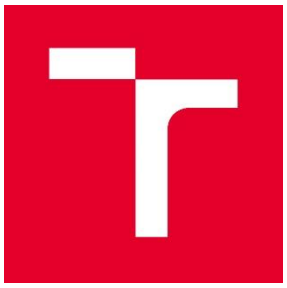


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

ZPRACOVÁNÍ UŽIVATELSKÝCH RECENZÍ

PROCESSING OF USER REVIEWS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — **BC. DITA CIHLÁŘOVÁ**
AUTHOR

VEDOUCÍ PRÁCE — **ING. VLADIMÍR BARTÍK, PH.D.**
SUPERVISOR

BRNO 2019

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav informačních systémů (UIFS) Akademický rok 2018/2019

Zadání diplomové práce

Studentka: **Cihlářová Dita, Bc.**
Program: Informační technologie Obor: Informační systémy
Název: **Zpracování uživatelských recenzí**
Processing of User Reviews
Kategorie: Data mining

Zadání:

1. Seznamte se s problematikou získávání znalostí z textu a souvisejícími oblastmi, zaměřte se na problematiku sumarizace textu.
2. Navrhněte algoritmus, který nalezne v textu uživatelských recenzí produktů v českém jazyce nejvíce komentované vlastnosti produktu a určí, zda je sentiment komentáře pozitivní či negativní.
3. Navrhněte koncepci aplikace, která bude implementovat algoritmus z bodu 2 a zvolte vhodné implementační prostředí.
4. Aplikaci implementujte a proveďte experimenty ověřující úspěšnost implementovaného algoritmu.
5. Zhodnoťte dosažené výsledky a diskutujte další možné pokročování tohoto projektu.

Literatura:

- Feldman, R., Sanger, J.: The Text Mining Handbook. Cambridge University Press, 2007.
- Das, D., Martins, A.: A Survey on Automatic Text Summarization. Carnegie Mellon University, 2007.

Při obhajobě semestrální části projektu je požadováno:

- Body 1 až 3.

Podrobné závazné pokyny pro vypracování práce viz http://www.fit.vutbr.cz/info/szz/

Vedoucí práce: **Bartík Vladimír, Ing., Ph.D.**
Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.
Datum zadání: 1. listopadu 2018
Datum odevzdání: 22. května 2019
Datum schválení: 23. října 2018

Abstrakt

Velmi často lidé nakupují na internetu zboží, které si nemohou prohlédnout a vyzkoušet. Spoléhají se tedy na recenze ostatních zákazníků, ale těch už může být v dnešní době příliš mnoho na to, aby je člověk mohl sám rychle a pohodlně zpracovat. Cílem této práce je nabídnout aplikaci, která dokáže v českých recenzích rozpoznat, jaké vlastnosti produktu jsou nejvíce komentované a zda je vyznění komentářů pozitivní či negativní. Výsledky pak mohou ušetřit velké množství času zákazníkům e-shopů a poskytnout zajímavou zpětnou vazbu výrobcům prodávaných produktů.

Abstract

Very often, people buy goods on the Internet that they can not see and try. They therefore rely on reviews of other customers. However, there may be too many reviews for a human to handle them quickly and comfortably. The aim of this work is to offer an application that can recognize in Czech reviews what features of a product are most commented and whether the commentary is positive or negative. The results can save a lot of time for e-shop customers and provide interesting feedback to the manufacturers of the products.

Klíčová slova

zpracování přirozeného jazyka, dolování v textu, analýza sentimentu, strojové učení, Naive Bayes, Maximum Entropy, vektor příznaků, sumarizace textu, předzpracování textu, NLTK

Keywords

natural language processing, text mining, sentiment analysis, machine learning, Naive Bayes, Maximum Entropy, feature vector, text summarization, text preprocessing, NLTK

Citace

Cihlářová Dita: Zpracování uživatelských recenzí, diplomová práce, Brno, FIT VUT v Brně, 2019

Zpracování uživatelských recenzí

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracovala samostatně pod vedením Ing. Vladimíra Bartíka, Ph.D. Uvedla jsem všechny literární prameny a publikace, ze kterých jsem čerpala.

……
Dita Cihlářová
22. 5. 2019

Poděkování

Ráda bych poděkovala svému vedoucímu doktoru Bartíkovi za cenné rady, velkou trpělivost a pomoc při tvorbě této práce.

Obsah

1 Úvod ... 3
2 Sumarizace textu ... 4
2.1 Předzpracování textu ... 4
2.1.1 Segmentace a tokenizace ... 4
2.1.2 Part of Speech Tagging ... 5
2.1.3 Lemmatizace a stematizace ... 5
2.1.4 Zpracování synonym ... 6
2.1.5 Odstranění stop slov ... 6
2.2 Nalezení frekventovaných množin ... 7
2.2.1 Apriori algoritmus ... 7
3 Analýza sentimentu ... 8
3.1 Slovníkové metody ... 8
3.2 Klasifikace pomocí strojového učení ... 8
3.2.1 Učení s učitelem ... 9
3.2.2 Naive Bayes ... 10
3.2.3 Maximum Entropy ... 10
3.2.4 Vyhodnocování úspěšnosti ... 10
3.2.5 Učení bez učitele ... 12
4 Současná řešení ... 13
4.1 Metoda využívající Apriori a Wordnet ... 13
4.2 SumView – systém pro sumarizaci recenzí ... 13
4.3 Využití LDA a učení bez učitele ... 14
5 Implementace ... 15
5.1 Návrh aplikace ... 15
5.2 Implementace aplikace ... 17
5.2.1 Skript pro stahování recenzí ... 17
5.2.2 Předzpracování ... 19
5.2.3 Nalezení vlastností produktů ... 20
5.2.4 Analýza sentimentu pomocí slovníku ... 20
5.2.5 Analýza sentimentu pomocí strojového učení ... 21
5.2.6 Report výsledků analýzy ... 22
6 Experimenty ... 23
6.1 Dataset ... 23
6.2 Nalezení nejefektivnějšího přístupu ... 24

6.2.1 Hledání témat a vlastností produktu 24
6.2.2 Čas učení modelu 25
6.2.3 Porovnání metod analýzy sentimentu 26
6.2.4 Vyhodnocení úspěšnosti slovníkové metody 28
6.3 Vyhodnocení úspěšnosti aplikace 29
6.3.1 Nejčastěji diskutovaná témata 29
6.3.2 Nejvíce pozitivní témata 30
6.3.3 Nejvíce negativní témata 31
6.4 Shrnutí experimentů 31
7 Závěr 33
8 Bibliografie 35
9 Příloha A – Použitý seznam stop slov 37
10 Příloha B – Doplňující grafy ke kapitole Experimenty 38
11 Příloha C – Ukázkový výstup aplikace 39

1 Úvod

Podle průzkumu Českého statistického úřadu z roku 2018 nakoupilo alespoň jednou za život na internetu 65,3 % Čechů.[*] Co víc, 19 % obyvatel nakoupilo během 3 měsíců alespoň třikrát.[†] Nakupování na internetu se stává stále běžnější součástí našich životů. Velmi často na internetu vybíráme také zboží, které jsme předtím nikdy neviděli nebo nevyzkoušeli. V takovém případě chceme zjistit předem o produktu co nejvíce. Obchodníci ale většinou vyzdvihují hlavně přednosti, proto je dobré se podívat, co o dané věci píší ostatní uživatelé. Recenze již zakoupeného zboží jsou bohatým zdrojem informací, kde lze zjistit silné, ale i slabé stránky.

Dnes už ale recenzí bývá velké množství. I v češtině mají populárnější produkty na větších e-shopech desítky až stovky recenzí. Např. chytré hodinky *Samsung Galaxy Watch 46mm* mají na internetovém obchodě alza.cz 59 recenzí, herní konzole *Nintendo Switch - Neon Red&Blue Joy-Noc* má 30 a *iPhone 6s 32GB Gold* má 179 recenzí.[‡] Takový počet už není pro potenciálního zákazníka pohodlné procházet manuálně. Tato práce si klade za cíl usnadnit zákazníkovi internetového obchodu jeho rozhodování o koupi tím, že dostane k dispozici seznam prvků produktu komentovaných ostatními uživateli. Zákazník se dozví, co jsou nejčastěji zmiňované dobré a špatné vlastnosti produktu, aniž by musel ručně procházet desítky recenzí. Tato funkcionalita může být zároveň zajímavá také pro výrobce, kteří tak mohou pohodlně získat cennou zpětnou vazbu na své produkty.

Tato práce je rozčleněná do několika logických celků. Kapitola 2 se věnuje předzpracování textu, sumarizaci a hledání frekventovaných množin. V kapitole 3 jsou nastíněny základní metody analýzy sentimentu – slovníková metoda a strojové učení. Kapitola 4 ukazuje některé současné přístupy ke zpracování uživatelských recenzí. V kapitole 5 lze nalézt popis navrhované aplikace, některých nástrojů a knihoven, které byly použity pro implementaci, a představení samotné implementace. Kapitola 6 se věnuje experimentům a vyhodnocení, které metody přinesly nejlepší výsledky. Vše je pak shrnuto a uzavřeno v kapitole 7, kde je také nastíněno možné pokračování práce.

[*] Průzkum Českého statistického úřadu: Jednotlivci v ČR nakupující na internetu, 2018. https://www.czso.cz/documents/10180/61508128/0620041887.pdf/6945e327-b595-499a-a263-6d2afe2be9cd?version=1.2

[†] Průzkum Českého statistického úřadu: Počet nákupů na internetu uskutečněných jednotlivci v ČR během 3 měs., 1.čtvrtletí 2018. https://www.czso.cz/documents/10180/46014700/06200417101.pdf/68a0159c-734d-48ac-bf7a-89bb918cb296?version=1.1

[‡] Internetový obchod www.alza.cz. Navštíveno 14. 1. 2019.

2 Sumarizace textu

Sumarizace textu je jednou z úloh v oblasti dolování v textu. Zabývá se extrakcí důležitých informací z dokumentů*. Das a Martins [1] popisují výstup sumarizace ve třech bodech:

1) Souhrn je možno získávat z jednoho nebo více dokumentů.
2) Souhrn zachovává důležité informace.
3) Souhrn by měl být krátký.

Tato práce se drží všech tří bodů, ale především akcentuje bod 3. Místo celých reprezentativních vět se soustředí pouze na nalezení nejčastěji komentovaných prvků. O způsobech, kterými je toho možné dosáhnout, pojednává sekce 2.2. Ještě předtím je ale potřeba daný text připravit (předzpracovat), jak popisuje následující sekce 2.1.

2.1 Předzpracování textu

Dolování dat v textu se liší od klasického dolování informací z databází tím, že vstupní data bývají nestrukturovaná. Může se jednat o články z novinářských portálů, obsáhlé dokumenty, příspěvky na blogu nebo na fórech a sociálních sítích. Prvním krokem v práci s nimi je odstranění případných formátovacích znaků a konverze na čistý text (plain text). Čistý text lze definovat jako sekvenci alfanumerických, interpunkčních, oddělovacích grafických znaků a některých speciálních symbolů (např. *, % apod) [2]. Převodem na čistý text se ovšem ztrácí případné metainformace, z nichž některé mohou být hodnotné – například členění textu, nadpisy kapitol apod. Tyto důležité informace ale lze také uložit, například ve formátu XML, a případně je využít v dalších fázích dolování v textu.

V dalších krocích předzpracování je možné text například segmentovat a tokenizovat, provést značení slovních druhů (Part of speech tagging), lemmatizaci nebo stematizaci. Dalšími užitečnými operacemi jsou odstranění „stop slov“, nahrazení synonym, oprava překlepů a mnohé další podle toho, co se právě hodí ke zvýšení efektivity konkrétní úlohy dolování v textu. Následující podkapitoly přináší přehled těch metod předzpracování, které jsou nejdůležitější pro tuto práci.

2.1.1 Segmentace a tokenizace

Velmi důležitým krokem v předzpracování textu je segmentace a tokenizace. Segmentací lze čistý text rozdělit na elementární jednotky textu. Těmi jsou souvislé řetězce alfanumerických znaků nebo jednotlivé interpunkční znaky [2].

* Dokumentem se zde rozumí ucelený text pojednávající o nějakém tématu– například jedna uživatelská recenze.

Například větu:

„Vím, co se stalo 28. 6. 1914.“

lze rozdělit na jedenáct elementárních jednotek textu:

[Vím] [,] [co] [se] [stalo] [28] [.] [6] [.] [1914] [.]

Předzpracování textu pak pokračuje tokenizací, která vytváří tokeny – skupiny znaků s definovaným, srozumitelným významem. Tokeny se identifikují pomocí slovníků přípustných tvarů slov a také použitím různých pravidlových systémů [2]. Předchozí věta by se mohla rozdělit na tokeny například takto:

[Vím] [,] [co] [se] [stalo] [28. 6. 1914] [.]

Segmenty tvořící „28. 6. 1914“ se zde spojily do jediného tokenu, protože posloupnost číslic a teček v takovémto formátu vyjadřuje datum – tedy skupinu znaků s konkrétním sémantickým významem.

Pro některé úlohy dolování v textu se navíc z identifikovaných tokenů vyberou jenom některé, které mají význam pro konkrétní operaci. Těm se pak říká termy. Pokud například operace nepotřebuje pro správné fungování znát interpunkci, pak se z ukázkové věty odstraní a ke zpracování zbyde pouze pět termů:

[Vím] [co] [se] [stalo] [28. 6. 1914]

Term je tedy klíčový prvek textu, který je tvořen jedním nebo více tokeny. Avšak ne z každého tokenu se stane term.

2.1.2 Part of Speech Tagging

Tato metoda (dále jako POS tagging) slouží k označení každého slova v textu jeho slovním druhem. K dalšímu zpracování do podoby termů se budou hodit primárně přídavná a podstatná jména, naopak lze většinou vynechat spojky, předložky nebo částice, které v textu mívají minimální informační hodnotu.

