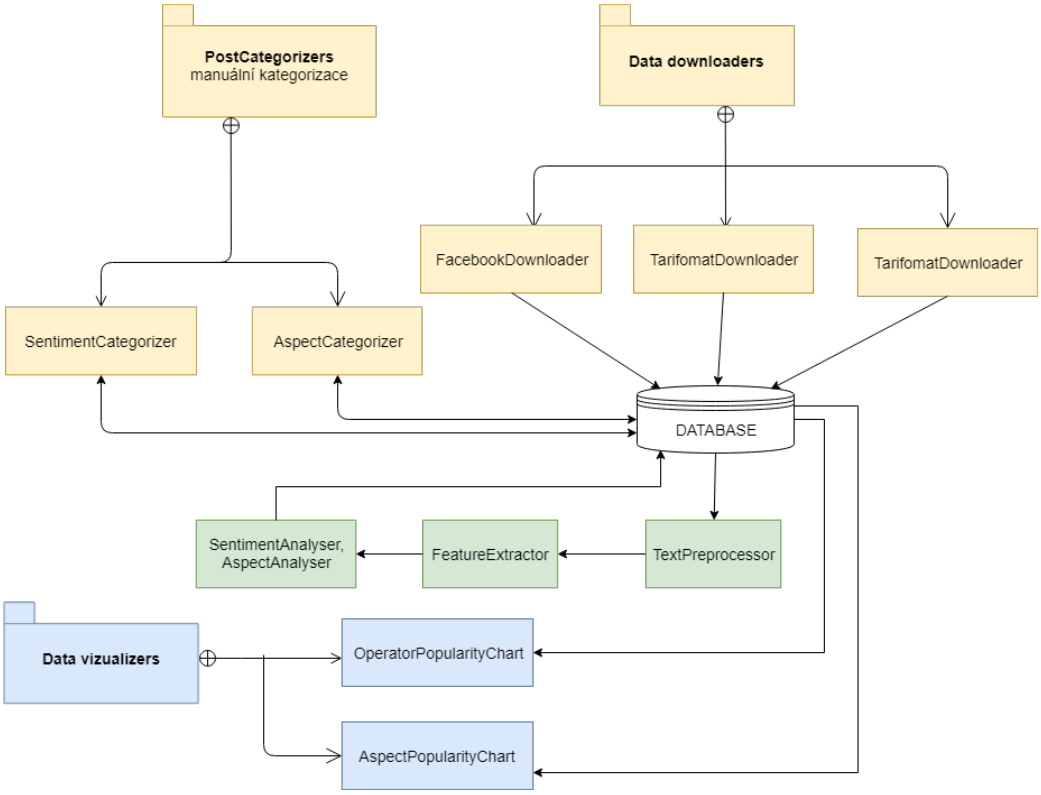
F1 skóre je spočítáno jako *harmonický průměr* přesnosti a návratovosti a je počítán následovně:

$$F1_score = \frac{2 * Precission * Recall}{Precision + Recall}$$

Kapitola 4

Implementace systému

Tato kapitola se zabývá samotnou implementací systému. Nejedná se ani tak o podrobný popis jednotlivých částí, jako spíše o zevrubný pohled na implementovaný systém s popisem jednotlivých součástí a jejich propojení. V úvodu kapitoly je obecné schéma systému, v němž jsou zachyceny komponenty, z nichž se skládá 4.1. Jednotlivé komponenty budou v rámci této kapitoly popsány v takovém pořadí, v jakém v programu pracují. Důležité, či složitější komponenty budou mít v podkapitole vlastní schéma odhalující vnitřní implementaci dané komponenty.



Obrázek 4.1: Základní komponenty systému

4.1 Stahování dat

Zdrojem dat pro implementovaný systém je Facebook[1], Twitter a Tarifomat. Každému ze zdrojů je věnována samostatná podkapitola vzhledem k naprosto rozdílnému způsobu stahování dat.

Aby nedocházelo ke stažení již uložených příspěvků, tak systém obsahuje v databázi tabulku *sentimentAnalyser_lastspostsdownloaded* v níž se pro Facebook, Tarifomat a operátory, k nimž se stahují data, nachází informace o posledním staženém příspěvku. Přesně se jedná o jeho čas vzniku. U stahování dat z Twitteru si stačí zapamatovat id posledního staženého příspěvku, jelikož novější příspěvek bude mít vždy id vyšší. Toto id je uloženo v databázi v relaci *sentimentAnalyser_lastpostsdownloadedtwitter*.

4.1.1 Stahování dat z Facebooku

Facebook byl zvolen z důvodu, že se na něm vyskytuje největší množství příspěvků týkajících se českých operátorů a loňským rokem ještě platilo, že jsou data jednoduše získatelná. Facebook poskytuje oficiální API (*Graph API*) poskytující přístup k datům. Aby byla data přístupná, je nutné si u Facebooku zaregistrovat aplikaci a tu pak využít jako vstupní bod do vývojářské platformy Facebooku.

Graph API je struktura typu graf, která sdružuje informace o uživatelích, stránkách, akcích atd. Tato struktura je složena z uzlů, hran a polí.

Uzly se rozumí konkrétní entity, jako je například stránka, uživatel, fotografie, komentář atd. Každý uzel má unikátní id, čehož se využívá pro přístup k němu.

Hranou je propojení mezi jednotlivými uzly. Příkladem může být komentář k fotografii.

Polemi jsou informace ke konkrétnímu elementu. Tím mohou být například narozeniny u konkrétní osoby. Jednotlivé požadavky se posílají pomocí HTTP požadavku typu GET. Data, jež Facebook vrací jsou ve formátu JSON.

```
GET graph.facebook.com /{node-id}/{edge-name}
```

Aby bylo možné Graph API používat, je nutné mít vygenerovaný **přístupový token**, jenž je pro každou aplikaci unikátní. Facebook poskytuje několik druhů přístupových tokenů. Jedná se o *uživatelský přístupový token*, *aplikační přístupový token* a *přístupový token stránky*.

Uživatelský přístupový token je potřeba, když GraphApi požaduje čtení, zápis, nebo modifikaci uživatelských dat na přání uživatele. Je získán na základě přihlašovacího dialogu a vyžaduje uživatelské potvrzení, aby byl přidělen. Jeho platnost je 2 hodiny. Díky těmto zmíněným vlastnostem není pro vhodný pro použití v systému implementovaném v rámci této bakalářské práce.

Aplikační přístupový token se používá k nastavení a modifikaci aplikace. Také může být využíván k odběru dat z Graph API. Platnost tohoto tokenu je neomezená. Vzhledem k těmto vlastnostem je nejvhodnější pro použití. Jeho omezením je, že pomocí něj lze přistupovat pouze k omezenému množství dat. V březnu 2018 Facebook zrušil přístup k veškerým datům týkajících se uživatelů za pomocí tohoto tokenu. Z příspěvků již tedy nelze získat informace o člověku, který jej publikoval.