Jelikož jedno slovo může být v různých kontextech různého slovního druhu, nestačí porovnat jej se slovníkem. Je třeba při určování přihlédnout i k okolním slovům ve větě.

V češtině se často jako seznam tagů používají poziční morfologické značky od prof. Jana Hajiče[*].

2.1.3 Lemmatizace a stematizace

Lemmatizace je převedení slova na jeho základní slovníkový tvar (lemma) [2, s. 21]. Např. slovo „kočce“ se převede na „kočka“. Při práci s textem je toto předzpracování velice užitečné, protože

[*] Jejich seznam je dostupný na http://ucnk.korpus.cz/doc/popis_znacek.pdf.

dovoluje pracovat se slovem, které se v textu objeví ve více morfologických tvarech (pádech, číslech, osobách,…), jako s jediným slovem. Můžeme tak například snadněji spočítat výskyt tohoto slova apod. Pro podstatná a přídavná jména je lemma první pád jednotného čísla, pro slovesa infinitiv.

Stematizace (také stemming) je nalezení kmene slova [2, s. 21]. Zatímco výstupem lemmatizace je smysluplné slovo ve slovníkovém tvaru, stematizace původní slovo pouze „ořízne“ o případné předpony, přípony, koncovky. Slovo „kočce“ by se po provedení stematizace změnilo na „koč“. Každá z těchto metod, ač vypadají na první pohled podobně, má tedy zcela jiné výsledky (viz tabulka) a jsou využívány k jiným cílům.

Původní slovo	Lemmatizace	Stematizace
kočce	kočka	koč
řeší	řešit	řeš
vrchů	vrch	vrch
běžel	běhat	běž
Honzovi	Honza	Honz

Tabulka 1: Ukázka rozdílných výstupů lemmatizace a stematizace

2.1.4 Zpracování synonym

Pro různé operace s textem je výhodné, aby byla synonyma nahrazena jedním vybraným slovem. Například pokud je cílem spočítat všechny věty v dokumentu pojednávající o „procesoru“, měly by se zahrnout i takové věty, kde se vyskytuje slovo „CPU". K tomu může posloužit slovník synonym neboli tezaurus. Různé slovníky mají samozřejmě různá úskalí – často třeba vykazují nekompletnost v odborných oblastech či moderních slovech („displej“, „klikat“…).

Dalším problémem jsou slova, která nelze spolehlivě nahradit pouze pomocí slovníku, protože mohou mít různé významy. Např. „těžký“ může znamenat „náročný“, ale i „hodně vážící“. Nebo slovo „kočka“ má synonyma „šelma“, „kožešina“ a „dívka“.[*]

2.1.5 Odstranění stop slov

Stop slova jsou velmi frekventovaná slova, která však mají minimální význam pro sémantickou analýzu věty. Je tedy možné je odstranit, aniž by byl zásadně narušen nebo změněn význam textu. Jde především o krátká slova jako předložky a spojky („a“, „v“, „se“, „na“…), časté je také sloveso „být“ v různých tvarech.

Výběr stop slov může probíhat například na základě analýzy frekvence výskytu jednotlivých slov ve vhodném korpusu. Je ale vhodné také seznam stop slov zkontrolovat manuálně; i některá velmi často

[*] Seznam synonym vyhledaný v online slovníku: http://www.slovnik-synonym.cz/web.php/slovo/kocka].

se vyskytující slova mohou být důležitá pro smysl sdělení. Seznam stop slov je také samozřejmě značně ovlivněn vybraným korpusem. Mnoho korpusů sestává hlavně z novinových článků, takže do seznamu častých slov se mohou dostat i pojmy jako „koruna“, „rok“ nebo „strana“. [3]

2.2 Nalezení frekventovaných množin

Frekventovaná množina (*frequent itemset*) v kontextu dolování v textu je množina slov, která se často vyskytují v daném vzorku dat. Pojem byl poprvé použit v roce 1994 v [4]. V této vědecké publikaci je nalezení frekventovaných množin pouze mezikrokem k určení asociačních pravidel. Ta se používají k tzv. „analýze nákupního košíku“[*], tedy určení, jaké produkty se často prodávají společně, co dalšího si často zákazník pořídí, když si koupí např. chleba apod. Na základě těchto informací pak společnosti mohou přizpůsobovat svůj sortiment nebo lépe cílit reklamu.

Frekventované množiny ale mohou být užitečné i samy o sobě [5, s. 24]. Při zpracování uživatelských recenzí se hodí spočítat, které pojmy se často opakují, tedy o nich uživatelé hodně píší. K tomu lze využít Apriori algoritmus na nalezení frekventovaných množin, popsaný v další sekci.

2.2.1 Apriori algoritmus

Důležitými pojmy v Agrawalově algoritmu na hledání asociačních pravidel jsou podpora (*support*) a spolehlivost (*confidence*) [4]. Pro nalezení frekventovaných množin se používá jen podpora, definovaná jako pravděpodobnost, že transakce obsahuje položky X a Y. Požadovanou hodnotu podpory určí uživatel.

Apriori pracuje se souborem transakcí. V transakcích hledá slova, která se vyskytují aspoň n-krát (slova s podporou větší než n). Pak nalezená slova spojí do dvojic a udělá další průchod, kdy ověřuje, že daná dvojice se vyskytuje aspoň v n transakcích společně. Ty, co to splňují, se pokusí spojit do trojic a opět ověřit, zda se vyskytují alespoň v n transakcích společně atd. Všechny množiny, které mají podporu větší než n, označí jako frekventované množiny. Pro každou takovou množinu platí tzv. Apriori vlastnost: každá podmnožina frekventované množiny musí být frekventovaná.

Obvyklý problém základního algoritmu Apriori, použitého pro analýzu nákupního košíku s velkým množstvím dat, spočívá v četnosti přístupů do databáze transakcí a velkém počtu kandidátních frekventovaných množin. Tato zvýšená výpočetní zátěž se ale dá případně různými způsoby řešit.

[*] Pojem „Market basket analysis“: https://www.techopedia.com/definition/32063/market-basket-analysis

3 Analýza sentimentu

Analýza sentimentu (nebo také postojová analýza či dolování názorů) v kontextu dolování v textu si klade za cíl zjistit postoj autora textu k tomu, co popisuje. Nejzákladnější formou analýzy sentimentu je zjištění, zda věta vůbec nějaký sentiment obsahuje, či zda se jedná o faktické tvrzení. Pak se může provádět detekce polarity, tedy zda je sentiment (názorová orientace) textu pozitivní, či negativní. [6] Existují i pokročilejší metody detekce polarity, které se snaží sentiment řadit do více kategorií („spíše negativní“ apod.).

Analýzu sentimentu lze zařadit mezi úlohy klasifikace. Úkolem je zařadit daný dokument (či jinou část textu) do jedné z několika kategorií, zde například „pozitivní“ a „negativní“. Pro řešení tohoto typu úloh existují dva rozšířené přístupy: slovníkové metody a metody strojového učení.

3.1 Slovníkové metody

Slovníkové metody využívají předem připravené slovníky. Ty obsahují slova manuálně roztříděná a ohodnocená podle jejich sentimentu, vyjádřeného numerickou hodnotou. Podle této hodnoty se slova dělí do několika skupin [7]:

1) Pozitivní („krásné“, „blaženost“, „dobro“…)
2) Negativní („děs“, „lůza“, „zapáchající“…)
3) Neutrální („dům“, „provozní“, „řada“…)

Další kategorie nemají přímo určený sentiment, ale ovlivňují ho u jiných slov ve větě (ne nutně u přímo následujících):

4) Zdůrazňující význam slova – pozitivní i negativní („velmi“, „nepochybně“…)
5) Snižující význam slova („možná“, „sotva“, „jakž takž“)
6) Invertující význam slova („není“, „nemyslím si, že…“, „postrádá“…)

Slovníky mohou být vytvářeny manuálně nebo automaticky (pomocí „*seed words*“). Algoritmy založené na slovníkových metodách pak nějakým způsobem vypočítají sentiment věty či dokumentu na základě hodnot obsažených slov, nalezených ve slovnících. [8]

3.2 Klasifikace pomocí strojového učení

Klasifikace je jednou z úloh strojového učení (*machine learning*). Jejím úkolem je správně rozpoznat, do které třídy patří vstupní data. V základu jsou třídy dány předem uživatelem, existují i ale různé algoritmy pro nalezení vhodných tříd (*open-class clasification*) nebo třeba přiřazení různých tříd jednomu vzorku dat (*multi-class classification*). Ke klasifikaci se většinou používá strojové učení s učitelem.

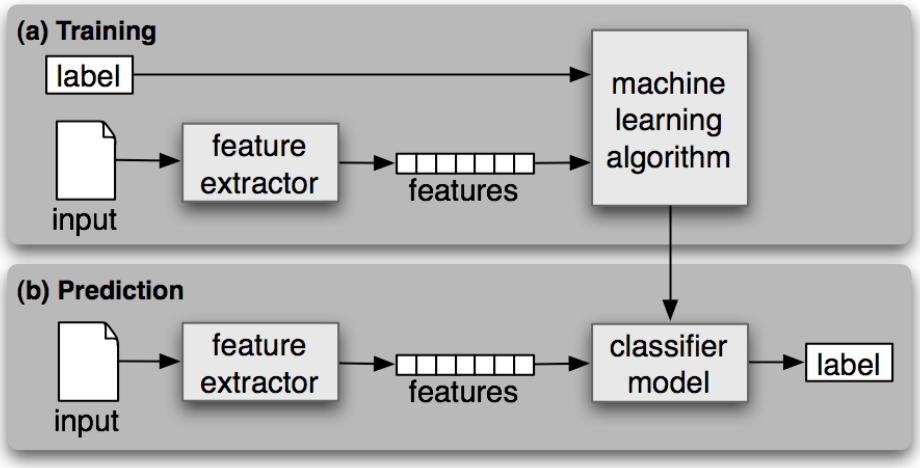
3.2.1 Učení s učitelem

Tento typ učení probíhá ve dvou krocích:

1) Naučení modelu na části předem označených dat (těm se také říká korpus).
2) Vyzkoušení klasifikátoru na jiné části korpusu a porovnání výsledků.

Pokud jsou výsledky dostatečně shodné s tím, jak byla data předem označena, „natrénovaný“ klasifikátor se může začít používat pro reálné aplikace.

Celý postup je dobře zřetelný z následujícího obrázku*:



Obrázek 1 Během fáze trénování jsou z dat extrahovány příznaky. Ty spolu s třídami slouží jako vstup pro algoritmus strojového učení. Vzniká klasifikátor (model). Ten pak podle příznaků vstupních dat určí třídu, do které vstup patří.

Pro dosažení relevantních výsledků je velmi důležité správně zvolit příznaky a jejich vhodnou reprezentaci. Metody strojového učení mohou dosáhnout významného zlepšení, pokud dostanou vektor příznaků, které jsou pečlivě vybrané například odborníkem v dané oblasti. [9] Typicky se ale začíná předložením všech možných příznaků a poté testováním a postupným výběrem těch, co se prokážou jako nejužitečnější. Pokud by příznaků zůstalo v konečné verzi příliš mnoho, mělo by to negativní dopad na dobu výpočtu. Navíc hrozí tzv. přeučení (*overfitting*), tedy přílišné přizpůsobení klasifikátoru konkrétním vstupním datům a následně jeho neschopnost správně určovat data nová.

V současné době jsou podle [10] pro zpracování přirozeného jazyka v textu rozšířené tři metody učení s učitelem:

1) Naive Bayes,
2) Maximum Entropy,
3) Support Vector Machines.

* Obrázek 1 převzat z: https://www.nltk.org/book/ch06.html

Autorka si s přihlédnutím k výsledkům srovnání uvedeného v [10] zvolila pro využití ve své práci první dvě metody, které budou představeny v následujících sekcích.

3.2.2 Naive Bayes

Klasifikátory typu Naive Bayes aplikují tzv. naivní přístup, tedy předpokládají, že výskyt každého slova či příznaku v dokumentu je zcela nezávislý na výskytech ostatních slov/příznaků. Tento přístup je výrazným zjednodušením skutečnosti, což se může projevit i na výsledcích klasifikace, na druhou stranu díky tomu Naive Bayes dosahuje nízkých časů učení a klasifikace.

Mějme dvě třídy: „Pozitivní“ a „Negativní“. Klasifikátor se snaží určit pravděpodobnost, že dokument *d* patří do jedné z tříd. Dokument je zařazen do třídy Pozitivní, pokud $p(Pozitivní|d) \geq 0{,}5$. Výpočet této pravděpodobnosti je založen na Bayesově teorému.

3.2.3 Maximum Entropy

Algoritmus Maximum Entropy využívá pro určení pravděpodobnosti $p(Pozitivní|d)$ více komplexní model, který se snaží reflektovat vztahy mezi jednotlivými příznaky. K tomu používá iterativní algoritmy, které na počátku určí jednotlivým třídám nějakou výchozí pravděpodobnost a poté v jednotlivých iteracích parametry modelu upravují a zvyšují tak přesnost modelu. Algoritmus však nemusí poznat, kdy dosáhl optimálního výsledku, proto je vhodné experimentálně ověřit počet iterací vhodný pro konkrétní zadání.

Nejčastěji používané iterativní algoritmy jsou [9]:

- Generalized Iterative Scaling (GIS)
- Improved Iterative Scaling (IIS)
- Conjugate Gradient (CG)
- Limited-memory BFGS (L-BFGS)

3.2.4 Vyhodnocování úspěšnosti

Aby se zjistilo, zda se klasifikátor úspěšně naučil vyhodnocovat nová data, je potřeba správným způsobem vyhodnotit jeho úspěšnost. Výsledky napoví, jak důvěryhodný je daný model, k čemu jej lze použít a případně jaká vylepšení ještě udělat.

3.2.4.1 Testovací data

Pro testování musí být k dispozici testovací data. Obvykle jde o předem vyhrazenou část původního korpusu. Data musí být dopředu označena správnou třídou, do které patří. Tato informace se pak porovná s odhadem klasifikátoru a zaznamenává se, kolikrát udělal správné rozhodnutí.

Testovací data samozřejmě musí být jiná než trénovací data, jinak nejde o ověřování schopnosti generalizace daného klasifikátoru. Poměrně často se ale stává, že označených dat není k dispozici dost

velké množství pro spolehlivé vyhodnocení. V tom případě se používá tzv. křížová validace (cross-validation). Z korpusu, který je k dispozici, se ve více iteracích oddělí pokaždé jiná malá část (podmnožina), která se nepoužije k trénování, ale k testování. Výsledky se pak zprůměrují. Tato technika zajistí spolehlivější vyhodnocování úspěšnosti i v případě, že jedna podmnožina by sama o sobě nestačila k přesnému určení přesnosti klasifikátoru.

3.2.4.2 Metriky

Nejjednodušší metrikou pro vyhodnocování úspěšnosti klasifikátoru je *accuracy*. Spočítá se tak, že počet správně zařazených prvků se vydělí počtem všech prvků testovací množiny. Tento způsob může najít uplatnění v hodnocení klasifikátorů, které pracují s vyváženými datasety. Jedná se o takový dataset, jehož prvky jsou rozloženy do jednotlivých tříd rovnoměrně (např. 50 % „pozitivní sentiment“, 50 % „negativní sentiment“). Čím víc se ale dataset od ideálního stavu odchyluje, tím nepřesnější se tato metrika stává. Mějme například dataset obsahující 95 % prvků označených „pozitivní“ a pouze 5 % prvků z třídy „negativní“. Pokud klasifikátor na tomto vzorku dosáhne accuracy 95 %, vypadá to jako skvělý výsledek, ale je nutné si uvědomit, že totéž mohl dosáhnout i klasifikátor, který by vše bez výjimky zařadil do třídy „pozitivní“. Obvykle je tedy pro získání více vypovídajících výsledků nutné použít sofistikovanější metriky – např. *precision* a *recall*. [11]

Obě metriky *precision* a *recall* se v rámci úloh klasifikace vztahují vždy k jedné klasifikační třídě. *Precision* znamená, kolik z prvků zařazených do dané třídy do této třídy skutečně patří. *Recall* znamená, kolik z prvků patřících do dané třídy bylo klasifikátorem skutečně do této třídy zařazeno. Mějme například klasifikátor operující s třídami „Pozitivní“ a „Negativní“. V tabulce jsou uvedeny všechny čtyři možné výsledky klasifikace.

	Pozitivní	Negativní
klasifikováno jako Pozitivní	*true positives*	*false positives*
klasifikováno jako Negativní	*false negatives*	*true negatives*

Metriky *precision* a *recall* pro třídu Pozitivní se vypočítají následovně:

precision = true positives / (true positives + false positives)
recall = true positives / (true positives + false negtives)

Tyto metriky pomáhají správně posoudit úspěšnost natrénovaného klasifikátoru. Existuje ještě tzv. *F-score,* které kombinuje *precision* a *recall* do jednoho čísla, které nicméně není tolik vypovídající.

Pro složitější klasifikační problémy s více třídami se používá tabulka nazývaná matice záměn (*confusion matrix*), kde buňka [*x*,*y*] značí, kolikrát byl prvek z třídy *x* zařazen do třídy *y*. Pokud bychom například chtěli v analýze sentimentu rozpoznávat ještě třetí třídu „Neutrální“, mohla by matice záměn pro nějaké měření vypadat třeba jako Tabulka 2.

	skutečně Pozitivní	skutečně Neutrální	skutečně Negativní
klas. jako Pozitivní	5	1	0
klas. jako Neutrální	4	3	2
Klas. jako Negativní	0	1	10

Tabulka 2: Příklad matice záměn.

Z matice lze např. vidět, že z 11 prvků, které klasifikátor označil jako Negativní, byly všechny kromě jednoho určeny správně. Problém naopak je, že klasifikátor zařadil správně pouze pět z devíti prvků patřících do třídy Pozitivní. Alespoň však žádný z nich neoznačil jako Negativní.

Z výše uvedeného je zřejmé, že vyhodnotit úspěšnost klasifikátoru není triviální problém. Vždy záleží na tom, k čemu má sloužit a která metrika bude pro uživatele nejdůležitější. Vezměme v úvahu například oblast zdravotnictví a aplikaci hledající v obraze případné nádory v lidském těle. V tomto případě bude rozhodně důležitější objevit všechny prvky ze třídy „Nádor“ (tedy vysoký *recall* pro třídu „Nádor“), a méně podstatné, jestli do této třídy chybně zařadí i některé prvky ze třídy „Zdravé“ (nižší *precision* třídy „Nádor“).

3.2.5 Učení bez učitele

Algoritmy strojového učení bez učitele (*unsupervised learning*) mají k dispozici pouze neoznačená data. Jejich účelem je najít samostatně podobnosti a odlišnosti mezi jednotlivými vzorky, odvodit vhodné kategorie a vzorky do nich rozdělit. Tomuto úkolu se také říká shlukování (*clustering*).

Příkladem metody využívající principů učení bez učitele je Turneyho třístupňový algoritmus [12]. Ten vypočítá sémantickou orientaci fráze jako: společné příznaky se slovem „excellent“ minus společné příznaky se slovem „poor“. Dokument je označen jako pozitivní, pokud průměrná sémantická orientace jeho frází je pozitivní [12].

4 Současná řešení

Zpracováním uživatelských recenzí se výzkumníci začali zabývat v minulém desetiletí (např. [13], [14], [12]). Proto je možné na toto téma najít hned několik prací, většina se ovšem zabývá zpracováním anglického jazyka. V této kapitole budou představeny některé z nich včetně stručného popisu postupu, který je na problém aplikován. Podkapitoly jsou děleny podle autorů těchto článků.

4.1 Metoda využívající Apriori a Wordnet

Článek *Mining Opinion Features in Customer Reviews* [13] se zabývá sumarizací názorů zákazníků na produkty prodávané online. Systém pracuje ve dvou krocích: nalezení komentovaných vlastností produktu a určení názorové orientace.

Autoři používají POS tagging k identifikaci podstatných jmen a jmenných frází. Ty pak projdou preprocessingem (odstranění stop slov, stematizace a fuzzy matching) a jsou uloženy do transakčního souboru. Následně jsou nalezeny frekventované množiny pomocí Agrawalova Apriori algoritmu a ořezávání chybně pozitivních vzorků (false positives). Výsledkem jsou nejčastěji komentované vlastnosti produktů. [13]

Další článek s názvem *Mining and Summarizing Customer Reviews* [14] je rozšířením předchozí práce autorů [13] a věnuje se především oblasti rozpoznávání a klasifikaci názorů zákazníků. Autoři nejdříve najdou frekventované vlastnosti (viz výše) a poté určují sentiment jednotlivých vět pomocí přídavných jmen před nalezenými vlastnostmi. Ke klasifikaci sentimentu adjektiva používají systém Wordnet.

4.2 SumView – systém pro sumarizaci recenzí

Práce této trojice s názvem *SumView – A Web-based engine for summarizing product reviews and customer opinions* představuje webový systém SumView. Jedná se o systém pro sumarizaci uživatelských recenzí. Automaticky nalezne nejvíce reprezentativní vyjádření zákazníků k různým vlastnostem produktu.

Nejprve stáhne všechny recenze zadaného produktu z Amazonu. Rozdělí je do vět a pomocí POS tagging označí každé slovo. Po odstranění stop slov je sestavena matice, kde každý řádek reprezentuje term a každý sloupec reprezentuje větu. Jsou nalezeny vlastnosti produktu, podobným způsobem jako v [14]. Pět nejčastějších vlastností je nabídnuto uživateli. Ten si z nich může vybrat, případně zadat jiné. Na základě vybraných vlastností provede systém algoritmus *feature-based weighted non-negative matrix factorization*, podle kterého se věty seskupí do clusterů podle obsažených produktových vlastností. Z každého clusteru je potom pro uživatele vybrána nejreprezentativnější věta. [15].

4.3 Využití LDA a učení bez učitele

Tento výzkum (*Summarizing Customer Reviews Based On Product Features*) se zaměřuje na extrakci a sumarizaci názoru, vyjádřeného v uživatelské recenzi, a jeho síly. Autoři kombinují LDA (Latent Dirichlet allocation) model a asociační pravidla, aby vybrali produktové vlastnosti a korespondující subjektivně zabarvená slova. Pomocí učení bez učitele pak těmto slovům spočítají sílu sentimentu. Oproti Hu a Liu [14] jsou schopni nalézt i „implicitní vlastnosti“, které nejsou v recenzích napsané přímo. [16] Recenze, které používají jako jejich dataset, jsou uživateli označena pomocí jedné až pěti hvězdiček.

Jejich systém pracuje v těchto krocích:

1) Identifikace všech produktových vlastností zmiňovaných v uživatelských recenzích.
2) Extrakce subjektivně zabarvených slov ovlivňující nalezené produktové vlastnosti.
3) Určení síly sentimentu slova na stupnici „strong“-„neutral“-„weak“.
4) Vygenerování stručného souhrnu (kolik názorů na vlastnost produktu bylo silně pozitivních apod.).

Zvláštní jsou poněkud matoucí kategorie výsledného souhrnu; kategorie „Positive“ a „Negative“ se každá dělí ještě na „Strong“, „Neutral“ a „Weak“, takže ve výsledku jsou některé názory zařazeny např. jako „negative neutral“.

5 Implementace

Tato kapitola nejprve v sekci 5.1 popisuje návrh aplikace, včetně Data Flow diagramu zobrazujícího průběh algoritmu a předávání dat mezi jednotlivými komponentami. Dále představuje dostupné nástroje, které by mohly být pro implementaci využity.

Sekce 5.2 pak v několika na sebe navazujících celcích obsahuje podrobný popis implementace a fungování všech vytvořených komponent a nástrojů.

5.1 Návrh aplikace

Podle zadání měl implementovaný nástroj zpracovávat uživatelské recenze produktů v českém jazyce, nalézt v textu nejvíce komentované vlastnosti produktu a určit, zda je sentiment komentáře pozitivní nebo negativní. Tento úkol autorka dekomponovala na jednotlivé dílčí části, které jsou následující:

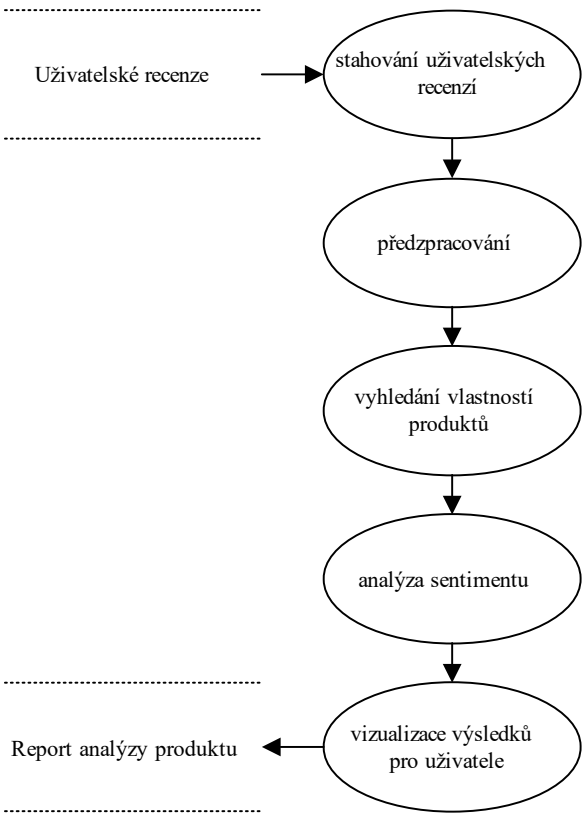
1) stahování uživatelských recenzí,
2) předzpracování,
3) vyhledání vlastností produktů,
4) analýza sentimentu,
5) vizualizace výsledků pro uživatele.

Aby bylo možné tyto úkoly řešit co nejvíce nezávisle na sobě, byla řešení jednotlivých částí navržena jako samostatné nástroje nebo komponenty, které na sebe navazují jasně definovanými vstupy a výstupy.

Vstupem aplikace jsou recenze stažené z internetu. Pokud uživatel nemá připravené vlastní, může použít implementovaný nástroj na stahování recenzí z Heureky[*]. Získané soubory nejprve projdou předzpracováním, zahrnujícím tokenizaci, POS tagging, lemmatizaci a odstranění stop slov.

Z kontextu komentářů obsahujících tyto vlastnosti se pak zjistí sentiment s nimi spojený. Aplikace tak bude schopna jako výstup poskytnout seznam dobrých a špatných vlastností produktu, které se nejčastěji objevily v uživatelských recenzích. Navrhovaný průběh práce aplikace je vidět na Obrázek 2.

[*] https://www.heureka.cz/



Obrázek 2: Návrh konceptu aplikace jako Data Flow diagram.

Pro implementaci aplikace byl vybrán programovací jazyk Python, protože jde o osvědčený nástroj pro efektivní a rychlé prototypování. Zároveň nabízí rozsáhlou open-source knihovnu pro práci s přirozeným jazykem. NLTK (Natural Language Toolkit) poskytuje snadno použitelná rozhraní k více než padesáti korpusům a lexikálním zdrojům (například WordNet). Součástí NLTK jsou knihovny nabízející metody pro klasifikaci, tokenizaci, stematizaci, značkování apod.[*] Některé jsou užitečné i pro zpracování českého jazyka, jiné jsou vhodné pouze pro angličtinu či jiné rozšířené jazyky.

Pro POS tagging v češtině byl na Univerzitě Karlově vyvinut nástroj Morče[†], dostupný pod GNU General public licencí. Pro účely lemmatizace lze zase využít například morfologický analyzátor Majka[‡] publikovaný na Masarykově univerzitě také pod GPL licencí. Dokonce oběma funkcionalitami pak disponuje nástroj MorphoDiTa z Univerzity Karlovy. Velkou výhodou MorphoDiTy je také dobře použitelné rozhraní pro Python.