Přístupový token stránky slouží pro čtení, zápis a modifikaci dat na určité stránce. Tudíž je pro aplikaci nevhodný.

[1] Do března 2018

Ke stahování dat z Facebooku slouží třída *FacebookDownloader*, jež umožňuje asynchronně stahovat data ze stránek různých operátorů v jeden okamžik. Data stahuje na základě seznamu id stránek mobilních operátorů. Ze stránek se stahují jen příspěvky publikované uživateli. Příspěvky vytvořené mobilním operátorem obvykle nejsou pro analýzu postojů podstatné, jelikož se v naprosté většině případů jedná o propagaci, nebo soutěže. Stejně tak se neukládají komentáře k jednotlivým příspěvkům, protože se téměř vždy jedná o odpovědi od operátora na otázku, nebo stížnost. Po stažení jsou data transformována do formátu v němž jsou ukládána do databáze a odeslána na server.

Stahování dat z Facebooku se provádí na základě kliknutí na tlačítka pro stahování dat z Facebooku.

Omezení stahování dat z Facebooku

Na konci března došlo k dočasnému omezení neaktivních aplikací na Facebooku. Za neaktivní aplikace Facebook považuje takové, které nejsou přístupně uživatelům a jsou Facebookem neschválené [2]. Toto se dotklo i tohoto systému, tudíž je stahování z Facebooku v současnosti nefunkční. Tento stav by měl odeznít po tom, co Facebook vyřeší problémy se zabezpečením dat uživatelů.

4.1.2 Stahování dat z Twitteru

Dalším významným zdrojem dat pro aplikaci je Twitter. Ke stahování je použita knihovna *Tweepy* napsaná pro jazyk Python. Na serveru, kde je aplikace nasazena, je nastaveno každodenní stahování dat. Data se stahují v 1:30 ráno. Ve 2:00 poté dochází ke klasifikaci postojů. Aby bylo možné stahovat data z Twitteru, je nutné mít u něj zaregistrovanou aplikaci a vygenerované přístupové tokeny a bezpečnostní klíče. Ty jsou v rámci zapezpečení nastaveny v systému, na němž běží aplikace, jako *proměnné prostředí*.

O stahování se stará třída *TarifomatDownloader*. Ta se nejprve připojí na API, jež poskytuje Twitter, poté zjistí id posledního staženého příspěvku a na základě dotazu vyhledává a stahuje příspěvky, jež mají vyšší id, než má poslední uložený záznam. Dotaz je omezen tak, aby se nestahovaly tzv. *retweets*. To jsou příspěvky, které uživatel nenapsal sám, ale pouze sdílel od někoho jiného. V případě, kdy by se stahovaly také, by vznikalo vysoké množství duplicitních příspěvků. Dotaz má následující tvar:

```
O2 OR vodafone OR T-Mobile OR mobilní data OR tarif -filter:retweets
```

Jelikož se na Twitteru příspěvky k jiným operátorům, než je O2, Vodafone, nebo T-Mobile vyskytují dle pozorování jen velmi zřídka, tak nejsou v dotazu uvažovány.

Po stažení dat dojde k odstranění příspěvků majících za autora některého z operátorů. Tyto příspěvky totiž většinou žádný postoj nenesou. Po vyfiltrování jsou stažená data převedena do společného formátu, který je popsán v sekci zabývající se způsobem uložení dat. Dochází také k uložení identifikátoru posledního staženého příspěvku do databáze.

4.1.3 Stahování dat z Tarifomatu

Jako další platforma, z níž se stahují data, byl zvolen Tarifomat. Hlavním důvodem je, že obsahuje nejenom příspěvky vztahující se k operátorům, ale i hodnocení tvůrcem příspěvku

[2] Popis a současný stav problému: https://developers.facebook.com/status/issues/205942813488872/

ve formě palce nahoru/dolů. Tudíž lze tyto příspěvky využít jako bázi dat, na jejíž základě se bude klasifikátor učit přiřazovat postoj novým příspěvkům.

Nevýhodou stahování dat z Tarifomatu je fakt, že neposkytuje žádnou veřejnou API. Další nevýhodou je, že přiřazený postoj se ne vždy shoduje s textem příspěvku. Proto byl v systému zaveden validátor příspěvků. Ten umožňuje zkontrolovat a případně změnit přiřazené hodnocení.

Stahování příspěvků z Tarifomatu se provádí automaticky každé ráno v 1:00. Lze provést i manuálně z uživatelského rozhraní. Skript pro stahování je napsaný v jazyce Python a nachází se v souboru *download_posts_from_tarifomat.py*. Interně využívá třídu *TarifomatDownloader* starající se nejenom o stažení dat, ale také o transformaci do formátu vhodného pro uložení do databáze.

Třída TarifomatDownloader obsahuje seznam id operátorů. Tyto identifikátory jsou ve stejné podobě, jakou používá tarifomat. Pro každý z těchto identifikátorů se vytvoří URL a odešle se GET požadavek. Například pro operátora s id o2 se vytvoří následující URL.

```
https://tarifomat.cz/recenze/mobilni-tarify/o2
```

Část *https://tarifomat.cz/recenze/mobilni-tarify/* je stejná pro každého operátora.
Požadavek je odesílán na server s parametry:

```
params = {
     'ajax': 'true',
     'get_reviews': '1',
     'from': from_post,
     'count': download_posts_batch_size
}
```

Parametr *from_post* určuje od kolikátého příspěvku se data stahují a *download_posts_batch_size* ovlivňuje, kolik příspěvků bude staženo. Tento požadavek je volán rekurzivně, dokud nedojde ke zjištění, že aktuální příspěvek už byl stažen. Toto ověřování se provádí na základě data vytvoření příspěvku. To se porovnává s datem posledního staženého příspěvku pro daného operátora, jež je uloženo v databázi.

Zpracování dat stažených z Tarifomatu

Data stažená z Tarifomatu jsou ve formátu HTML. Tudíž je potřeba je očistit a zachovat jen potřebné informace. Očištění dat je provedeno použitím regulárních výrazů. Získaná data obsahují následující informace: text příspěvku, přiřazené hodnocení (negativní/pozitivní), datum vytvoření, jméno autora a lokalita autora (pokud je zadaná). Kromě jmenovaných položek se ukládá ještě informace, že příspěvek je z Tarifomatu.