Dalším důležitým zdrojem jsou slovníky pro různé druhy zpracování textu. Například pro určování sentimentu slov se může hodit slovník Czech SubLex [17]. Obsahuje 4626 položek, jejich POS

[*] Další informace dostupné na: https://www.nltk.org/
[†] POS tagger českého jazyka Morče: http://hdl.handle.net/11858/00-097C-0000-0001-48FE-9
[‡] Morfologický analyzátor českého (a dalších) jazyků Majka: https://nlp.fi.muni.cz/ma/

tagy a polaritu. Základ slovníku byl získán překladem z již existujícího anglického slovníku. Zkušený anotátor pak dílo ručně zkontroloval, opravil a vylepšil. Pro vyhledávání a nahrazování synonym lze zase využít český tezaurus dostupný v rámci LibreOffice[*].

5.2 Implementace aplikace

Pro implementaci byl použit jazyk Python verze 3.7.2 a několik balíčků, jejichž seznam lze najít v souboru *requirements.txt*. Práce byla rozdělena na několik částí:

- stahování recenzí,
- předzpracování,
- nalezení častých témat recenzí,
- analýza sentimentu vět,
- grafický výstup pro uživatele.

Tyto části budou podrobně popsány v následujících sekcích.

5.2.1 Skript pro stahování recenzí

Pro učení klasifikátoru i vyhodnocení úspěšnosti celé aplikace bylo třeba vytvořit vhodný dataset (důvody a nároky na něj jsou popsány v kapitole 6.1). Součástí této práce je tedy i vlastní skript na získávání a formátování recenzí produktů ze serveru heureka.cz. Na Heurece má každý produkt vlastní stránku a na ní mj. sekci Recenze. Ty mohou psát jak lidé, o kterých Heureka ví, že si daný produkt koupili, tak anonymové. Hodnocení lze podle toho i filtrovat. Recenze se skládá z „plusů“ – bodového seznamu silných stránek – a „mínusů“ – seznamu slabin. Dále může obsahovat další text a také celkové hodnocení „doporučuji“/“nedoporučuji“. Tyto informace je nutné uchovat spolu s texty pro účely klasifikace a pozdější vyhodnocování úspěšnosti. Recenze může mít i další složky, které ale nejsou podstatné pro implementovanou aplikaci, a nemusí se tedy ukládat.

Před samotnou implementací byly definovány požadavky na funkcionalitu tohoto nástroje pro získávání recenzní produktů následovně. V nástroji má být možné určit konkrétní produkt ze serveru Heureka, jehož recenze budou zpracovány. Následně má být schopen plně automatizovaně analyzovat webový obsah příslušící danému produktu a identifikovat recenze od uživatelů. Texty recenzí má pak nástroj připravit do formátu vhodného pro další analýzu implementovanou aplikací. Na základě výše definovaných požadavků byla práce skriptu pro stahování recenzí rozdělena do následujících základních kroků:

[*] LibreOffice český slovník synonym: https://github.com/LibreOffice/dictionaries/tree/master/cs_CZ/thesaurus

1) Načtení seznamu identifikátorů produktů, jejichž recenze mají být získány.
2) Stažení webového obsahu, který prezentuje určené produkty, do souboru na disku.
3) Extrakce uživatelských recenzí a relevantních informací z odpovídajících souborů.
4) Transformace datové struktury a zápis ve formátu vhodném pro následnou analýzu implementovanou aplikací.
5) Vhodná organizace výsledných dat v adresářové struktuře.

Tento nástroj byl následně označen jako *reviewscraper* a implementován v jazyce Python. Výsledný skript je k dispozici ve složce *reviewscraper* jako `main.py`. Jedná se tedy o program v prostředí příkazové řádky. Nástroj po spuštění informuje o prováděných krocích a výsledcích akcí prostřednictví logování s nastavitelnou úrovní výpisu.

V úvodní části skriptu v konfigurační části se nachází kolekce `heureka_products`, ve které je možné definovat seznam produktů, pro které má nástroj získat a zpracovat recenze. K tomuto účelu slouží implementovaná třída `Product`, která při instanciaci vyžaduje předání:

- URL domovské stránky produktu na serveru Heureka,
- Textového identifikátoru produktu,
- Počtu stránek recenzní pro zpracování.

Stažení webového obsahu na disk zajišťuje funkce `download_heureka`. Na základě určení konkrétního produktu předáním instance třídy `Product` tato funkce projde odpovídající stránky na serveru Heureka a uloží je na disk jako HTML soubory. Funkce využívá zejména funkcionalitu Python balíčku `requests`. Stažené soubory jsou na disku organizovány do adresářů podle textového identifikátoru příslušeného produktu.

Extrakce informací ze stažených HTML souborů je zodpovědnost funkce `extract_heureka`. Funkce využívá funkcionality Python balíčku `BeautifulSoup`, který umožňuje parsování HTML dokumentů, efektivní navigaci mezi elementy a extrakci informací. Byla analyzována struktura HTML dokumentů s uživatelskými recenzemi na serveru Heureka. Na základě zjištění byly vytvořeny odpovídající selektory pro získání obsahu pozitivních bodů a negativních bodů každé recenze. Recenze dále obsahuje volný text, který byl také extrahován. Povaha volného textu recenze byla určena na základě elementu, který recenzi celkově shrnuje jako doporučující nebo nedoporučující.

Extrahované informace ze všech analyzovaných recenzí daného produktu byly akumulovány a následně uloženy na disk do zmíněných adresářů podle identifikace produktu. Výsledkem nástroje *reviewscraper* je tedy připravený dataset organizovaný do adresářové struktury podle jednotlivých produktů, kdy u každého produktu jsou kromě původních HTML souborů k dispozici následující zpracovaná data:

- <*product-name*>_negative_points.txt (soubor mínusů oddělených novými řádky)
- <*product-name*>_negative_texts.txt (soubor textů z recenzí hodnocených „nedoporučuji“)
- <*product-name*>_positive_points.txt (soubor plusů oddělených novými řádky)

- <*product-name*>_positive_texts.txt (soubor textů z recenzí hodnocených „doporučuji“)

Tyto soubory pak slouží jako vstup implementované aplikace.

5.2.2 Předzpracování

Pro velkou část úloh předzpracování byl použit open-source nástroj pro morfologickou analýzu přirozeného jazyka MorphoDiTa (Morphological Dictionary and Tagger) [18]. Tento software vytvořený na Ústavu formální a aplikované lingvistiky na Univerzitě Karlově v Praze se velmi zdařile vypořádává se specifickými překážkami provázejícími zpracování českého jazyka. Je schopen text tokenizovat, provést POS tagging a lemmatizaci. Software je distribuován včetně natrénovaných lingvistických modelů pod licencí Mozilla Public Licence 2.0.

V implementované aplikaci se předzpracování recenzí stažených pomocí skriptu (viz kapitola 5.2.1) zahajuje úpravou všech vstupních dat do jednotné podoby. Protože každý zákazník má jiný způsob psaní hodnocení (bodové seznamy vs. celé věty…), je potřeba vše sjednotit, aby bylo možné další automatizované zpracování. V rámci tohoto sjednocování je odstraněna nadbytečná interpunkce a další znaky, které netvoří slova (úvodní nebo ukončující mezery, tři tečky, vykřičníky…). Je dobré si uvědomit, že tímto postupem se odstraní i případné emotikony, které již mohou nést nějaký sentiment, ačkoli nejde o slova. Implementovaná aplikace s emotikonami nepracuje, ale mohlo by jít o vylepšení do budoucnosti.

Následně je každý řádek upraven na větu, aby začínal velkým písmenem a končil tečkou. Díky tomu jsou data připravena pro tokenizaci, kterou zajišťuje MorphoDiTa. Každá věta je rozdělena na slova. Slovo je uloženo jako objekt typu `Word`, spolu se svým lemmatem a řetězcem obsahujícím morfologické značky (POS tagy). Seznam slov jedné věty vytvoří objekt typu `ReviewSentence`. Zároveň ale slovo uchovává slabou referenci na všechny věty, ve kterých se vyskytuje.

Hned při vzniku objektu `Word` je také adresován problém příliš důkladné lemmatizace, která převádí negativní tvary slov na pozitivní (např. „nevkusnými“ na „vkusný“). To by rozhodně nebylo vhodné pro analýzu sentimentu. Naštěstí je informace o tom, zda byla původně ve slově přítomna negace, uchována jako morfologická značka, můžeme tedy případně předponu „ne-“ snadno připojit ke slovu zpátky.

Po vytvoření objektů `ReviewSentence` se přistoupí k odstranění stop slov. Všechna slova věty jsou porovnána se seznamem stop slov a pokud se jejich původní tvar nebo lemma shodují s některým záznamem, je celý objekt `Word` odstraněn. Slovo zůstane zaznamenáno pouze v atributu `raw_sentence` objektu `ReviewSentence`, který slouží pro uživatelsky srozumitelný výpis věty, ale v algoritmu se s ním jinak dále nepracuje. Seznam stop slov použitých v implementované aplikaci si lze prohlédnout v příloze 9.

V původním návrhu aplikace zahrnovala fáze předzpracování navíc nahrazení synonym. Na začátku vývoje bylo tudíž provedeno několik experimentů s českým slovníkem synonym LibreOffice.

Tyto pokusy bohužel nevykazovaly použitelné výsledky. V některých případech byla synonyma nahrazována příliš volně (např. „kočka“ za „dívka“). Také ale ve slovníku chyběla alespoň základní slova technického zaměření, takže např. slova „displej“ a „obrazovka“ nebo „mobil“ a „telefon“ zůstala odděleně. Řešením by mohlo být vytvořit si vlastní přizpůsobený slovník. Takové vylepšení ovšem nebylo oproti jiným vyhodnoceno jako tak velký přínos, zůstává tudíž možným rozšířením práce do budoucna.

5.2.3 Nalezení vlastností produktů

První předpoklad, který si autorka určila pro hledání vlastností produktů byl takový, že vlastnosti produktu budou vždy podstatná jména. První prototyp této funkcionality tedy pracoval tak, že vybral z recenzí všechna podstatná jména, seřadil je od nejčastějšího k nejméně častému a vrátil požadovaný počet slov s nejčastějším výskytem u daného produktu.

Tento přístup se ukázal jako poměrně efektivní, ale také odhalil další skutečnost, kterou bylo nutno vzít v úvahu. Ne všechna podstatná jména, která se objevují v uživatelských recenzích, jsou totiž zároveň vlastnostmi produktu. Velké zastoupení ve výsledcích mají i některá velmi obecná slova (trh, problém, spokojenost) a také slova specifická pro daný kontext, ale bez nějakého přínosu (telefon, mobil). Pro takové pojmy je obtížné zavést jednoznačná všeobecně platná pravidla, která by z výsledků odfiltrovala méně smysluplná témata. Uživatel si ale tato slova, která chce z analýzy vynechat, může zvolit sám pomocí úpravy seznamu stop slov. Nicméně i pokud ve výstupech zůstanou, stále mohou uživateli poskytnout zajímavé doplňující informace díky přidruženým větám. Z těchto důvodů byl původní záměr nalezení komentovaných vlastností produktů posunut na nalezení důležitých témat, ke kterým se zákazníci nejčastěji vyjadřují.

V prvních verzích se také často mezi vybranými pojmy opakoval i název produktu či značky („Samsung“, „Galaxy“), tomu se však předešlo přidáním slov z názvu produktu mezi stop slova. Dále se občas vyskytují i názvy jiných značek či modelů (např. „iPhone“ v recenzích Samsungu). Ty mají naopak v analýze své místo; porovnání s obdobným modelem od jiné značky může být pro potenciálního zákazníka velmi zajímavé.

5.2.4 Analýza sentimentu pomocí slovníku

Tato sekce popisuje první, přímočařejší přístup k určování sentimentu – metodu využívající slovník. V implementaci byl použit slovník Czech Sublex 1.0 [17], který obsahuje 4626 záznamů včetně jejich polarity. Tento slovník nejdříve musíme načíst ze souboru a převést ho na objekt typu `dictionary`. Poté se vyberou data k analýze a podrobí se předzpracování a transformaci do seznamu objektů ReviewSentence.

Z těchto vět se pak vyberou nejčastější témata (viz 5.2.3) a pro každé z nich se zjišťuje jeho skóre sentimentu. K tomu je třeba určit kategorii sentimentu každé věty, ve které se téma vyskytuje. Zde

přichází na řadu slovník – pokud se ve větě vyskytne slovo, které je ve slovníku označeno sentimentem ‚pos', přičteme větě jeden bod; pokud má slovo záznam ‚neg', jeden bod větě odečteme. Když je skóre celé věty kladné, zařadíme větu jako pozitivní a navýšíme (vždy o jedničku) i skóre tématu, v opačném případě je věta negativní a skóre tématu se sníží. Tímto způsobem se nakonec dostaneme k výslednému skóre, které najde uplatnění například ve výstupní analýze vygenerované pro uživatele (viz 6.3).

5.2.5 Analýza sentimentu pomocí strojového učení

Pro tento způsob určování sentimentu je potřeba nejprve vybrat vhodnou metodu. Na základě informací uvedených v sekci 3.2.1 byly zvoleny k porovnání metody Naive Bayes a Maximum Entropy. Podle výsledků experimentů (viz 6.2) se nakonec autorka rozhodla zařadit do implementace klasifikátor Maximum Entropy.

Prvním krokem ke klasifikaci je naučit model rozpoznávat vzory na trénovacích datech. Pro transformaci předzpracovaných vstupních dat do podoby vektorů příznaků byla implementována funkce `prepare_data_for_classifier`. Ta nabízí několik možných algoritmů pro získání příznaků. Volba „*true-and-false*" znamená vytvoření vektoru například o několika stech slovech (počet je určen vstupním parametrem funkce) z vybraných slov z testovacích dat. Každé z těchto slov se v konkrétním vektoru příznaků označí *true* nebo *false* podle toho, zda se vyskytuje v právě zpracovávané větě.