4.2 Způsob uložení dat

Data jsou ukládána do *PostgreSQL* databáze. Příspěvky se ukládají do tabulky *sentimentAnalyser_post*, která obsahuje sloupce **id**, **operator** a **post**. Id je automaticky generované číslo jednoznačně určující záznam. Operator je string obsahující identifikátor operátora. Post je datového typu *jsonb* a obsahuje všechny ostatní informace o příspěvku. Ty jsou následující:

```
{
    "id": id,
    "dataSource": dataSource,
    "created_time": created_time,
    "message": message,
    "from" {
        "id": id,
        "name": name,
        "location": location
    }
}
```

Kromě těchto dat se příspěvku dynamicky přidávají ještě následující atributy:

- aspects - přidává se po přiřazení aspektů
- sentiment - přidává se po přiřazení postoje k příspěvku
- validation_sentiment - obsahuje informaci, zda byl postoj validován
- validation_aspects - obsahuje informaci, zda byly přiřazené aspekty validovány
- sentiment_source - obsahuje postoj, jež byl příspěvku přiřazen zdrojem (například na Tarifomatu)
- sentiment_manual - obsahuje postoj, jež byl příspěvku přiřazen manuálně
- sentiment_auto - obsahuje postoj, jež byl příspěvku přiřazen automaticky

Aktuální hodnota postoje je vždy pod atributem *sentiment*. Atributy související s postojem mají následující prioritu: *sentiment_manual* > *sentiment_source* > *sentiment_auto*. V atributu *sentiment* se vždy nachází stejná hodnota, jako je obsažena v atributu vztahující se k postoji s nejvyšší prioritou. Toto rozdělení umožňuje zkoumat, v jakých případech dělá klasifikátor nejčastěji chyby.

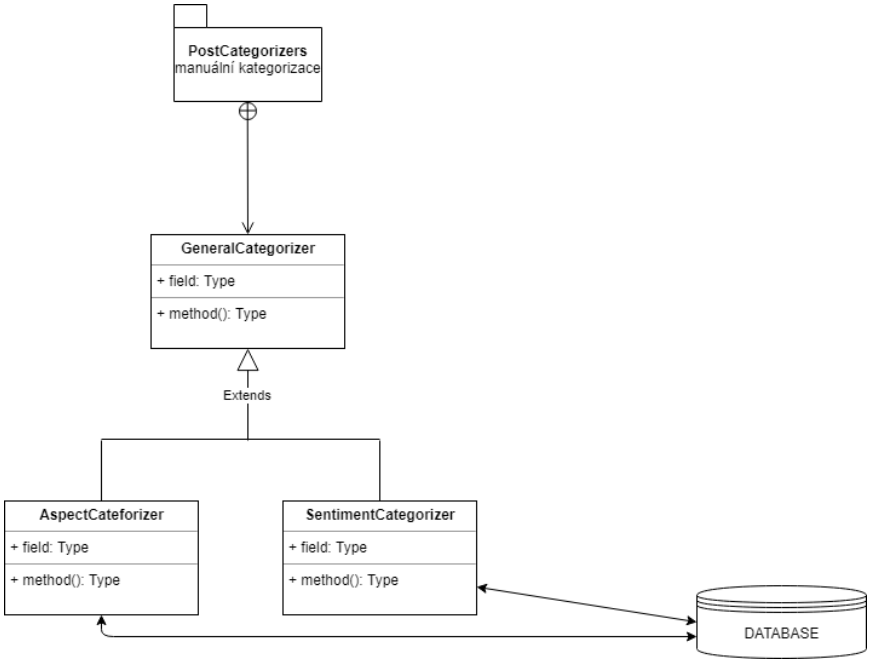
Políčka *validation_sentiment* a *validation_aspects* slouží k označení příspěvků, jejichž přiřazení je potvrzené administrátorem systému.

4.3 Manuální klasifikace dat

Jelikož jsem se pro klasifikaci dat rozhodl použít strojové učení s učitelem, bylo nejprve nutné anotovat data. Za tímto účelem byly vytvořeny na třídy *GeneralCategorizer*, *AspectCategorizet* a *SentimentCategorizer*.

Díky využití dědičnosti obsahují třídy *AspectCategorizet* a *SentimentCategorizer* jen seznamy možných položek k výběru ke kategorizaci a veškerou logiku a komunikaci se serverem obstarává třída *GeneralCategorizer* 4.2.

Kategorizace probíhá v dialogu, který zobrazí určitý počet příspěvků (20), jež uživatel následně hodnotí. Kdykoliv v průběhu hodnocení lze dialog uzavřít, nebo uložit a uzavřít 4.3. Po uložení jsou data odeslána na server a uložena do databáze s nastaveným atributem *sentiment_manual* a *sentiment* u každého z hodnocených příspěvků.



Obrázek 4.2: Manuální kategorizace - UML

Obrázek 4.3: Manuální kategorizace

4.4 Předzpracování dat

Před použitím dat pro klasifikaci dochází k jejich zpracování. Zprávy příspěvků jsou nejdříve tokenizovány na věty, k čemuž se využívá knihovna *NLTK*. Poté jsou z jednotlivých vět za pomoci regulárních výrazů odstraněny URL adresy a je provedena tokenizace na úroveň slov. Takto upravená data přijímá třída *TokenizedTextPreprocessor* a dále s nimi pracuje.

Tato třída umožňuje převést slova na malá písmena, odstranit interpunkci, odstranit stop slova odstranit diakritiku a získat lemmu. Ve vývojové verzi obsahuje tato třída dva různé „lemmatizátory“. Jeden pro slova s diakritikou, druhý pro slova bez ní. Pro tyto účely se využívá knihovna Morphodita[3]. V produkčním serveru se nachází pouze jeden kvůli jejich paměťové náročnosti a nastaveným limitům na Heroku, kam je aplikace nasazena.

Lemmatizaci lze ovlivnit několika parametry. Mezi ně patří odstranění neznámých slov, zachování negace a zachování superlativů. Odstranění neznámých slov snižuje velikost vektoru příznaků tím, že slova, jež nejsou Morphoditou rozpoznána, se neberou v potaz.

Morphodita neprovádí pouze lemmatizaci, ale i POS tagging. Toho lze využít pro zachování negace a superlativů. Negace je důležitá, jelikož často může ovlivnit výsledný postoj, jenž příspěvek nese. Superlativy se zachovávají z toho důvodu, že mají pro výsledný postoj obvykle větší význam, než ostatní slova.

Třída *TokenizedTextPreprocessor* obsahuje i metodu připravenou pro opravení chybných výrazů, ale ta se v současnosti nevyužívá.

4.5 Automatická klasifikace

Systém dokáže automaticky přiřadit příspěvku postoj a kategorii.

4.5.1 Automatické přiřazení postojů

Automatické přiřazení postojů se provádí manuálně na základě přání uživatele s administrátorským oprávnění a automaticky každé ráno ve 2:00. Akce, jež jsou prováděny automaticky má na starosti skript *scheduler.py*. Jelikož se všechny automatické akce provádějí v jiném procesu, než v tom, v němž běží samotná aplikace, tak nedochází ke zvýšení latence při klasifikaci.

Automatické operace jako je stahování dat z Tarifomatu a klasifikace postojů fungují v současnosti pouze na serveru, kde operační systém podporuje příkaz *fork*. V příloze je popsáno, jak spustit aplikaci na operačním systému Windows.

Klasifikátor pracuje následovně. Nejprve získá data, které mají přiřazené hodnocení manuálně, nebo je validované. Ty tvoří bázi pro učení klasifikátoru. Poté načte dokumenty k nimž ještě není postoj přiřazen.

Obě tyto skupiny jsou následně předzpracovány způsobem popsaným v předešlé sekci a jsou z nich vytvořeny vektory příznaků za pomocí knihovny scikit-learn[4]. Po získání vektoru příznaků je použit SGD klasifikátor také z knihovny scikit-learn pro natrénování a následné přiřazení postoje k jednotlivým příspěvkům. Tento klasifikátor se osvědčil jako nejlepší z testovaných. Podrobný přehled metrik jednotlivých technik pro klasifikaci se nachází v následující kapitole věnující se vyhodnocení systému.

Přiřazený postoj může být buď kladný, záporný, nebo neutrální. Neutrální je ten, v němž autor nevyjadřuje žádný subjektivní názor vůči službě, či produktu. Typicky se jedná o dotazy.

4.5.2 Automatické přiřazení kategorie příspěvku

Příspěvek může patřit do jedné, či více z následujících kategorií: signál, internet, tarif, komunikace, smlouva, služby, cena, produkt, nezařazeno. Pro přiřazení této kategorii auto-

[3] http://ufal.mff.cuni.cz/morphodita
[4] http://scikit-learn.org/stable/

maticky byl zvolen stejný postup jako pro přiřazení postoje. To se ukázalo jako nevhodné řešení, jelikož daný algoritmus dokáže přiřadit příspěvku pouze jednu třídu. Navíc při učení očekává u příspěvku přiřazenu pouze jednu hodnotu, což vede k nutnosti ze seznamu kategorií, jež má dokument přiřazené, vybrat pouze jednu.

Vzhledem k těmto potížím se přiřazení kategorií provádí jen na popud uživatele. To znamená, že se neprovádí každý den v určitou dobu jako přiřazení postoje. Toto by však šlo v případě použití lepšího algoritmu jednoduše změnit. Přesné výsledky klasifikace se nachází v následující kapitole.

4.6 Vizualizace dat

4.6.1 Vizualizace ve formě diagramů

Pro vizualizaci dat byla použita JavaScriptová knihovna amCharts[5]. Systém vykresluje dva různé diagramy. První zobrazuje hodnocení operátorů v rámci jednoho roku. Druhý vykresluje hodnocení jednotlivých kategorií v průběhu roku. Rok je volitelný. Původní návrh obsahoval možnost zvolit si zobrazení hodnocení v rámci měsíce, ale kvůli nedostatku dat z něj sešlo.

Oba tyto grafy mají společného předka *GeneralPopularityChart*. Ta obstarává vykreslení grafu. Potomci, kterými jsou *AspectsPopularityChart* a *OperatorsPopularityChart* se starají o načtení dat ze serveru a vykreslení popisku v jednotlivých bodech diagramu. Struktura části systému starající se o vytvoření grafů je následující 4.4. Výsledný graf vypadá například pro rok 2017 následovně 4.5.

Po najetí na jednotlivé body v grafu lze vidět detailní informace pro daný měsíc. Detail se skládá z celkového počtu příspěvků a informace o tom, kolik z nich bylo pozitivních, negativních a neutrálních. Tyto počty jsou důležité z toho důvodu, že umožňují uživateli rozhodnout, zda je hodnocení reprezentativní, nebo ne (pokud má operátor v daném měsíci jen jedno hodnocení, tak je porovnání s operátorem, jenž jich má desítky nicneříkající).

V případě grafu popularity mobilních operátorů se dají jednotlivé body ještě rozkliknout. Po této operaci dojde k přesměrování na stránku, kde jsou zobrazeny pro daného operátora ve vybraném měsíci příspěvky.

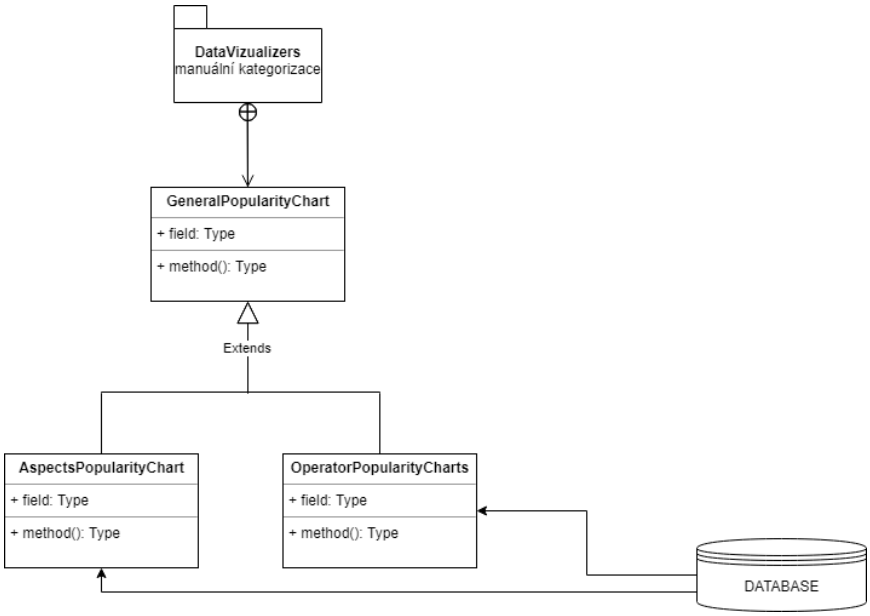
4.6.2 Vizualizace ve formě seznamu příspěvků

Druhým způsobem, jak systém vizualizuje data je jako seznam příspěvků. Tento způsob zajišťuje mnohem větší volnost ve filtrování a také umožňuje uživatelům s administrátorským oprávněním měnit, mazat a validovat data. Kvůli velkému množství uložených příspěvků v databázi je použito stránkování, kdy je na stránce zobrazeno 20 příspěvků.