Z experimentů (viz sekce 6.2.3) se zjistilo, že tento přístup je vhodný pro klasifikátor typu Naive Bayes, nikoli však pro Maximum Entropy. Bylo ověřeno, že v tomto případě mnohem lépe vyhovuje jednodušší volba „*true-only*". Ta do jednotlivých vektorů příznaků zařadí právě jen ta slova, která jsou součástí nyní zpracovávané věty.

Když jsou připravená trénovací data v podobě vektorů příznaků, můžeme již téměř přistoupit k samotnému učení modelu. Stačí jen vybrat iterativní algoritmus (zde IIS, který je již součástí implementace klasifikátoru Maximum Entropy v NLTK) a počet iterací. Měření prokázalo, že v našem případě je optimální volbou 50 iterací. S těmito parametry zavoláme metodu `nltk.MaxentClassifier.train()` a po nějakém čase (závislém na velikosti vstupních dat a zvolených parametrech) dostaneme na výstupu natrénovaný klasifikátor.

Obdobným způsobem se připraví také data, která má klasifikátor vyhodnotit – tedy recenze jednoho produktu. Algoritmem pro výběr nejčastějších témat se určí témata k vyhodnocení a ta se předají, spolu s odkazem na klasifikátor, implementované funkci `analyze_themes()`.

Funkce nechá naučený model klasifikovat každou větu každého tématu a v závislosti na sentimentu jednotlivých vět upravuje skóre sentimentu celého tématu. Nakonec vytvoří a vrátí objekt typu `ThemeAnalysis`. Ten obsahuje název tématu, jeho skóre, všechny kladné a záporné věty ve kterých se téma vyskytuje a výsledné označení „pozitivní", nebo „negativní".

5.2.6 Report výsledků analýzy

Aby bylo možné výsledky analýzy prezentovat uživateli ve srozumitelné a dále jednoduše uživatelem zpracovatelné formě, implementovaná aplikace zahrnuje také funkcionalitu pro vytváření přehledných dokumentů s reporty výsledků. Tento záměr realizuje třída `AnalysisReport` v rámci modulu *presenter.py*. Pro vytvoření reportu je třeba předat následující informace:

- název produktu a URL domovské stránky produktu na serveru Heureka,
- nejčastěji diskutovaná témata,
- graf nejvíce diskutovaných témat,
- nejvíce pozitivní témata,
- nejvíce negativní témata.

Třída `AnalysisReport` byla implementována s využitím funkcionality Python balíčku `python-docx`. Při instanciaci je vytvořena struktura dokumentu. Z předaných dat z objektů `ThemeAnalysis` jsou pak získány výsledky analýzy. Výsledná data jsou následně začleněna do připraveného dokumentu. Instance implementované třídy `AnalysisReport` ve výsledku umožňuje uložení reportu na disk ve formátu *.docx*.

Uživatel implementované aplikace tak může analyzovat recenze produktu a výsledek získává ve formě srozumitelného dokumentu. Díky použití rozšířeného formátu *.docx* si uživatel může například velmi snadno výsledný report dále doplnit o vlastní poznámky k produktu a uložit nebo vytisknout.

6 Experimenty

Tato kapitola se zaměřuje na popis experimentů prováděných s implementovanou aplikací. Pro účely předzpracování dat a analýzy sentimentu bylo implementováno několik různých metod. Součástí výsledné aplikace se nakonec stala metoda Maximum Entropy, a to na základě následujících srovnávacích experimentů. Nejprve byly porovnány metody strojového učení s učitelem Naive Bayes a Maximum Entropy. Poté se autorka zaměřila na vstupní data klasifikace: jaká míra předzpracování a výběru příznaků vede k nejlepším výsledkům? Dále byla vyhodnocena úspěšnost jednoduché metody analýzy sentimentu pomocí slovníku.

Nejefektivnější metoda vybraná na základě těchto experimentů pak byla podrobena dalším testům a výsledky byly vyhodnoceny v sekci 6.3.

6.1 Dataset

Pro záměry této práce bylo potřeba získat dostatečné množství dat, na kterých by bylo možné otestovat vytvořené řešení. Na dataset byly kladeny následující požadavky:

1) Český jazyk,
2) Označení sentimentu vět (recenzí),
3) Dostatečné množství recenzí pro natrénování klasifikátoru,
4) Snadná dostupnost a možnost využití pro nekomerční účely.

Jelikož bylo obtížné najít dataset, který by vyhovoval všem podmínkám, jako nejúčinnější řešení se ukázalo implementovat vlastní skript pro stahování recenzí (viz sekce 5.2.1). Pomocí něj byla získána data ze serveru Heureka[*]. Pro experimenty byly vybrány produkty z kategorie Mobilní telefony, protože jde o oblast s velkým množstvím komentářů, včetně negativních, což není samozřejmé u všech produktů. Recenze obsahují „plusy“ a „mínusy“, tedy krátké body s jasně určeným sentimentem, a někdy také delší text s označením „doporučuji“ nebo „nedoporučuji“. Sloučením všech těchto vět dohromady vznikne následující dataset:

	Počet vět
Pozitivní	3288
Negativní	1051
Celkem	**4339**

Tabulka 3: Složení a velikost vytvořeného datasetu používaného k experimentům.

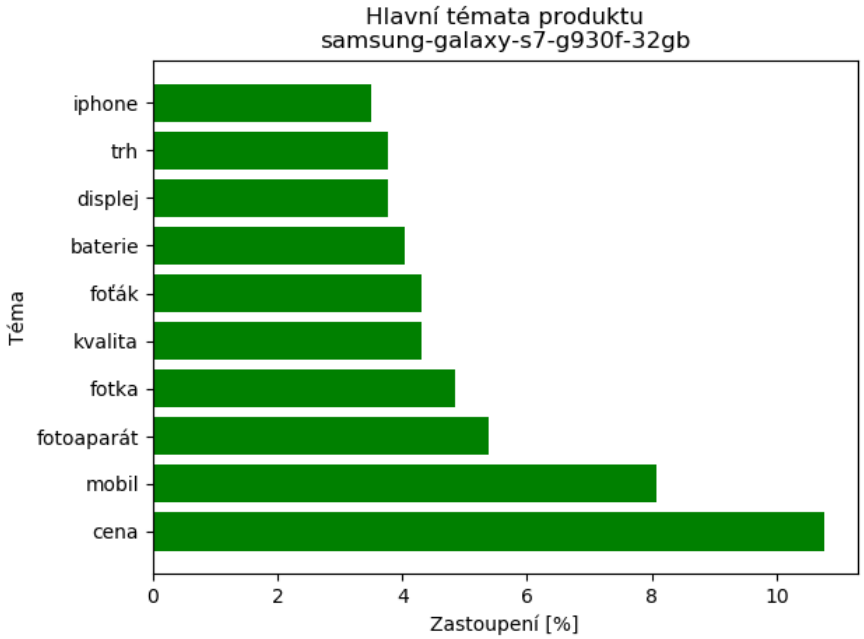
[*] https://www.heureka.cz/

6.2 Nalezení nejefektivnějšího přístupu

Závěrečné verzi aplikace předcházelo mnoho experimentů, které měly za úkol porovnat a zhodnotit různé možné přístupy. Tato sekce se zabývá výběrem nejvhodnější metody klasifikace. Pomocí několika experimentů nejdříve srovnává čas učení jednotlivých metod (Naive Bayes a Maximum Entropy). Poté představuje porovnání efektivity těchto dvou metod a také slovníkové metody s přihlédnutím k metrikám *precision* a *recall*.

6.2.1 Hledání témat a vlastností produktu

Důležitým úkolem bylo zjistit z recenzí témata, o kterých uživatelé nejčastěji píší. Výsledky příslušného algoritmu pro recenze mobilního telefonu Samsung Galaxy S7 jsou zobrazeny v Graf 1.



Graf 1: Hlavní témata produktu Samsung Galaxy S7 a jejich zastoupení mezi všemi podstatnými jmény v recenzích na vstupu.

Z grafu vidíme, že mezi nejčastějšími podstatnými jmény v recenzích produktu převažují vlastnosti produktu (cena, fotoaparát...). Navzdory některým dílčím nedostatkům (např. „foťák“ a „fotoaparát“ jsou uvedeny jako rozdílná témata) jsou výsledky uspokojivé a mají místo ve výsledné aplikaci. Více informací o implementaci, řešení specifických problémů a možných vylepšení do budoucna se lze dočíst v 5.2.3. Podrobné zhodnocení funkcionality hledání témat je pak uvedeno v sekci 6.3.1.

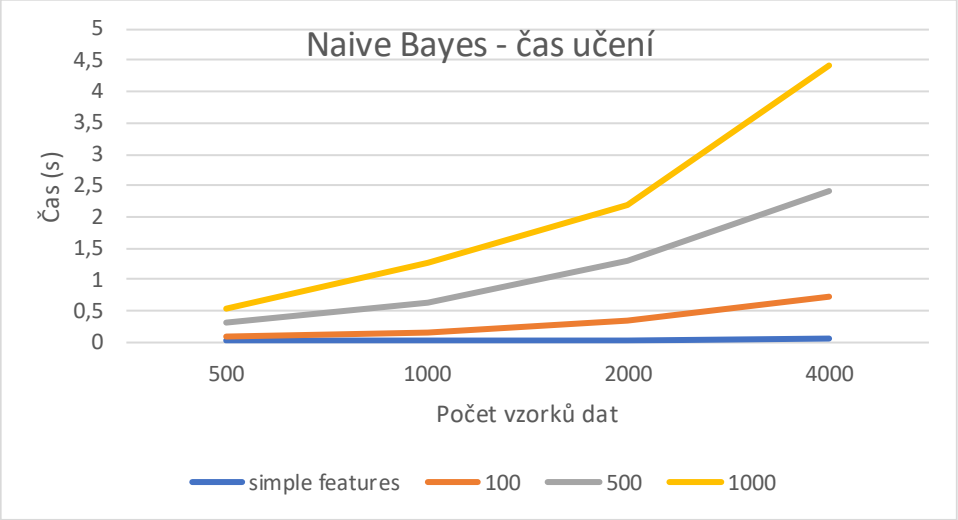
6.2.2 Čas učení modelu

Jedním z kritérií výběru metody klasifikace může být časová náročnost trénování modelu. Následující experiment byl proveden za účelem zjištění, zda existují významné rozdíly mezi časy potřebnými k naučení modelů typu Naive Bayes a Maximum Entropy.

Vstupní data byla nejprve podrobena předzpracování, jež zahrnovalo tokenizaci, lemmatizaci, POS tagging a odstranění stop slov. Poté byla převedena na vektory příznaků. Protože počet příznaků může velmi ovlivnit celkovou dobu učení (a později klasifikace), byly otestovány následující možnosti:

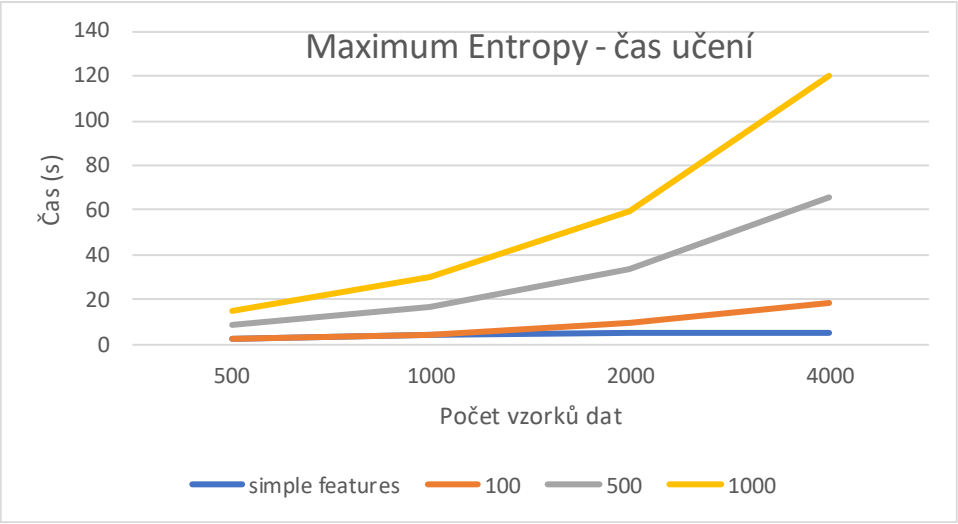
1) Příznaky jsou všechna slova vyskytující se v daném dokumentu (větě recenze).
2) Vektor příznaků obsahuje 100, 500 nebo 1000 vybraných slov a ke každému z nich booleovskou hodnotu, která značí, jestli se dané slovo v dokumentu vyskytuje, nebo nevyskytuje.

Naměřené hodnoty byly vyneseny do grafu pro 500, 1000, 2000 a 4000 vzorků dat. Každé měření bylo provedeno pětkrát, v grafech se objevuje průměr těchto časů. Čas výpočtu byl měřen pomocí funkce `perf_counter()` modulu `time`. Níže jsou vidět podrobné grafy pro metodu Naive Bayes a Maximum Entropy.



Graf 2: Čas učení modelu Naive Bayes v závislosti na počtu vzorků dat a velikosti vektoru příznaků.

Z Graf 2 můžeme vidět, že Naive Bayes dosahuje při učení velmi rozumných časů. I při velikosti vektoru příznaků 1000 narůstá čas s množstvím vstupních dat poměrně pomalu.



Graf 3: Čas učení modelu Maximum Entropy v závislosti na počtu vzorků dat a velikosti vektoru příznaků.

Maximum Entropy vychází z tohoto srovnání hůře. Jak je vidět v Graf 3, s většími vektory příznaků se dostáváme na desítky sekund už při relativně malém množství vstupních dat. Při 4000 vzorků dat a s vektorem o tisíci příznacích je Maximum Entropy při trénování modelu asi 26x pomalejší než Naive Bayes. Pro učení byl použit iterační algoritmus Improved Iterative Scaling (IIS). Pro dosažení uvedených výsledků bylo použito pouze 10 iterací. Při vyšším počtu lze očekávat ještě další výrazný nárust času.