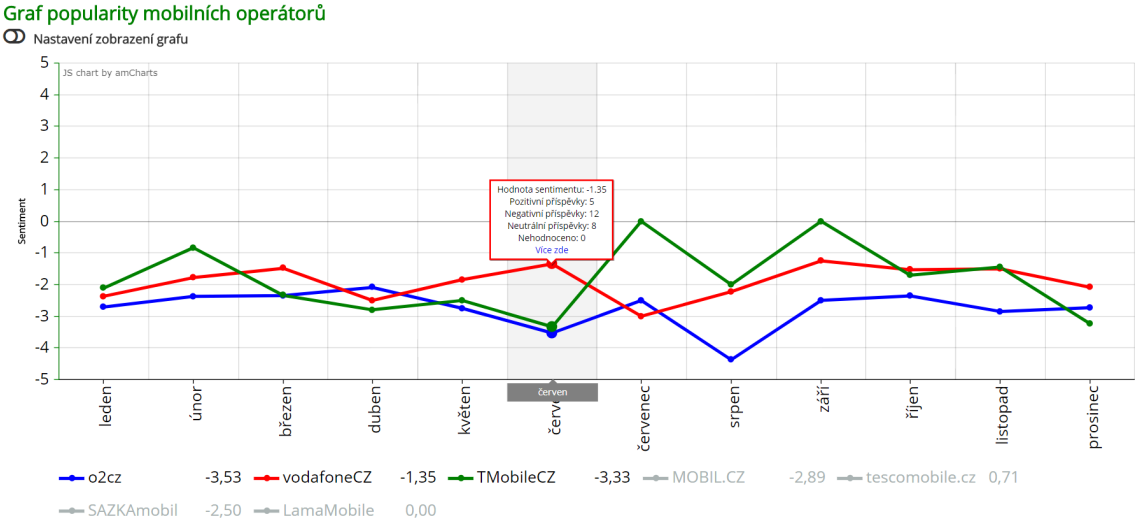
Příspěvky jsou od sebe barevně odlišeny přiřazeným postojem 4.6.

Tato stránka by měla sloužit jako hlavní místo pro editaci dat a detailní vyhledávání v případě, kdy uživatelům nestačí obecný grafický přehled. Tím, že zde mohou uživatelé s oprávněním validovat a měnit přiřazené kategorie bude pravidelně docházet ke zvyšování přesnosti systému.

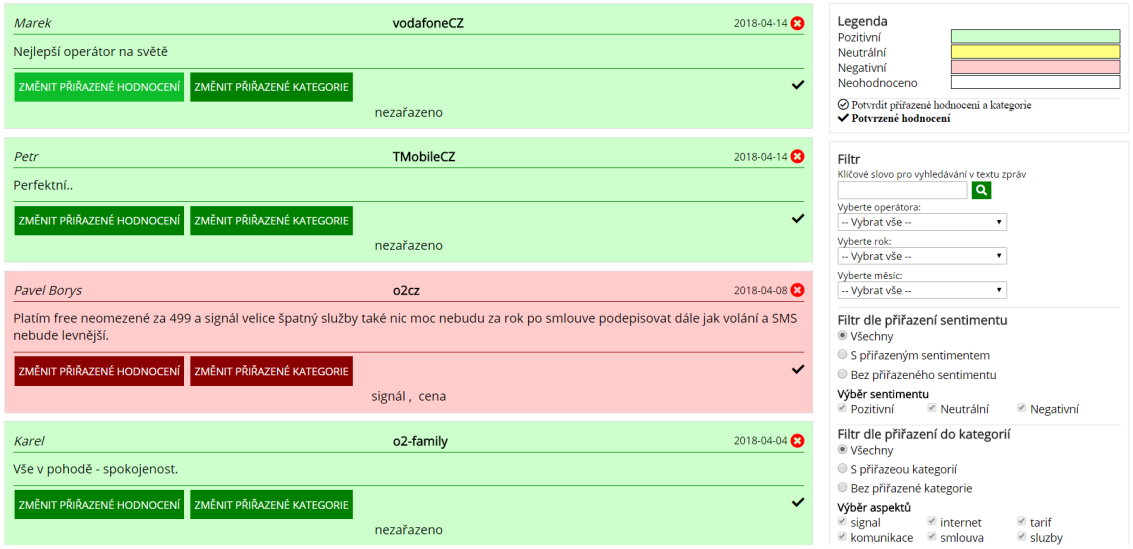
[5] https://www.amcharts.com/



Obrázek 4.4: Vizualizace dat - schéma



Obrázek 4.5: Vizualizace dat - graf popularity operátorů



Obrázek 4.6: Vizualizace dat - seznam příspěvků

Kapitola 5

Vyhodnocení systému

Tato kapitola nejdříve popisuje rozsah dat, které byly nasbírány a diskutuje, zda je to množství dostatečné.

Poté popisuje výsledky jichž je systém schopný při klasifikaci dosáhnout. Nachází se zde porovnání jednotlivých algoritmů a technik využitých v implementovaných v rámci této práce.

5.1 Rozbor a velikost datasetu

Dne 30. 4. 2018 se v databázi nacházelo 10 268 příspěvků s přiřazeným postojem. Příspěvky jsou z let 2014 - 2018. Zdrojem těchto příspěvků je Facebook a Tarifomat. Z těchto příspěvků má kategorii přiřazeno 1457 příspěvků. Důvod, proč není kategorie přiřazena všem příspěvkům je v současnosti nízká přesnost systému při přiřazování kategorií. Viz podrobnosti v následující sekci.

Podrobný přehled o příspěvcích udává následující tabulka 5.1:

Tabulka 5.1: Příspěvky v databázi

KATEGORIE	POČTY PŘÍSPĚVKŮ				
	Pozitivní	Negativní	Neutrální	Nehodnoceno	Celkem
Signál	16	8	23	0	47
Internet	19	79	201	0	299
Tarif	6	21	34	0	61
Komunikace se zákazníkem	9	22	164	0	195
Smlouva	0	14	38	0	52
Služby	7	55	52	0	114
Cena	15	27	112	0	154
Produkty	0	8	3	0	11
Nezařazeno	68	321	135	0	524
Nekategorizováno	2319	2356	4136	0	8811
Celkem	2459	2911	4898	0	10268

5.2 Přesnost systému

Verze, která je veřejně přístupná nedosahuje z testovaných konfigurací nejlepších výsledků, ale zároveň o moc nezaostává. Důvod, proč není veřejná verze ta nejpřesnější, je ten, že nejpřesnější verze má téměř dvojnásobné paměťové nároky. To je zapříčiněno tím, že se využívají dva různé lemmatizátory a POS taggery (v závislosti na tom, zda dokument obsahuje diakritiku). Tato verze se poté nevejde do limitů, které jsou nastaveny na Heroku, kam je aplikace nasazena.

5.2.1 Přesnost při přiřazení postojů

Výsledky nasazené verze jsou následující 5.2:

Tabulka 5.2: Metriky systému - nasazená verze

	Accuracy	Precision	Recall	F1 score
BOW + Naive Bayes	0,705	0,753	0,705	0,668
BOW + SGD	0,753	0,748	0,753	0,744
TF- IDF + Naive Bayes	0,680	0,800	0,680	0,618
TF-IDF + SGD	0,710	0,742	0,710	0,675

Výsledky nejúspěšnější konfigurace systému se pohybují v závislosti na použitých klasifikačních algoritmech o jednotky procent výše, což ukazuje tabulka 5.3.

Tabulka 5.3: Metriky systému - nejpřesnější verze

	Accuracy	Precision	Recall	F1 score
BOW + Naive Bayes	0,712	0,760	0,712	0,679
BOW + SGD	0,753	0,755	0,753	0,744
TF- IDF + Naive Bayes	0,698	0,803	0,698	0,645
TF-IDF + SGD	0,723	0,784	0,723	0,689

Pro automatickou klasifikaci byl zvolen klasifikátor, jenž využívá *Stochastic Gradient Descent* a k extrakci příznaků je použita *BOW* reprezentace. Tato konfigurace podává nejstabilnější výsledky v rámci jednotlivých metrik.

V použité konfiguraci se využívají všechny techniky pro zpracování textu popsané výše. Nejprve dochází k rozdělení textu do vět, poté k odstranění URL adres z textu, následně k tokenizaci na slova, převodu všech písmen malá, odstranění interpunkce, odstranění stop slov, lemmatizaci a odstranění slov nepatřících do slovníku.

Poté je vytvořen vektor příznaků za použití BOW reprezentace, jelikož ta pro testovací sadu dosahovala většinou lepších výsledků, než TF-IDF reprezentace.

Při použití extrakce příznaků došlo k experimentování s velikostí jednotek textu - unigramy, bigramy a trigramy. Nejlepších výsledků bylo dosaženo s unigramy, což odpovídá teorii, že velikost datasetu může mít vliv na výběr vhodné velikosti textových jednotek při extrakci příznaků. Rozdíly však byly pouze v rámci jednotek procent.

Dalším zajímavým zjištěním bylo, že *Naive Bayes* klasifikátor funguje lépe s unigramy, zatímco *Stochastic Gradient Descent* s bigramy. Při použití trigramů a vyšších n-gramů došlo ke snížení přesnosti klasifikace.

Jak je vidět, tak systém nedosahuje nejlepších výsledků v porovnání se systémy popsanými ve druhé kapitole. To je zapříčiněno především menším datasetem a tím, že i v datasetu

použitém pro natrénování systému se vyskytují chyby (pokud se jedná o příspěvky stažené z Tarifomatu).

5.2.2 Přesnost při přiřazování kategorií

Přesnost při přiřazování kategorií zdaleka nedosahuje přesnosti při přiřazování postojů. Vliv na to má několik faktorů. Prvním z nich je, že kategorií, do nichž může být text přiřazen, je více než druhů postojů (8 vs. 3). Dalším důvodem je, že dokument může mít přiřazeno více kategorií, zatímco postoj má pouze jeden. Jako příklad uvádím větu „Internet je děsný a výpadky signálu jsou časté.“. V tomto případě dokument patří do kategorií *internet* a *signál*.

Zvolený algoritmus pro klasifikaci však neumí pracovat s přiřazením více kategorií jednomu dokumentu, tudíž bylo nutné vždy vybrat jen jednu ze zvolených kategorií. To se bezesporu také projevilo na výsledcích klasifikace, které se nacházejí na následující tabulce 5.4.

Tabulka 5.4: Přiřazení kategorií

	Accuracy	Precision	Recall	F1 score
BOW + Naive Bayes	0,820	0,476	0,221	0,302
BOW + SGD	0,806	0,208	0,159	0,180
TF- IDF + Naive Bayes	0,816	0,366	0,142	0,204
TF-IDF + SGD	0,812	0,254	0,168	0,202

5.3 Testování na uživatelích

V rámci testování systému jsem provedl i testování na uživatelích. Toto testování probíhalo formou dotazníku, v němž se nacházely otázky, jež se vztahovaly k operátorům a odpovědi na něž bylo možné v systému najít.

Dotazník byl následující:

- Ve kterém měsící v roce 2017 mělo O2 vyšší hodnocení než Vodafone a T-Mobile?
- Ve kterém měsící v roce 2016 se k O2 vztahovalo nejvíce příspěvků (v porovnání s ostatními měsící v daném roce)?
- Co zapříčinilo, že byl v daném měsící vyšší počet příspěvků než v ostatních měsících v daném roce?
- Nachází se mezi daty příspěvky vztahující se k O2TV, které by měly kladné hodnocení? Pokud ano, kolik jich je?
- Kolik příspěvků je z března 2018 a vztahují se k T-Mobile? Kolik z nich je kladných, kolik neutrálních a kolik záporných?

Dotazy byly položeny tak, aby se ne vždy stačilo podívat na graf (což stačí pouze u první otázky), ale bylo nutné i hlubší zkoumání dat. Například otázka 3 vede k tomu, že by měl uživatel odhalit, že za zvýšeným množstvím příspěvků v daném měsíci stojí změna ve způsobu, jak se O2 zachovalo k vyčerpání mobilních dat. Zatímco předtím došlo k snížení

rychlosti, tak v červnu 2016 nastala změna a O2 mobilní data po vyčerpání omezila úplně. Tato otázka má za cíl ukázat, že systém poskytuje prostředky pro zobrazení výkyvů v hodnocení/počtu příspěvků a nalezení důvodů, proč k nim došlo.

Čtvrtá otázka se naopak vztahuje ke konkrétní službě, takže ke správné odpovědi bylo nutné využít textové vyhledávání. Tato otázka má doložit, že je systém schopný poskytnou odpovědi vztahující se ke konkrétní službě, kterou operátor posyktuje.

Mezi lidmi, kteří na dané otázky odpovídali byli spolužáci z vysoké školy, spolubydlící, přátelé a někteří kolegové z práce. Celkově se ho zúčastnilo 14 osob.

Na první dvě otázky odpověděli všichni správně. Třetí otázka byla problematická a kompletně správných odpovědí bylo pouze 6. Odpovědi typu, že byla změněna smlouva, nebo zvýšena cena tarifu jsem neuznával, protože byly příliš vágní. Tato otázka vyžadovala procházení a pročítání příspěvků, což nejspíše způsobilo nižší počet správných odpovědí. Proto by se do programu hodil zapracovat podsystém, který by příspěvky sumarizoval. Na čtvrtou otázku odpověděli správně opět všichni. Poslední otázku mělo správně 8 lidí. To bylo zapříčeněno tím, že někteří dotazník vyplňovali ještě v době, kdy se v systému nalézala chyba se špatným indexováním měsíců.

Toto testování dopomohlo k odhalení chyby, kdy se na stránce s příspěvky špatně indexovaly měsíce. Tudíž po nastavení na červen byly zobrazeny příspěvku ke květnu. Ukázalo se, že většina uživatelů je v systému schopná najít infromace, které jsem zadal. Největší problém způsobila otázka, jež se ptala na příčinu výkyvu počtu příspěvků v určitém měsící pro konkrétního operátora. Také se dle zpětné vazby ukázalo, že by si uživatelé přáli do systému přidat graf, který by zobrazoval počty příspevků v jednotlivých měsících. V současném stavu musí v grafu najet na konkrétní bod, aby tuto informaci zjistili.