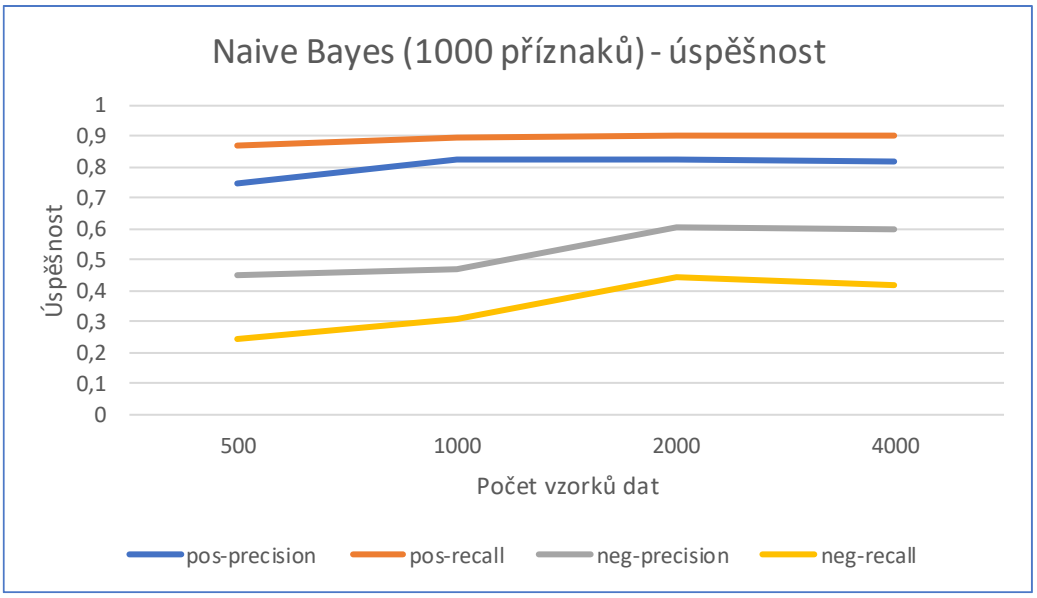
Otázka ovšem zní, zda Maximum Entropy takové množství příznaků potřebuje. Dalšími experimenty bylo zjištěno, že vektory obsahující „true“ nebo „false“ pro n vybraných slov (které dobře fungují pro klasifikátor Naive Bayes) jsou v praxi pro Maximum Entropy nepoužitelné. Mnohem lépe si poradí s typem vektoru nazvaným v grafu „simple features“, který obsahuje maximálně malé desítky slov. Pokud vezmeme v úvahu jenom tento typ vektoru, je čas potřebný k učení (při deseti iteracích) již srovnatelný s Naive Bayes.

Výše uvedená čísla se týkají pouze času potřebného k vykonání funkce `train()`. Pokud bychom ovšem chtěli znát celkový čas nezbytný k získání naučeného modelu, musíme započítat i transformaci vstupních dat do vektorů příznaků. Trvání tohoto procesu bude závislé opět na počtu vzorků vstupních dat a velikosti vektoru příznaků, jak můžeme pozorovat v Graf 9 v příloze.

6.2.3 Porovnání metod analýzy sentimentu

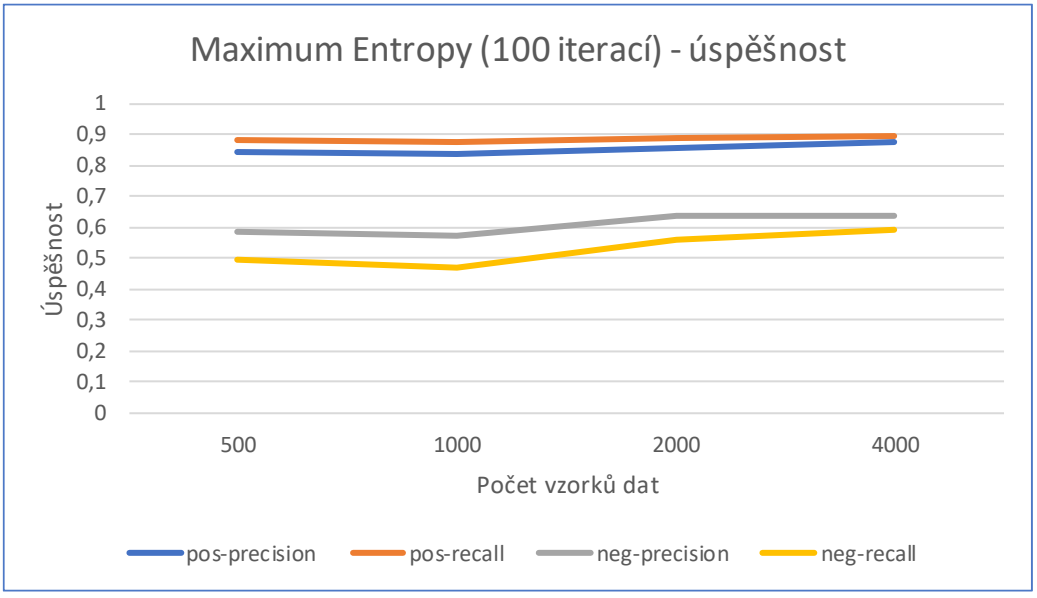
Při porovnávání přesnosti jednotlivých klasifikačních metod nás zajímá úspěšnost klasifikace, vyjádřená metrikami *positive precision*, *positive recall*, *negative precision*, *negative recall* (viz sekce 3.2.4), v závislosti na počtu vzorků vstupních dat.

Pro tento experiment byly náhodně vybrány části datasetu o následujících velikostech: 500, 1000, 2000, 4000 vzorků. Model byl vždy trénován na 90 % zvolených dat a zbylých 10 % bylo využito k otestování. V grafech jsou zprůměrovány výsledky z pěti opakování pro každou hodnotu počtu vzorků.



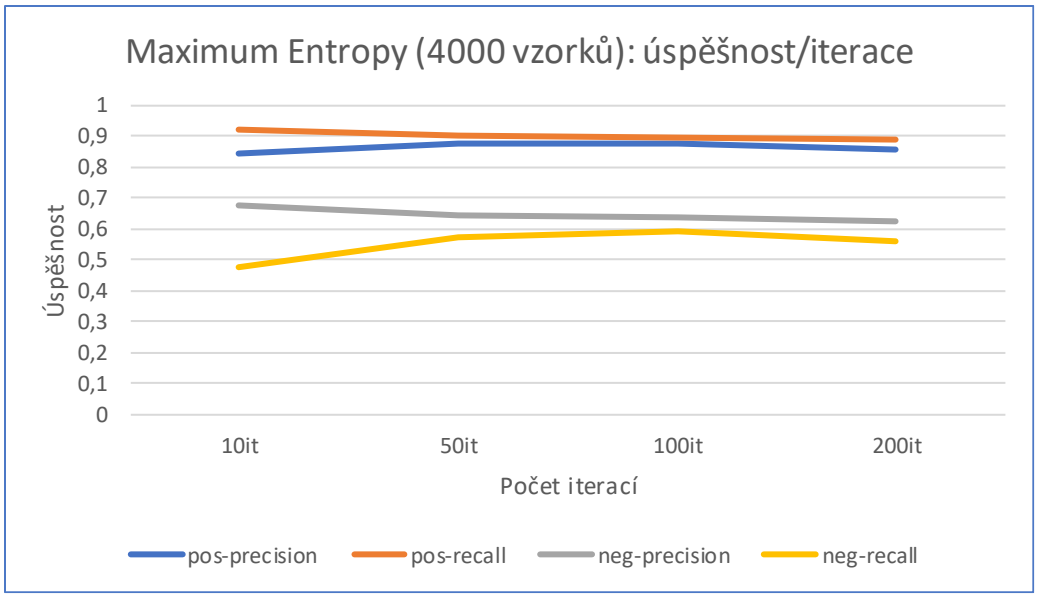
Graf 4: Úspěšnost klasifikátoru Naive Bayes s vektorem příznaků o velikosti 1000 vzhledem k počtu vzorků vstupních dat.

V Graf 4 můžeme vidět, že Naive Bayes má sice vysokou úspěšnost v předvídání třídy *positive*, ale horší výsledky pro třídu *negative*. Např. při 2000 vzorcích zařadil správně méně než polovinu dat, která měla patřit do třídy *negative*. Dále si můžeme všimnout, že další zvýšení počtu vzorků dat zřejmě nebude mít na kvalitu výsledků velký vliv. Stejně tak zvyšování velikosti vektoru příznaků nepřináší žádné zlepšení, naopak spíše mírné zhoršení v metrice *negative precision* (viz Graf 10 v příloze).



Graf 5: Úspěšnost klasifikátoru Maximum Entropy vzhledem k počtu vzorků vstupních dat.

Klasifikátor Maximum Entropy si podle Graf 5 vede lépe. *Positive recall* zůstává velmi dobrý, *positive precision* se dokonce trochu zlepšuje. Hlavně se ale úspěšnost v určování třídy *negative* dostává alespoň nad hranici náhodného odhadu. I zde vidíme, že počet vzorků vstupních dat skoro vůbec neovlivňuje úspěšnost klasifikátoru. Ve snaze o zlepšení bylo vyzkoušeno navýšení počtu iterací trénovacího algoritmu. Graf 6 ale ukazuje, že již od 50 iterací nemá další navyšování význam, a dokonce dochází k velmi pozvolnému snižování úspěšnosti ve všech metrikách.



Graf 6: Úspěšnost klasifikátoru Maximum Entropy vzhlede k počtu iterací trénovacího algoritmu IIS. Použito 4000 vzorků.

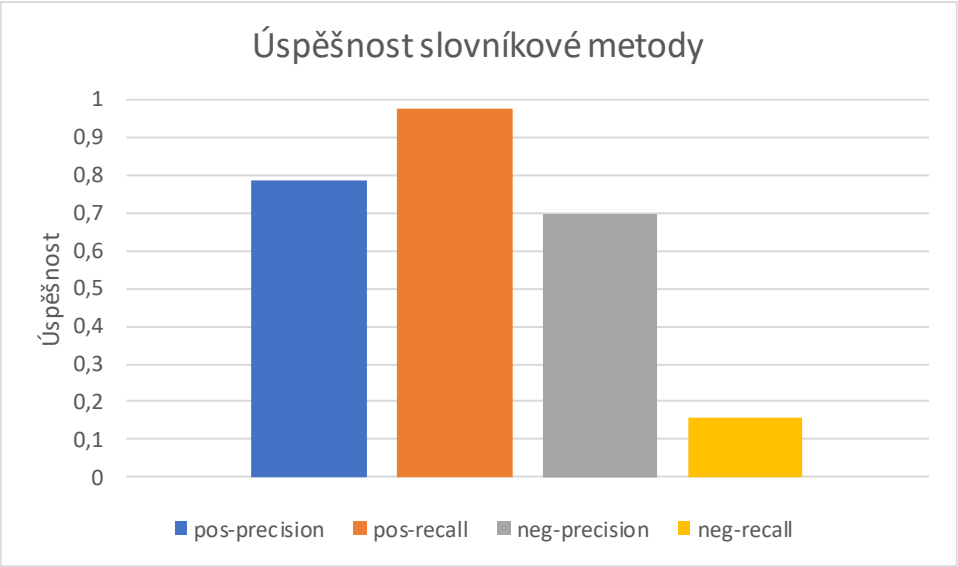
Jiná možná vylepšení metody Maximum Entropy by mohla zahrnovat například:

- jiné příznaky,
- jiný iterativní algoritmus než IIS,
- jiný dataset (např. více vyvážený).

I přesto ale ze srovnání vychází lépe Maximum Entropy. Disponuje lepší úspěšností ve všech zvolených metrikách již od 500 vzorků trénovacích dat. Naive Bayes si hůře poradil s nevyváženým datasetem a zejména v méně zastoupené třídě *negative* byla jeho úspěšnost správného zařazení prvku nižší než Maximum Entropy.

6.2.4 Vyhodnocení úspěšnosti slovníkové metody

Velkou výhodou slovníkové metody je fakt, že nepotřebuje žádná data ani čas k učení. Jedinou prerekvizitou je slovník obsahující slova spolu s jejich sentimentem. Čím kvalitnější tento slovník je, tím lépe bude metoda fungovat. Objevuje se ale jiný problém – různá slova mohou mít v různém kontextu jiný sentiment. Použitý slovník Czech Sublex 1.0 [17] např. uvádí slovo „spoušť“ jako negativní (ve smyslu „Tu spoušť, co nadělal náš kocour, jsme uklízeli celý den.“). V kontextu recenzí fotoaparátů jde ale o pojmenování vlastnosti produktu bez jakéhokoli citového zabarvení. Podobně dopadla slova jako „ostrý“ a „nízký“; podle slovníku mají negativní nádech, nicméně ve větách „Displej je ostrý.“ nebo „Cena je nízká.“ je tomu naopak. S ironií a sarkasmem si pak slovníková metoda nemůže poradit nijak. I přes tato omezení by ale mohla mít určité uplatnění, proto se autorka rozhodla podrobit ji podobnému měření jako metody strojového učení.



Graf 7: Úspěšnost slovníkové metody vyhodnocená pomocí metrik precision a recall.

První tři hodnoty vypadají v Graf 7 velmi slušně, celkový výsledek ale bohužel kazí fakt, že ze slov patřící do třídy *negative* metoda správně zařadila ani ne dvě z deseti. Většinu negativních vět slovníková metoda s použitím slovníku SubLex 1.0 vůbec neodhalila, proto je méně vyhovující pro analýzu sentimentu než výše uvedené metody strojového učení. Mohla by je ale podle [17] vhodným způsobem zajímavě doplňovat – je to jedna z možností dalšího rozšíření této práce.

6.3 Vyhodnocení úspěšnosti aplikace

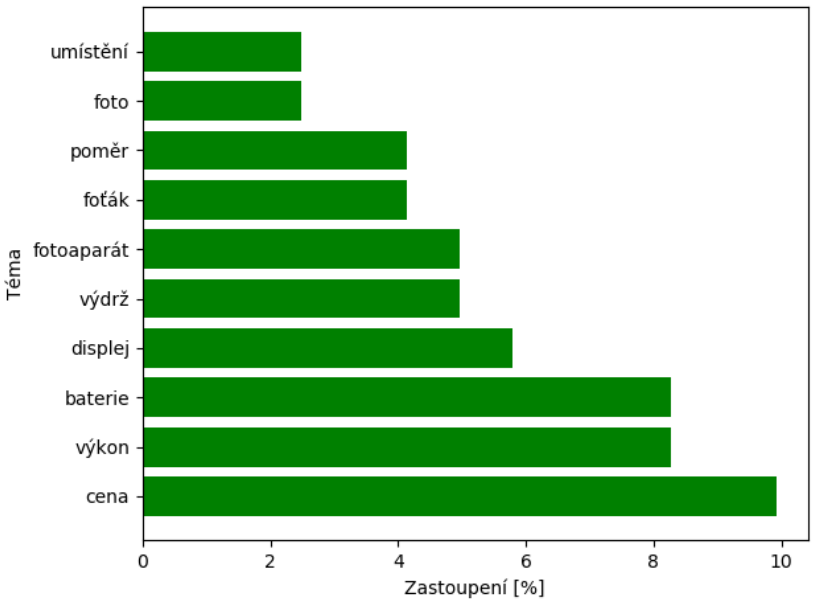
V příloze (Příloha C – Ukázkový výstup aplikace) je ukázán příklad vizualizace implementované analýzy pro koncového uživatele. Ukázkovým produktem je zde mobilní telefon *Xiaomi Redmi Note 7 4GB/64GB*. Dokument typu *.docx*, vygenerovaný automaticky implementovanou aplikací, zobrazuje uživateli nejdůležitější informace získané z recenzí konkrétního produktu. Všechny tyto informace jsou přehledně umístěny, nastylovány a ve většině případů se vejdou na dva listy velikosti A4. Report má následující části:

1) Nadpis, název produktu, URL produktu na Heurece.
2) Sloupcový graf deseti nejčastěji diskutovaných témat.
3) Pět nejčastěji diskutovaných témat včetně identifikace převažujícího sentimentu + tři ukázkové věty recenzí ke každému tématu.
4) Tři nejvíce pozitivní témata (včetně skóre) + tři ukázkové věty recenzí ke každému tématu.
5) Tři nejvíce negativní témat (včetně skóre) + tři ukázkové věty recenzí ke každému tématu.

6.3.1 Nejčastěji diskutovaná témata

V Graf 8 přímo převzatém z této vizualizace si můžeme všimnout, že hned prvních šest nejčastějších témat tvoří vlastnosti produktu (cena, výkon, baterie, displej, výdrž, fotoaparát), které mají

pro uživatele největší význam. Na sedmém místě je slovo „foťák“. To dává prostor k budoucímu vylepšení – slučování synonym – aby se věty o „foťáku“ přiřadily k větám o „fotoaparátu“. Osmý pojem je „poměr“. Ten se v recenzích mobilních telefonů vyskytuje velmi často kvůli velice frekventovaným hodnocením typu „Skvělý poměr cena/výkon.“. Zároveň většinou sdílí reprezentativní věty právě s pojmy „cena“ a „výkon“, je tedy kandidátem na zařazení uživatelem do stop slov, protože sám o sobě nepřináší příliš mnoho nových informací. Poslední dvě místa v seznamu nejčastěji diskutovaných témat zaujímají slova „foto“ a „umístění“. Zatímco „foto“ může nést podobné informace jako „fotoaparát“ či „foťák“, pojem „umístění“ je zajímavá vlastnost, na kterou se podíváme dále v textu.



Graf 8: Výskyt nejvíce komentovaných témat v recenzích mobilního telefonu Xiaomi Redmi Note 7 4GB/64GB. Převzato z výstupu implementované aplikace.

Prvních pět nejčastějších témat je v reportu podrobně rozvedeno. Uživateli se v nadpisu druhé úrovně zobrazí název tématu a převažující názor – pozitivní a negativní. V naprosté většině testovaných případů jsou tato nejčastější témata označena jako pozitivní. Je to dáno několika specifiky daného datasetu. V první řadě musíme uvážit, že nejvíce recenzí získají nejvíce nakupované produkty, je tedy logické, že jejich silné stránky silně převažují nad těmi slabými – jinak by se tolik neprodávaly a v důsledku nekomentovaly. V uvažovaném datasetu je kladných recenzí asi třikrát více než záporných. Tento fenomén se ale projevuje i na nižší úrovni – i většina vlastností produktu nakonec od uživatelů získá více kladných než záporných recenzí (např. „baterie“ mívá často pozitivní i negativní komentáře, plusy ale většinou převažují). Z těchto důvodů aplikace správně vyhodnotí pozitivní ladění většiny vět vztahující se k danému tématu a nejčastěji téma zařadí jako pozitivní.

6.3.2 Nejvíce pozitivní témata

Nejvíce pozitivní témata, tedy to, co uživatelé na produktu nejvíce oceňují, jsou určena na základě skóre sentimentu (více v sekci 5.2.5). Z důvodů uvedených v předchozí sekci jsou tato vybraná témata ve

většině případů shodná s nejčastěji diskutovanými tématy (v tomto příkladu „cena“, „výkon“, „baterie“). Nemusí to ale platit vždy, proto je dobré je pro úplnost v analýze také uvést.

Další vylepšení práce by ale mohlo spočívat v implementaci jiného algoritmu pro výběr reprezentativních vět než v předchozí části. Pak by vybrané zobrazené věty, byť třeba o stejných vlastnostech, mohly přinášet uživateli nové informace.

6.3.3 Nejvíce negativní témata

I zde rozhoduje o zařazení konkrétního tématu do konečného reportu jeho skóre, tentokrát musí být co nejvíce záporné. Většinou bývá v absolutní hodnotě výrazně nižší než skóre silných stránek (opět z důvodů popsaných v 6.3.1), ale téměř vždy se nějaké negativní vlastnosti najdou. V uvedeném příkladu jsou to: „aplikace“, „bateria“, „umístění“.

Téma aplikací je relevantní a pro uživatele užitečné, jak dokazují ukázkové věty (např. „Nefungují aplikace." nebo „Nastavení některých aplikací je problém, buď nejde, nebo složitě, takže pořád kontroluji, jestli zrovna běží, nebo se zase vypnula."). Do výsledků analýzy tedy rozhodně patří.

Zajímavý je výsledek „bateria“ – mohlo by jít o recenzi ve slovenštině, ale spíše se zde vyskytla chyba lemmatizátoru, který v určitém kontextu nedokázal správně lemmatizovat nějaký tvar slova „baterie“. Pokud by lemmatizace proběhla správně, tyto recenze by se sloučily s hodnoceními pojmu „baterie“, kterému by tím snížily skóre sentimentu, ale stále by pravděpodobně zůstal v pozitivních vlastnostech.

Posledním zobrazeným negativním tématem je „umístění“. Toto slovo se dokonce vyskytuje v nejčastěji komentovaných (na desátém místě), je tedy velmi vhodné, že se téma zahrne do analýzy pro uživatele i podrobněji v této části reportu. Vybrané věty „Umístění reproduktoru při hraní her." a „Špatné umístění notifikační diody." dokládají, že skutečně existují negativní hodnocení v této oblasti.

6.4 Shrnutí experimentů

V rámci provedených experimentů byl nejprve představen dataset, vytvořený pomocí vlastního skriptu pro stahování recenzí z Heureky. Poté byl otestován implementovaný způsob hledání nejčastěji komentovaných témat, jehož výsledky se ukázaly jako uspokojivé a použitelné pro výslednou aplikaci.

Dále byly porovnány dvě metody klasifikace, Naive Bayes a Maximum Entropy, a to z hlediska času potřebného k učení a přesnosti při vlastní klasifikaci. Úspěšnost byla vyjádřena metrikami *precision* a *recall*. Z tohoto srovnání vyšla lépe metoda Maximum Entropy díky větší přesnosti v identifikaci vět s negativním sentimentem. Nakonec byla podobná měření provedena i pro slovníkovou metodu analýzy sentimentu. Výsledky byly znatelně horší než u Maximum Entropy, avšak rychlost této metody a fakt, že nepotřebuje trénovací data, nabízí určité možnosti do budoucna; metoda by mohla být použita například k vylepšení vektoru příznaků pro Maximum Entropy.

Na závěr byly popsány a zhodnoceny výsledky samotné implementované aplikace. Výstupem je dokument ve formátu *.docx* obsahující analýzu zpracovanou pro uživatele. Report zahrnuje sloupcový graf nejčastěji diskutovaných témat, ukázkové věty k těmto tématům, a také seznam nejvíce pozitivních a negativních témat v recenzích, taktéž včetně vybraných vět.

Bylo zjištěno, že výsledky aplikace hodně závisí na charakteristikách vstupních recenzí. Čím kvalitnější jsou poskytnutá data (množství recenzí pozitivních i negativních, celé smysluplné věty, málo překlepů…), tím lepší jsou i výstupy. Při analýze populárních modelů mobilních telefonů (alespoň přes 100 recenzí) dosahuje aplikace většinou slušných výsledků, které by mohly skutečně pomoci reálnému zákazníkovi.

7 Závěr

Tato práce se zabývala problematikou získávání znalostí z textu, konkrétně z uživatelských recenzí produktů prodávaných na internetu. Motivací bylo vytvořit nástroj, který by pomohl zákazníkům internetových obchodů zorientovat se v možných desítkách až stovkách recenzí, které může jednotlivý produkt mít, a poskytl ucelenou analýzu nejvíce diskutovaných témat v těchto hodnoceních.

Pro dosažení efektivního zpracování recenzí a odvozování jejich významu byly vyžadovány vědomosti z oblasti dolování v textu. Autorka tedy nejprve nastudovala informace o zpracování přirozeného jazyka, strojovém učení, analýze sentimentu a další. Příslušná látka je shrnuta v kapitolách 2 a 3. Poté byly prozkoumány již existující práce na dané téma. Těch existuje vícero, avšak většinou se zabývají pouze angličtinou. Přesto byly některé postupy a informace inspirující nebo přímo použitelné i v kontextu češtiny, proto jsou práce představeny v kapitole 4.

V kapitole 5 byl navržen algoritmus, který dokáže nalézt nejvíce komentovaná témata a určit, zda se o tématu píše spíše pozitivně či negativně. Dále byla v několika sekcích popsána implementace tohoto návrhu. Pro analýzu sentimentu byla implementována dokonce tři různá řešení. Kapitola 6 se věnovala porovnání těchto řešení a dalším experimentům, včetně podrobného vyhodnocení úspěšnosti výsledné aplikace.

Případné pokračování práce vidí autorka především ve vylepšení úspěšnosti metody analýzy sentimentu Maximum Entropy. Bylo by možné například vyzkoušet modernější iterativní algoritmy než IIS, zlepšit způsob zpracování a transformace dat určených pro trénování, nebo se více zaměřit na výběr příznaků. V tom by mohla být nápomocná i slovníková metoda, která sice byla implementována, ale při experimentech se ukázalo, že není příliš vhodná k samostatnému použití. Třeba by se tedy dala využít spíše ke zdokonalení strojového učení. [17] Další nápady na menší vylepšení se týkají například využití emotikonů v recenzích k určování sentimentu, nebo vytvoření vlastního slovníku synonym. Vzhledem k tomu, že vstupní data obsahují množství překlepů, dílčí zlepšení výsledků by také mohla přinést jejich automatizovaná oprava.

Specifikem práce byla čeština. Zatímco v angličtině jsou témata této práce už dlouhou dobu velmi populární a existuje množství řešení pro jednotlivé popisované problémy, v našem jazyce stále chybí například větší výběr nástrojů na lemmatizaci a morfologickou analýzu. Totéž se týká různých druhů slovníků, korpusů a datasetů; není jich mnoho, a ne vždy se dá najít ten pravý bez kompromisů. Proto autorka nad rámec zadání vytvořila i skript pro tvorbu vlastního datasetu.

Výstupy implementované aplikace a výsledky experimentů (viz sekce 6.3) potvrzují, že vytvoření české aplikace v oblasti zpracování přirozeného jazyka je možné. Spojení dostupných nástrojů (český morfologický analyzátor MorphoDiTa, klasifikační algoritmy poskytované knihovnou NLTK, slovník sentimentu Czech Sublex a další) s vhodně navrženým algoritmem do uceleného systému přineslo

výsledky přínosné pro koncového uživatele. Tyto výsledky jsou navíc uživateli prezentovány v atraktivní a srozumitelné formě v dokumentu formátu *.docx*.

Této práci se podařilo zmapovat možnosti analýzy textů uživatelských recenzí a na základě testů a měření porovnat možné přístupy. Výsledná aplikace umožňuje analýzu recenzí produktů a výsledky shrnuje uživatelsky přívětivým způsobem. Uživatelé si tak mohou lépe a rychleji udělat představu o silných a slabých stránkách produktu jejich zájmu.

8 Bibliografie

[1] DAS, Dipanjan a André F. T. MARTINS. *A Survey on Automatic Text Summarization*. In: . 2007.

[2] PARALIČ, Ján, Karol FURDÍK, Gabriel TUTOKY, Peter BEDNÁR, Martin SARNOSKÝ, Peter BUTKA a František BABIČ. *Dolovanie znalostí z textov*. Prvé. Košice: Equilibria, s.r.o., Košice, 2010. ISBN 978-80-89284-62-7.