5.4 Možné rozšíření a zdokonalení systému

V současnosti se v systému nachází několik částí, jež by bylo možné v budoucnu vylepšit. Především se jedná o jiný způsob přiřazení kategorií příspěvkům, jelikož zvolený přístup se ukázal jako neefektivní.

Tento problém by pravděpodobně vyřešilo, kdyby byla prováděna aspektově orientovaná analýza postojů. Poté by se totiž kategorie nepřipisovala celému dokumentu, ale pouze části, k níž se vztahuje postoj (k aspektu).

Dalším možným a doporučeným rozšířením by bylo zakomponování stahování dat z různých zdrojů. To především z důvodu, že Facebook omezil přístup k informacím neaktivním aplikacím. A v současnosti neprovádí revize, aby se aplikace dala aktivovat. Tento problém by měl být v budoucnu vyřešen, přesto by však RSS zdroje pravděpodobně poskytovaly stabilnější zdroj dat.

Mezi další vylepšení patří poskytování více druhů zobrazení (různé typy diagramů) a udržování si informací o přispěvatelích.

Kapitola 6

Závěr

V rámci této práce byl implementován systém, který umožňuje automaticky získávat data týkající se mobilních operátorů, analyzovat je a následně přehledně vizualizovat. Většina informací, které se v současnosti dá z internetu o mobilních operátorech získat, je pouze v textové podobě. Ať už se jedná o příspěvky ze sociálních sítí, recenze, nebo diskuze u článků.

Tento systém poskytuje prostředky pro automatickou analýzu těchto dat a jejich vizualizace ve formě diagramů. Dále umožňuje pokročilé možnosti filtrování dat, ať už na základě přiřazené kategorie, nebo i prostého textu.

Privilegovaným uživatelům dále dovoluje měnit přiřazené hodnocení dokumentům, validovat jej a odstraňovat dokumenty, které se netýkají operátorů. Příkladem takového dokumentu je spam v diskuzích. Tato možnost vede k tomu, že přesnost klasifikace se bude postupně zvyšovat, pokud budou výsledky automatického přiřazení pravidelně revidovány.

Systém autonomně každý den stahuje data z Tarifomatu a Twitteru a poté provádí analýzu příspěvků. Příspěvky z Facebooku se v současnosti automaticky nestahují, jelikož Facebook omezil v březnu 2018 využívání GraphAPI pro nautorizované aplikace.

Výsledky přesnosti přiřazení postojů se pohybují mezi 70% - 75%, což je výsledek, jenž mírně zaostává za moderními systémy.

Při dalším vývoji projektu by bylo vhodné implementovat aspektově orientovanou analýzu postojů. Tento krok by pravděpodobně odstranil problém s nízkou přesností při zařazování dokumentů do kategorií. Dalším vylepšením by bylo stahovat data z více různých zdrojů. Především, pokud problém s omezením stahování dat z Facebooku bude trvat nadále.

V případě rozšíření o stahování z více zdrojů stačí udržet strukturu v jaké jsou příspěvky ukládány do databáze a analýza a vizualizace bude automaticky fungovat. Díky modularitě systému lze také jednoduše přidávat další diagramy.

Literatura

[1] Ahmed, K.; Tazi, N. E.; Hossny, A. H.: Sentiment Analysis over Social Networks: An Overview. In *2015 IEEE International Conference on Systems, Man, and Cybernetics*, 2015, s. 2174–2179.

[2] Akhtar, N.; Zubair, N.; Kumar, A.; aj.: Aspect based Sentiment Oriented Summarization of Hotel Reviews. *Procedia Computer Science*, roèník 115, 2017: s. 563–571.

[3] Asghar, M. Z.; Khan, A.; Ahmad, S.; aj.: A review of feature extraction in sentiment analysis. *Journal of Basic and Applied Scientific Research*, roèník 4, è. 3, 2014: s. 181–186.

[4] Bermingham, A.; Conway, M.; McInerney, L.; aj.: Combining social network analysis and sentiment analysis to explore the potential for online radicalisation. In *Social Network Analysis and Mining, 2009. ASONAM'09. International Conference on Advances in*, IEEE, 2009, s. 231–236.

[5] Brants, T.: TnT: a statistical part-of-speech tagger. In *Proceedings of the sixth conference on Applied natural language processing*, Association for Computational Linguistics, 2000, s. 224–231.

[6] Cambria, E.; Das, D.; Bandyopadhyay, S.; aj.: *A practical guide to sentiment analysis*, roèník 5. Springer, 2017, ISBN 978-3-319-55392-4.

[7] Chmelař Petr, H. M. a. B. V., Hellebrand David: Nalezení slovních kořenů v češtině.

[8] Collomb, A.; Costea, C.; Joyeux, D.; aj.: A study and comparison of sentiment analysis methods for reputation evaluation. *Rapport de recherche RR-LIRIS-2014-002*, 2014.

[9] Cui, H.; Mittal, V.; Datar, M.: Comparative experiments on sentiment classification for online product reviews. In *AAAI*, roèník 6, 2006, s. 1265–1270.

[10] Glez-Peña, D.; Lourenço, A.; López-Fernández, H.; aj.: Web scraping technologies in an API world. *Briefings in Bioinformatics*, roèník 15, è. 5, 2014: s. 788–797.

[11] Hassan, A. U.; Hussain, J.; Hussain, M.; aj.: Sentiment analysis of social networking sites (SNS) data using machine learning approach for the measurement of depression. In *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, 2017, s. 138–140.

[12] Hodson, H.: Twitter hashtags predict rising tension in Egypt. 2013.

[13] Jana, K.: Sémantická anlýza a jmenné fráze. 2016.

[14] Khan, A.; Baharudin, B.; Khan, K.: Sentiment classification from online customer reviews using lexical contextual sentence structure. In *International Conference on Software Engineering and Computer Systems*, Springer, 2011, s. 317–331.

[15] Liu, B.; Hu, M.; Cheng, J.: Opinion observer: analyzing and comparing opinions on the web. In *Proceedings of the 14th international conference on World Wide Web*, ACM, 2005, s. 342–351.

[16] Mitkov, R.: *The Oxford Handbook of Computational Linguistics*. Oxford University Press, ISBN 9780199276349.

[17] Moghaddam, S.; Ester, M.: Opinion digger: an unsupervised opinion miner from unstructured product reviews. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, ACM, 2010, s. 1825–1828.

[18] Moghaddam, S.; Ester, M.: ILDA: interdependent LDA model for learning latent aspects and their ratings from online product reviews. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, ACM, 2011, s. 665–674.

[19] Mudinas, A.; Zhang, D.; Levene, M.: Combining lexicon and learning based approaches for concept-level sentiment analysis. In *Proceedings of the first international workshop on issues of sentiment discovery and opinion mining*, ACM, 2012, str. 5.