[3] Český stoplist. *Centrum zpracování přirozeného jazyka: Fakulta informatiky na Masarykově univerzitě* [online]. b.r. [cit. 2019-01-02]. Dostupné z: https://nlp.fi.muni.cz/cs/StopList

[4] AGRAWAL, Rakesh a Ramakrishnan SRIKANT. Fast Algorithms for Mining Association Rules in Large Databases. In: *Proceedings of the 20th International Conference on Very Large Data Bases*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994, s. 487-499. ISBN 1-55860-153-8. Dostupné také z: http://dl.acm.org/citation.cfm?id=645920.672836

[5] *The text mining handbook: advanced approaches in analyzing unstructured data*. 1st edition. New York: Cambridge University Press, 2007. ISBN 978-0-521-83657-9.

[6] SYCHRA, Martin. *Analýza sentimentu s využitím dolování dat*. Brno, 2016. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Vladimír Bartík.

[7] ZÍBAR, Karel. *Analýza sentimentu v sociálních sítích*. Plzeň, 2014. Bakalářská práce. Západočeská univerzita v Plzni, Fakulta aplikovaných věd, Katedra informatiky a výpočetní techniky. Vedoucí práce Josef Steinberger.

[8] TABOADA, Maite, Julian BROOKE, Milan TOFILOSKI, Kimberly VOLL a Manfred STEDE. Lexicon-Based Methods for Sentiment Analysis. In: *Computational Linguistics*. 2011, **37**(2), s. 267-307. DOI: 10.1162/COLI_a_00049. ISSN 0891-2017. Dostupné také z: http://www.mitpressjournals.org/doi/10.1162/COLI_a_00049

[9] BIRD, Steven, Ewan KLEIN a Edward LOPER. *Natural language processing with Python*. Cambridge [Mass.]: O'Reilly, 2009. ISBN 05-965-1649-5.

[10] PATOČKA, Michal. *Metody strojového učení pro analýzu sentimentu*. Plzeň, 2013. Diplomová práce. Západočeská univerzita v Plzni, Fakulta aplikovaných věd, Katedra informatiky a výpočetní techniky. Vedoucí práce Ivan Habernal.

[11] Pojmy precision a recall. *Wiki Český národní korpus* [online]. b.r. [cit. 2019-05-21]. Dostupné z: https://wiki.korpus.cz/doku.php/pojmy:precision

[12] TURNEY, Peter. Thumbs up or thumbs down?. In: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02*. Morristown, NJ, USA: Association for Computational Linguistics, 2002, s. 417-422. DOI: 10.3115/1073083.1073153. Dostupné také z: http://portal.acm.org/citation.cfm?doid=1073083.1073153

[13] HU, Minqing a Bing LIU. Mining Opinion Features in Customer Reviews. In: *Proceedings of the 19th National Conference on Artifical Intelligence*. San Jose, California: AAAI Press, 2004, s. 755-760. ISBN 0-262-51183-5.

[14] HU, Minqing a Bing LIU. Mining and summarizing customer reviews. In: *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '04*. New York, New York, USA: ACM Press, 2004, s. 168-177. DOI: 10.1145/1014052.1014073. ISBN 1581138889. Dostupné také z: http://portal.acm.org/citation.cfm?doid=1014052.1014073

[15] WANG, Dingding, Shenghuo ZHU a Tao LI. SumView: A Web-based engine for summarizing product reviews and customer opinions. In: *Expert Systems with Applications*. Tarrytown, NY, USA: Pergamon Press, Inc., 2013, **40**(1), s. 27-33. DOI: 10.1016/j.eswa.2012.05.070. ISSN 09574174. Dostupné také z: https://linkinghub.elsevier.com/retrieve/pii/S0957417412007865

[16] LIU, LiZhen, WenTao WANG a HangShi WANG. Summarizing customer reviews based on product features. In: *2012 5th International Congress on Image and Signal Processing*. IEEE, 2012, s. 1615-1619. DOI: 10.1109/CISP.2012.6469932. ISBN 978-1-4673-0964-6. Dostupné také z: http://ieeexplore.ieee.org/document/6469932/

[17] VESELOVSKÁ, Kateřina a Ondřej BOJAR. *Czech SubLex 1.0* [online]. In: . LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University, 2013 [cit. 2019-05-20]. Dostupné z: http://hdl.handle.net/11858/00-097C-0000-0022-FF60-B

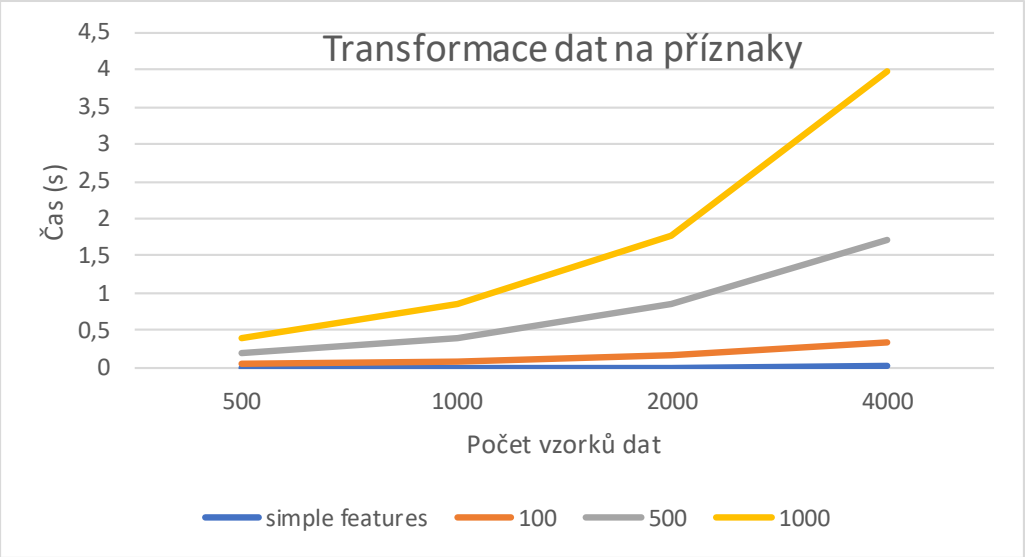
[18] STRAKOVÁ, Jana, Milan STRAKA a Jan HAJIČ. Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition. In: *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations* [online]. Baltimore, Maryland: Association for Computational Linguistics, 2014, s. 13-18 [cit. 2019-05-20].

9 Příloha A – Použitý seznam stop slov

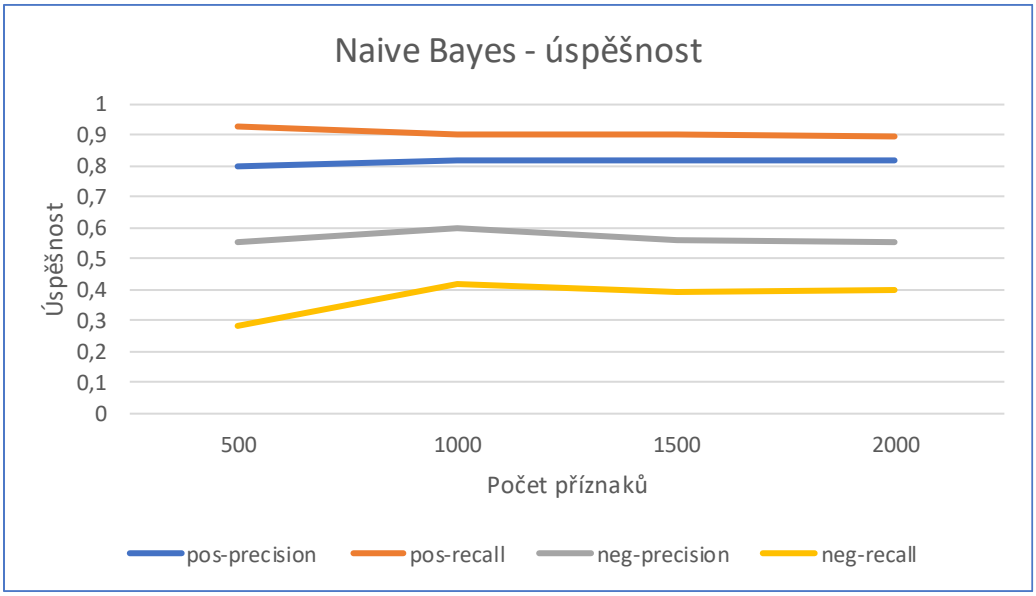
..., ., ,, :, ;, !, ?, (,), /, *, %, &, že, než, a, s, k, o, i, u, v, z, že, dnes, temto, budeš, budem, byli, jseš, muj, svuj, ta, tomto, tohle, tuto, tyto, jej, zda, proc, máte, tato, kam, tohoto, kdo, kterí, mi, nem, tom, tomuto, met, nic, proto, kterou, byla, toho, protože, asi, ho, naši, re, což, tem, takže, svých, její, svými, jste, aj, tu, tedy, teto, bylo, kde, ke, prave, ji, nad, nejsou, ci, pod, tema, mezi, pres, ty, pak, všem, ani, když, však, jsem, tento, aby, jsme, pred, pta, jejich, byl, ješte, až, bez, také, pouze, prvne, vaše, který, nás, pokud, jeho, své, jiné, není, vás, jen, podle, zde, už, být, bude, již, než, které, by, která, co, nebo, ten, tak, me, pri, od, po, jsou, jak, ale, si, se, ve, to, jako, za, ze, do, pro, je, na, atd, atp, jakmile, pricemž, jí, on, ona, ono, oni, ony, my, vy, ji, mi, mne, jemu, tomu, tem, nemu, nemuž, jehož, jelikož, jež, jakož, nacež, má, zatím, já, všechen

Seznam je založen na seznamu stop slov použitém a zveřejněném v [10]. Některá slova byla přidána, odebrána nebo upravena, aby seznam lépe vyhovoval účelům implementované aplikace.

10 Příloha B – Doplňující grafy ke kapitole Experimenty



Graf 9: Čas potřebný k transformaci vstupních dat na příznaky vektorů při různých velikostech výsledného vektoru.



Graf 10: Úspěšnost klasifikátoru Naive Bayes při 4000 vzorků vstupních dat vzhledem k velikosti vektoru příznaků. Nejlepší výsledky dostáváme při počtu 1000 příznaků.

11 Příloha C – Ukázkový výstup aplikace

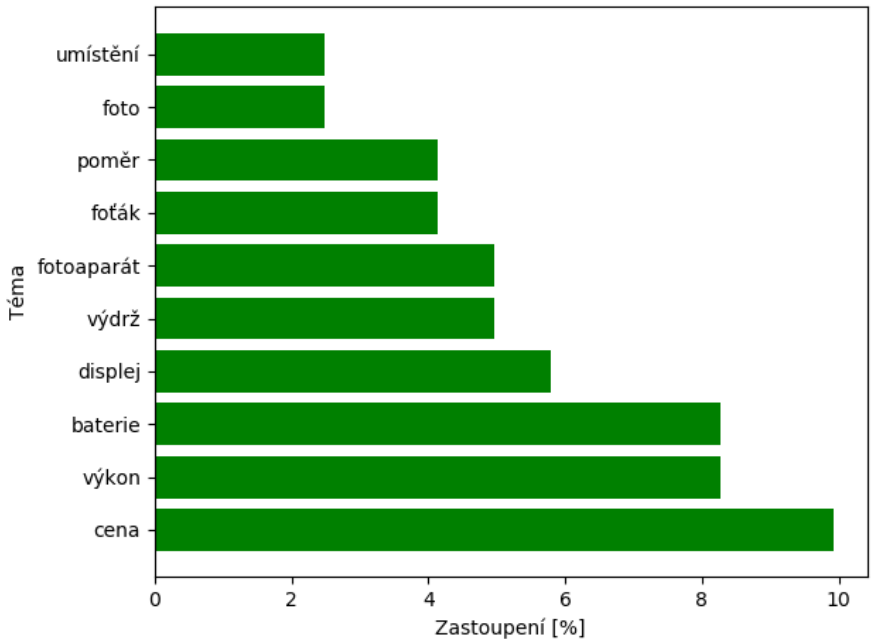
Příloha C se nachází na následujících dvou listech.

Analýza recenzí produktu

Xiaomi Redmi Note 7 4GB/64GB

https://mobilni-telefony.heureka.cz/xiaomi-redmi-note-7-4gb-64gb/

Nejčastěji diskutovaná témata



cena (převažující názor: pozitivní)

- "Cena je příznivá ."
- "Jednoduché ovládání,výborná cena ."
- "Poměr cena/výkon ."

výkon (převažující názor: pozitivní)

- "Dostačující výkon ."
- "Dostatečný výkon na všechno možné ."
- "Poměr cena/výkon ."

baterie (převažující názor: pozitivní)

- "Skvělá výdrž baterie ."
- "Silná baterie,rychlonabijeni, ."
- "Nadstandardní výdrž baterie ."

displej (převažující názor: pozitivní)

- "Jen malý výřez (kapka) v displeji ."
- "Displej má hezké barvy ."
- "Velký displej ."

výdrž (převažující názor: pozitivní)

- "Skvělá výdrž baterie ."
- "Nadstandardní výdrž baterie ."
- "Výdrž baterie ."

Co uživatelé nejvíce oceňují?

cena (skóre: 10)

- "Cena je příznivá ."
- "Jednoduché ovládání,výborná cena ."
- "Poměr cena/výkon ."

výkon (skóre: 10)

- "Dostačující výkon ."
- "Dostatečný výkon na všechno možné ."
- "Poměr cena/výkon ."

baterie (skóre: 10)

- "Skvělá výdrž baterie ."
- "Silná baterie,rychlonabijeni, ."
- "Nadstandardní výdrž baterie ."

Co jsou největší nedostatky?

aplikace (skóre: -3)

- "MIUI otravne, bez rootu neumlcitelne aplikace :-(."
- "Nefungují aplikace ."
- "Nastavení některých aplikací je problém,buď nejde,nebo složitě, takže pořád kontroluji,jestli zrovna běží,nebo se zase vypnula ."

bateria (skóre: -2)

- "Baterii musí měnit v servisu ."
- "Baterie ."

umístění (skóre: -1)

- "Umístění reproduktoru při hraní her ."
- "Špatné umístění notifikační diody ."