[20] Ortigosa, A.; Martín, J. M.; Carro, R. M.: Sentiment analysis in Facebook and its application to e-learning. *Computers in Human Behavior*, roèník 31, 2014: s. 527–541.

[21] Pang, B.; Lee, L.; Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, Association for Computational Linguistics, 2002, s. 79–86.

[22] Pang, B.; Lee, L.; aj.: Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, roèník 2, è. 1–2, 2008: s. 1–135.

[23] Ratnaparkhi, A.: A maximum entropy model for part-of-speech tagging. In *Conference on Empirical Methods in Natural Language Processing*, 1996.

[24] Sarkar, D.: *Text Analytics with Python: A Practical Real-World Approach to Gaining Actionable Insights from your Data*. Apress, 2016, ISBN 978-1484223871.

[25] Sasmita, D. H.; Wicaksono, A. F.; Louvan, S.; aj.: Unsupervised aspect-based sentiment analysis on Indonesian restaurant reviews. In *2017 International Conference on Asian Language Processing (IALP)*, Dec 2017, s. 383–386.

[26] Statista: *Most famous social network sites worldwide as of January 2018, ranked by number of active users* . [Online; navštíveno 15.04.2018].
URL https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/

[27] Wang, D.; Zhu, S.; Li, T.: SumView: A Web-based engine for summarizing product reviews and customer opinions. *Expert Systems with Applications*, roèník 40, è. 1, 2013: s. 27–33.

[28] Wikipedie, P.: *Search engine indexing*. [Online; navštíveno 23.04.2018].
URL https://en.wikipedia.org/wiki/Search_engine_indexing/

[29] Yang, D.; Zhang, D.; Yu, Z.; aj.: A sentiment-enhanced personalized location recommendation system. In *Proceedings of the 24th ACM Conference on Hypertext and Social Media*, ACM, 2013, s. 119–128.

[30] Zhang, M.; Zhang, Y.; Fu, G.: Tweet sarcasm detection using deep neural network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, s. 2449–2460.

[31] Zobel, J.; Moffat, A.; Ramamohanarao, K.: Inverted files versus signature files for text indexing. *ACM Transactions on Database Systems (TODS)*, roèník 23, è. 4, 1998: s. 453–490.

Příloha A

Nastavení a instalace aplikace

Aplikace je volně přístupná na adrese https://guarded-journey-49460.herokuapp.com/BachelorThesis. Aby byly dostupné všechny operace, které lze s daty provádět, je nutné se přihlásit jako administrátor. K tomu slouží uživatelské jméno **admin** s heslem **admin_admin**.

Vlastní instalace aplikace je složitější. Stroj, na kterém má aplikace běžet musí obsahovat Python 3.6. Knihovny potřebné pro běh programu jsou následující:

- APScheduler==3.5.1
- certifi==2017.11.5
- chardet==3.0.4
- click==6.7
- dj-database-url==0.5.0
- Django==2.0.1
- gunicorn==19.7.1
- idna==2.6
- nltk==3.2.5
- numpy==1.13.3
- oauthlib==2.0.7
- psycopg2==2.7.3.2
- PySocks==1.6.8
- python-dateutil==2.7.2
- pytz==2017.3
- redis==2.10.6
- requests==2.18.4

- requests-oauthlib==0.8.0
- rq==0.10.0
- scikit-learn==0.19.1
- scipy==1.0.0
- six==1.11.0
- tweepy==3.6.0
- tzlocal==1.5.1
- ufal.morphodita==1.9.2.1
- urllib3==1.22
- whitenoise==3.3.1

Pro nltk je potřeba dostahovat **Punkt Tokenizer Models**:

```
nltk.download('punkt')
```

Také je nutné mít nastavené následující proměnné prostředí:

- *TWITTER_ACCESS_TOKEN*: 904052882546323456-ZDIMqzGKoVO9Tjvb5T23BYbQSumR6ag
- *TWITTER_ACCESS_TOKEN*: 904052882546323456-ZDIMqzGKoVO9Tjvb5T23BYbQSumR6ag
- *TWITTER_CONSUMER_KEY*: pvznnnb5WfLzLFRAURpto9Zbe
- *TWITTER_CONSUMER_SECRET*: LlWxpll4U95de7zo35j3CvGpi5FbmVf7WGsSDCuJfg4oglUbA9
- *BP_SECRET_KEY*: _qqwyshe%a0bg%v(2!4mz)ohnd0yxpgu@9i5w!_qk*t8k3+bxh

V případě, kdy je aplikace spouštěna na Windowsu je nutné nastavit proměnnou prostředí:

- BP_WINDOWS_DEVELOPMENT = true

Na serveru se musí nacházet PostgreSQL databázi, do níž se budou ukládat záznamy.

Po nainstalování všech závislostí je nutno v adresáři, kde se nachází soubor *manage.py*, spusti následující příkazy:

```
python manage.py makemigrations
python manage.py migrate
```

Následně lze server spustit příkazem

```
python manage.py runserver
```

Doporučuji však využít adresu, kde je aplikace zveřejněná, otestovaná a veškerá konfigurace je již nastavená.

Příloha B

Plakát

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

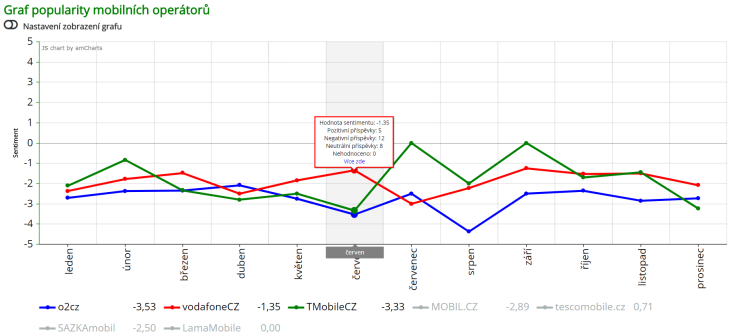
ANALÝZA OBSAHU SOCIÁLNÍCH SÍTÍ TÝKAJÍCÍ SE ČESKÝCH MOBILNÍCH OPERÁTORŮ

JAN PAVLŮ

Cílem této práce bylo vytvořit systém, který umožňuje stahovat, indexovat, analyzovat a vizualizovat data týkající se českých mobilních operátorů.

Výsledná aplikace provádí stažení, uložení a přiřazení postoje automaticky bez nutnosti zásahu uživatele.

SYSTÉM PŘIŘAZUJE POSTOJ S PŘESNOSTÍ OKOLO 75%.



Aplikace také umožňuje uživateli měnit a validovat přiřazené hodnocení a přiřadit příspěvkům kategorii

.

Dále přehledně zobrazuje výsledky ve formě diagramu, na němž se nachází zobrazení spokojenosti uživatelů s operátorem v průběhu roku. Samostatný diagram také vyobrazuje spokojenost s jednotlivými službami operátora, jako je například tarif, mobilní data,..